Title

A mini-review of single-cell Hi-C embedding methods.

Authors

Ma, Rui

Huang, Jingong

Jiang, Tao

et al.

Publication Date

2024-12-01

DOI

10.1016/j.csbj.2024.11.002

Peer reviewed

Computational and Structural Biotechnology Journal 23 (2024) 4027–4035

Contents lists available at ScienceDirect

Computational and Structural Biotechnology Journal

journal homepage: www.elsevier.com/locate/csbj

Mini-Review

A mini-review of single-cell Hi-C embedding methods

Rui Ma [a], Jingong Huang [b], Tao Jiang [b,c,*], Wenxiu Ma [a,c,*]

[a] Department of Statistics, University of California Riverside, 900 University Ave., Riverside, 92521, CA, USA
[b] Department of Computer Science and Engineering, University of California Riverside, 900 University Ave., Riverside, 92521, CA, USA
[c] Institute of Integrative Genome Biology, University of California Riverside, 900 University Ave., Riverside, 92521, CA, USA

ARTICLE INFO

Keywords:
Single-cell Hi-C
Genome architecture
Embedding
Dimensionality reduction

ABSTRACT

Single-cell Hi-C (scHi-C) techniques have significantly advanced our understanding of the 3D genome organization, providing crucial insights into the spatial genome architecture within individual nuclei. Numerous computational and statistical methods have been developed to analyze scHi-C data, with embedding methods playing a key role. Embedding reduces the dimensionality of complex scHi-C contact maps, making it easier to extract biologically meaningful patterns. These methods not only enhance cell clustering based on chromatin structures but also facilitate visualization and other downstream analyses. Most scHi-C embedding methods incorporate strategies such as normalization and imputation to address the inherent sparsity of scHi-C data, thereby further improving data quality and interpretability. In this review, we systematically examine the existing methods designed for scHi-C embedding, outlining their methodologies and discussing their capabilities in handling normalization and imputation. Additionally, we present a comprehensive benchmarking analysis to compare both embedding techniques and their clustering performances. This review serves as a practical guide for researchers seeking to select suitable scHi-C embedding tools, ultimately contributing to the understanding of the 3D organization of the genome.

1. Introduction

Over the past two decades, researchers have extensively investigated the three-dimensional (3D) organization of the genome [1–5]. Within the confined 3D space of the cell nucleus, DNA—the genetic material of the cell—is intricately compacted and organized [5]. The development of chromatin conformation capture (3C) technology [6] marked a significant breakthrough, enabling the inference of spatial proximity between genomic loci based on the frequencies of chromatin contacts within the nuclei. This innovation paved the way for various 3C-based techniques, such as 4C [7], 5C [8], Hi-C [9,10], Micro-C [11,12], ChIA-PET [13,14], and Hi-ChIP [15]. These techniques were developed to profile chromatin contacts in a higher-throughput manner and have been instrumental in revealing the multi-scale organization of the 3D genome, offering profound insights into nuclear architecture and gene expression regulation [16,17].

Among the various 3C-based techniques, Hi-C has been widely employed to study the 3D genome architecture. However, the variability in chromatin contacts across cells, even within a functionally homogeneous population, arises from the stochastic nature of chromatin conformation and spatial genome organization [18]. Consequently, while Hi-C effectively captures the spatial arrangements of complex chromatin structures, relying solely on Hi-C data is considered insufficient for depicting the diversity of higher-order chromosome structures at the single-cell level. To address this limitation, several single-cell Hi-C (scHi-C) techniques [19–28] have been developed. These advancements have enabled the investigation of multi-scale spatial genome organization at the single-cell level, yielding invaluable insights into the dynamics and variability of the 3D genome [17,18,29].

Single-cell 3D mapping techniques, developed to study the 3D genome architecture at the single-cell level, can be broadly categorized into the following three groups [16,17]: imaging-based protocols [18,30,31], proximity ligation-based protocols [19–25], and ligation-free protocols [32]. The imaging-based methods visualize chromatin targets within cells as fluorescently labeled spots, thereby detecting chromatin contacts based on the spatial positions of imaged loci. The proximity ligation-based techniques, including the aforementioned 3C-based methods, measure the frequencies of chromatin contacts between genomic loci by digesting crosslinked DNA with enzymes, ligating restriction fragments, and quantifying the sheared and purified frag-

* Corresponding authors at: University of California Riverside, 900 University Ave., Riverside, 92521, CA, USA.
E-mail addresses: jiang@cs.ucr.edu (T. Jiang), wenxiu.ma@ucr.edu (W. Ma).

https://doi.org/10.1016/j.csbj.2024.11.002
Received 10 August 2024; Received in revised form 1 November 2024; Accepted 1 November 2024
Available online 7 November 2024

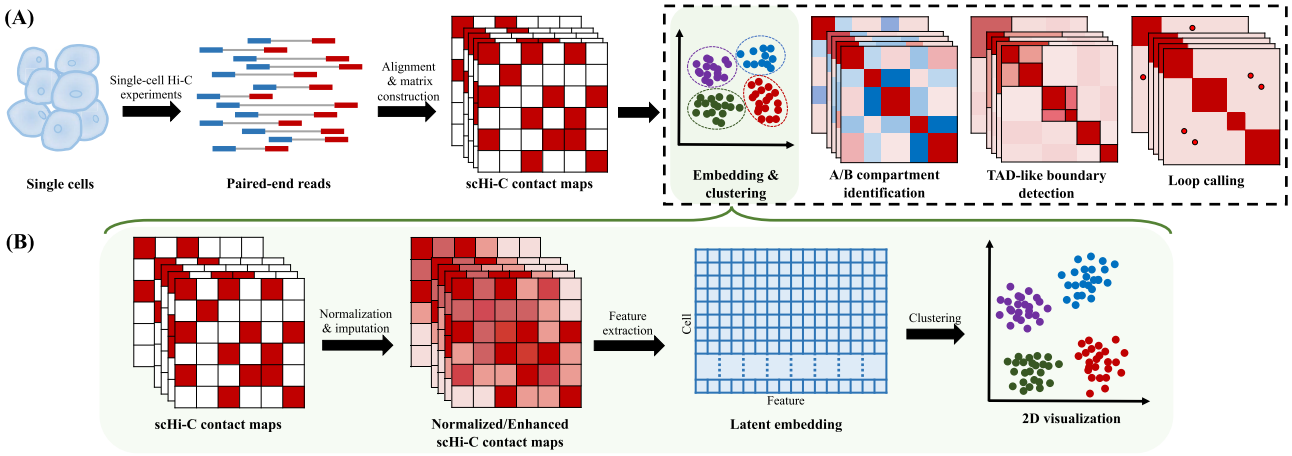

Fig. 1. Single-cell Hi-C analysis workflow. (A) A simplified workflow of scHi-C data analysis; (B) Typical scHi-C embedding workflow: scHi-C contact maps serve as input and often undergo normalization and/or imputation prior to dimensionality reduction. This process extracts important features and outputs latent embeddings for further analysis, such as clustering.

ments through high-throughput paired-end sequencing. In contrast, the ligation-free approaches, such as the "single-cell split-pool recognition of interactions by tag extension" (scSPRITE), provide novel insights into 3D genome topology. Additionally, single-cell simultaneous profiling techniques have been developed to investigate the correlation between chromatin contact frequencies and functional characteristics, such as DNA methylation [26–28] and gene expression [33,34]. Among these single-cell 3D mapping approaches, scHi-C, a proximity ligation-based technique, has been extensively employed to explore the heterogeneity and dynamics of 3D genome organization.

Recently, several embedding methods [35–43] have been developed to improve the analysis and interpretation of scHi-C data. Embedding, synonymous with dimensionality reduction, extracts lower-dimensional features from the original 2D chromatin contact maps, which represent genomic interaction. This process seeks to capture essential patterns while eliminating redundant or noisy information, thereby enhancing computational efficiency and the effectiveness of subsequent analyses, such as clustering, visualization, and differential analysis. ScHi-C embedding methods are particularly useful for distinguishing different cell (sub)types, identifying clusters of cells of the same (sub)type, and visualizing cell separation and clustering. By revealing cell-type-specific features, these methods help uncover underlying patterns in complex, large-scale single-cell datasets.

In contrast to one-dimensional genomic sequencing data, such as single-cell RNA-seq (scRNA-seq) or single-cell ATAC-seq (scATAC-seq), scHi-C data presents complex, hierarchical information within a 2D contact map, adding complexity to the embedding task. Additionally, scHi-C contact maps exhibit significantly higher sparsity compared to those of the traditional Hi-C or other single-cell genomic datasets. While assays like scRNA-seq and scATAC-seq typically reflect approximately 70% of the genome, scHi-C is often limited to less than 5% of all possible contacts [44]. To address the challenges posed by this sparsity, as well as to mitigate systematic biases and reduce experimental noise, various strategies have been developed, including normalization and imputation [36,38,39,42,43,45]. When applied prior to embedding, these strategies can significantly enhance the performance of scHi-C embedding, clustering, and other downstream analyses.

In this review article, we summarize ten recently developed computational methods designed to improve the embedding of scHi-C contact maps. We outline their methodologies and discuss their capabilities in normalization, imputation, and batch effect correction by focusing on their strengths and limitations. Additionally, we present a comprehensive benchmarking analysis that evaluates and compares the performances of these embedding techniques and their impact on the subsequent clustering results. This review aims to guide researchers in selecting the most appropriate methods for their studies.

2. ScHi-C embedding methods

Starting with the sequencing reads generated by scHi-C experiments, several pre-processing steps are required to create the 2D scHi-C contact maps. First, the paired-end reads are mapped to the reference genome to identify the loci of interacting chromatin fragments. Next, a quality control assessment is performed on the mapped reads to remove duplicates and erroneous pairs. In addition to this read filtering, cell filtering can be performed by excluding low-quality cells based on sequencing depth and the ratio of intra-chromosomal contacts to inter-chromosomal contacts. Following both read- and cell-level filtering, the remaining read pairs are used to construct the matrices of scHi-C contact frequencies by binning contacts with a fixed bin size (referred to as "resolution") for each cell. These scHi-C contact matrices serve as the foundation for various analytical tasks, including embedding, clustering, and investigating 3D genome features, such as A/B compartment identification, TAD-like boundary detection, and loop calling (Fig. 1A).

Recently, embedding methods have been developed for scHi-C contact maps to facilitate cell clustering and other downstream analyses. These methods take scHi-C contact matrices as input, extract important features, and output a latent embedding matrix with cell-by-feature dimensions, thereby reducing the complexity of the scHi-C data (Fig. 1B). The resulting embeddings can then be further reduced and projected onto a lower-dimensional subspace, typically visualized in a 2D scatter plot, where each dot symbolizes an individual cell. Similar to other single-cell genomics data, such as scRNA-seq and scATAC-seq, this 2D projection helps researchers differentiate and cluster cells for subsequent cell-type-specific and differential analyses.

To date, ten published computational methods have been specifically designed for the embedding of scHi-C data, including HiCRep/MDS [35], scHiCluster [36], Topic Modeling [37], scHiCTools [38], Higashi [39], scHiCExplorer [40], scHiCEmbed [41], BandNorm [42], scVI-3D [42], and Fast-Higashi [43]. These methods can be broadly categorized into two main groups: (1) deep learning-based methods, such as Higashi, scHiCEmbed, and scVI-3D, and (2) statistical methods, which include the remaining seven methods. These embedding methods employ various approaches by using either statistical techniques or neural networks, to generate latent embeddings from scHi-C datasets. Researchers can then apply conventional dimensionality reduction techniques, such as Uniform Manifold Approximation and Projection (UMAP) [46] and t-Distributed Stochastic Neighbor Embedding (t-SNE) [47], to project

Table 1
Summary of scHi-C embedding tools. This table outlines the core methodologies of each scHi-C embedding tool and summarizes their capabilities, including features such as contact matrix normalization, imputation, and batch effect removal.

Tools	Embedding strategies	Normalization	Imputation	Batch effect removal
BandNorm [42]	PCA on Combined Bin-pairs	✓		✓
Fast-Higashi [43]	Tensor Decomposition	✓	✓	✓
Higashi [39]	Hyper-SAGNN	✓	✓	✓
scHiCExplorer [40]	MinHash-kNN graph followed by PCA	✓		
scHiCluster [36]	Two-step PCA		✓	
scHiCEmbed [41]	Two-step PCA		✓	
HiCRep/MDS [35]	Pairwise Similarity followed by MDS		✓	
scHiCTools [38]	Pairwise Similarity followed by One Dimensional Reduction Method	✓	✓	
scHi-C Topics [37]	Latent Dirichlet Allocation followed by PCA			
scVI-3D [42]	Non-linear Latent Factor Model	✓	✓	✓

these embeddings onto a lower-dimensional subspace for visualization of further analysis.

To address the challenges in scHi-C data analysis, various embedding tools incorporate normalization and/or imputation into their pipelines to improve feature extraction. Normalization adjusts technical variability, such as differences in sequencing depth and library size across cells, ensuring that chromatin contacts are comparable throughout the dataset. Imputation helps recover missing or low-frequency chromatin contacts to address the inherent sparsity of scHi-C data. Additionally, some tools integrate batch effect removal to account for non-biological variations, such as differences in laboratory conditions, ensuring that the clustering results reflect true biological differences rather than technical artifacts. Below, we briefly discuss the scHi-C embedding tools and summarize their functions and strategies provided in Table 1.

2.1. HiCRep/MDS

Liu et al. [35] were the first to investigate the feasibility of embedding scHi-C data using methods originally developed for bulk Hi-C analysis. The authors evaluated one custom-designed Hi-C distance measure and three existing Hi-C similarity measures (HiCRep [48], GenomeDISCO [49], and HiC-Spector [50]) by combining each with the Multidimensional Scaling (MDS) [51] embedding method. Combining HiCRep with MDS was shown to effectively embedd scHi-C data into a low-dimensional space, revealing biological variations of 3D chromatin organization, even in datasets with low sequencing depth. While this similarity-based embedding approach effectively captures cell cycle dynamics, it struggles with forming distinct clusters of cell types and differentiating between chromatin structures [36]. Additionally, the method is computationally demanding due to the need for smoothing and pairwise comparisons among individual cells [37].

2.2. scHiCluster

scHiCluster, introduced by Zhou et al. [36], is one of the initial tools specifically designed for clustering scHi-C data. Its imputation approach combines linear convolution with random walk, thereby effectively addressing the inherent sparsity of scHi-C data and enabling accurate clustering of single cells and the identification of cell-type-specific features of 3D genome organization, such as TAD-like structures. The method first utilizes linear convolution to smooth each bin-pair with its neighbors and then employs a random walk with restart (RWR) algorithm [52] to effectively capture both the local and global information of the scHi-C contact maps. To mitigate coverage bias, scHiCluster selects only the top 20% of contacts before applying Principal Component Analysis (PCA) to project the data into a low-dimensional subspace. This approach preserves essential features while reducing data complexity, thereby facilitating the differentiation of various cell types even within the same cell cycle stage. Although scHiCluster does not include a built-in batch effect removal feature, the use of Harmony [53], a tool widely used for integrating scRNA-seq data, has been suggested to manage batch effects. Furthermore, although scHiCluster did not explicitly detail their downstream feature calling functions in the paper, their GitHub repository offers users commands for calling compartments, domains, and loops using, which are derived from other published methods.

2.3. scHi-C topics

scHi-C Topics, introduced by Kim et al. [37], leverages Latent Dirichlet Allocation (LDA) topic modeling, by providing a novel approach for scHi-C data embedding. Topic modeling has been widely used in natural language processing to uncover latent structures in large-scale, sparse, and discrete datasets. The application of LDA to scHi-C data builds on its successful use in scATAC-seq data for learning latent-space representations [54]. This method treats individual cells as “documents” and locus-pair contacts as “words” to generate two relationship matrices: (1) topics and cells; (2) topics and locus pairs. This is to identify topics that represent the distinctive features of different cell types. Specifically, Kim et al. successfully applied LDA to decompose the cell-by-locus pair matrix—derived from locus pairs within a 10 Mb genomic distance for each cell—into a cell-by-topic matrix and a topic-by-locus pair matrix. These topics facilitate the discovery of crucial locus pairs responsible for functional and structural differences across various cell types. By analyzing these cell-type-specific topics, the authors demonstrated the ability to reveal significant compartmental patterns, enrichment, and the finer dynamics of 3D genome topology. Furthermore, this scHi-C Topics method was shown to effectively cluster cells by type and separates cell cycle effects from 3D chromatin organization in scHi-C data.

2.4. scHiCTools

Among various methods, scHiCTools [38] stands out as a highly versatile toolkit that offers a variety of imputation and embedding approaches specifically tailored for scHi-C data. The core concept of scHiCTools is to effectively derive latent embeddings by leveraging pairwise cell similarity. This software supports a wide range of input formats for scHi-C contact maps, such as pre-processed matrices, edge lists, hic files, and cool files, as well as tools to summarize the quality of data. To address the inherent sparsity of scHi-C data, scHiCTools offers several user-selectable normalization strategies such as observed/expected (OE) normalization [9], Knight-Ruiz (KR) normalization [55], and Vanilla coverage (VC) normalization [9]. It also offers several imputation options, including linear convolution, random walk, network enhancing. Linear convolution smooths chromatin contacts over neighboring elements, while random walk captures both local and global signals across the genome. Network enhancing is a special type of random walk that was initially developed for bulk Hi-C data to enhance the contact map and improve the detection of TAD boundaries [56].

Following imputation, scHiCTools computes the cell-to-cell similarity matrix using Hi-C similarity measures, such as InnerProduct, fastHiCRep (a faster version of HiCRep [48]), and Selfish [57], to generate the latent embeddings. The software also incorporates multiple clustering approaches for comprehensive analysis. By projecting cells onto a

Table 2
Software tools of scHi-C embedding methods and their computational efficiency. This table summarizes the computational performance of various embedding tools, including CPU and GPU utilization, and the approximate runtime for analyzing two scHi-C datasets at the 500kb-resolution with different numbers of cells and sequencing depths for method comparison. The programming languages and software websites used for implementing these tools are also listed.

Tools	CPU	GPU	Runtime (Nagano et al.)	Runtime (Tan et al.)	Programming languages	Software URLs
BandNorm [42]	✓		~ 15 min	~ 25 min	R	github.com/keleslab/BandNorm
Fast-Higashi [43]	✓	✓	~ 6 min (on GPU)	~ 10 min (on GPU)	Python	github.com/ma-compbio/Fast-Higashi
Higashi [39]	✓	✓	~ 8.5 hrs (on GPU)	~ 8 hrs (on GPU)	Python	github.com/ma-compbio/Higashi
scHiCExplorer [40]	✓		~ 12 min	~ 25 min	Python	github.com/joachimwolff/scHiCExplorer
scHiCluster [36]	✓		~ 1.5 hrs	~ 2 hrs	Python	github.com/zhoujt1994/scHiCluster
scHiCEmbed [41]	✓	✓			R & Python	dna.cs.miami.edu/scHiCEmbed
scHiCTools [38]	✓		~ 45 min	~ 2 hrs	Python	github.com/liu-bioinfo-lab/scHiCTools
scHi-C Topics [37]	✓		~ 4.5 hrs	~ 7 hrs	R	github.com/khj3017/schic-topic-model
scVI-3D [42]	✓	✓	~ 2.5 hrs (on GPU)	~ 6 hrs (on GPU)	Python	github.com/yezhengSTAT/scVI-3D

lower-dimensional subspace, scHiCTools facilitates the investigation of structural heterogeneity across scHi-C contact maps. Linear convolution has been demonstrated to effectively handle dropout events in sparse matrices better than other imputation approaches. Comparative analyses have shown that InnerProduct, combined with effectively computes pairwise similarities, accurately projects the Nagano et al. dataset, preserving global pairwise distances.

2.5. Higashi

Diverging from traditional linear convolution and random walk methods, Higashi [39] integrates embedding and imputation into a deep learning-based framework. For this appraoach, a novel hypergraph representation of scHi-C data was introduced, where nodes correspond to genomic loci and cells, while hyperedges represent interactions between a cell node and two corresponding genomic bin nodes. Higashi was built on Hyper-SAGNN [58], a generic hypergraph neural network framework, to capture the higher-order topological properties of the data, learning node embeddings and predicting hyperedges. Furthermore, in Higashi, global structural information is shared among cells in close proximity in the embedding space, as determined by their k-nearest neighbors. This approach leverages latent correlations between cell embeddings to improve the accuracy of imputation. For imputation, Higashi constructs a cell-dependent graph that integrates the Hi-C contact maps of the target cell and its k-nearest neighbors. The graph, along with the attributes of the genomic bin nodes, serves as inputs for the trained hypergraph neural network, which imputes missing edges while maintaining the unique features of each cell. Additionally, Higashi developed analysis methods for computing compartment scores and detecting TAD-like domain boundaries of imputed single-cell contact maps, enhancing the analysis of 3D genome structures at single-cell resolution.

2.6. scHiCExplorer

While scHiCluster provides tools for smoothing and clustering scHi-C data, it lacks a comprehensive toolbox for the entire analysis workflow, from raw data processing to cell clustering, matrix construction, and quality control. Additionally, previous methods' requirements to store contact matrices in text files can be space-consuming and complicate data sharing. In contrast, scHiCExplorer [40] addresses these challenges by offering a comprehensive software suite that supports the analysis of scHi-C data from raw FASTQ files to the final results desired by researchers. Specifically, scHiCExplorer includes functionalities for demultiplexing sequencing data by barcodes and mapping sequencing reads for individual cells. Similar to scHiCTools, scHiCExplorer also provides an option to generate quality control reports.

For embedding purposes, scHiCExplorer converts each single-cell contact matrix into a vector format and concatenates these vectors into a cell-by-bin-pair contact matrix. To overcome the curse of dimensionality, it computes similarity using a k-nearest neighbors (kNNs) graph based on the Jaccard index approximated by MinHash [59] before applying PCA to derive the latent embeddings. It has been claimed that the Jaccard index is particularly suitable for distinguishing contacts from non-contacts, compared to typical Euclidean distance, as it focuses on the features shared by cells. In this approach, scHiCExplorer calculates the similarity between two cells by tallying collisions across all MinHash functions, where each non-zero interaction is assigned a hash value. Cells that share more common features are considered more similar, while those with fewer shared features are considered less similar. Additionally, scHiCExplorer offers an option to apply KR normalization [55] to account for coverage bias. By employing the kNNs graph, scHiCExplorer achieves efficient runtime and memory utilization, making it a robust tool for scHi-C data analysis.

2.7. scHiCEmbed

Previous methods have demonstrated their effectiveness in smoothing scHi-C matrices, leading to improved cell type clustering compared to raw scHi-C matrices. ScHiCEmbed [41] further improves this aspect by employing an unsupervised approach to enhance contacts in scHi-C matrices, using bin-specific embedding on graph-structured data. Specifically, scHiCEmbed can take either raw or imputed scHi-C contact maps (e.g., imputed maps from scHiCluster) as an adjacency matrix and reconstruct the contact maps by performing bin-specific embedding using a graph auto-encoder. In this process, the encoder is designed to embed each bin into a higher-dimensional space while the decoder reconstructs the input scHi-C matrix using the bin-specific embeddings. After obtaining reconstructed contact maps, scHiCEmbed concatenates the reduced contact maps of all chromosomes and performs an additional round of dimensionality reduction via PCA.

Notably, the bin-specific embedding matrix learned by the encoder for each scHi-C map can be further used to reconstruct 3D genome structures and detect TADs. The optimal bin-by-3 embedding matrix learned by scHiCEmbed represents the 3D coordinates of the reconstructed single-cell structures. It has been demonstrated that chromatin can continue to expand in 3D space during the interphase state by using these reconstructed 3D structures from scHiCEmbed. Furthermore, this bin-specific embedding matrix can be used to generate a dissimilarity matrix, enabling the identification of TADs through constrained hierarchical clustering.

2.8. BandNorm

Zheng et al. [42] introduced BandNorm to tackle key challenges in scHi-C analysis, such as genomic distance bias, batch effect, and variability in sequencing depth. BandNorm specifically addresses the genomic distance bias associated with band effects [6] and normalizes sequencing depth between cells to enhance data quality for downstream analyses.

In Hi-C matrices, diagonals and off-diagonals are referred to as bands. The contact frequencies on the same band are expected to be

uniform across the dataset, as these contacts involve loci with similar genomic distances. The band effect indicates that with closer proximity to the diagonal of the Hi-C matrix, locus pairs generally display higher contact frequencies. BandNorm normalizes scHi-C matrices based on these principles. Given the symmetry of scHi-C matrices, BandNorm processes only the upper triangular part of given matrices. It constructs a band matrix by aggregating the bands from all cells and then normalizes contact frequencies of each band by dividing them by the band mean of the corresponding cell. Each band is then scaled by the average band mean across all cells.

BandNorm provides a fast and effective normalization approach that addresses band effects and sequencing depth variability. It also incorporates Harmony to remove batch effects in the latent embeddings, thereby improving the ability to distinguish between various cell subtypes and facilitating subsequent cell-subtype-specific analysis. Compared to methods like Higashi, scHiCluster, and scVI-3D, BandNorm excels in detecting TAD-like structures, showing the highest accuracy among these methods [42]. However, it does not address the issue of sparsity in scHi-C data.

2.9. scVI-3D

Alongside BandNorm, Zheng et al. [42] introduced scVI-3D, a deep generative model designed to effectively handle sparse band matrices. scVI-3D uses a zero-inflated negative binomial distribution to model the input band matrices and utilizes a denoising variational autoencoder (VAE) framework to address issues related to library sizes and batch effects. The software leverages the VAE implementation from the scvi-tools library [60]. Notably, scVI-3D explores various pooling strategies that concatenate several band matrices from different chromosomes. This pooling approach aims to enhance the robustness of cell embeddings and improve clustering performance, although results can vary depending on the pooling strategy used.

scVI-3D is robust and excels in clustering and preserving chromatin structures, such as TADs and A/B compartments. It demonstrates high recovery rates for TAD-like boundaries and maintains high consistency in bulk data. Additionally, the processed contact maps from scVI-3D can facilitate the recovery of cell-type relationships and the identification of significant interactions. However, similar to Higashi, scVI-3D is computationally demanding. A distinct VAE is trained for each band matrix and each chromosome, resulting in a large number of deep neural network models to train. Furthermore, scVI-3D assumes spatial independence between neighboring locus pairs, even though they are often correlated in reality.

2.10. Fast-Higashi

To improve scalability and model interpretability, the authors of Higashi introduced Fast-Higashi [43]. Unlike the deep learning approach used in Higashi, Fast-Higashi employs a tensor decomposition-based method to accelerate computations. Inspired by the concept of metagenes in scRNA-seq analysis, Fast-Higashi introduces “meta-interactions” to enhance interoperability in single-cell 3D genome analyses.

Fast-Higashi implements a random walk-based strategy to address data sparsity, similar to scHiCluster, but with increased efficiency. Rather than performing RWR on the entire matrix before tensor decomposition, it integrates these steps and conducts RWR in batches. The model applies the core-PARAFAC2 tensor decomposition model [61] to decompose the tensor representation of scHi-C data into four components: meta-interactions, a weight matrix, a cell embedding matrix, and a transformation matrix.

It has been demonstrated that meta-interactions can effectively capture cell-type-specific 3D chromatin features in both simulated datasets and complex tissues. By integrating meta-interactions with cell embeddings, Fast-Higashi offers a novel approach to studying differential 3D chromatin structures across various cell types. Additionally, meta-interactions are promising for multi-omics data integration. Fast-Higashi is significantly faster than scVI-3D and Higashi, respectively, while also achieving state-of-the-art cell clustering results.

3. Performance evaluation of scHi-C embedding methods

In this section, we compared the performances of the aforementioned scHi-C embedding methods, with a primary focus on their effectiveness in supporting downstream clustering analysis. We evaluated the following eight methods: BandNorm, Fast-Higashi, Higashi, scHiCExplorer, scHiCluster, scHiCTools, scHi-C Topics, and scVI-3D. scHiCEmbed was excluded from this analysis due to its focus on 3D structure reconstruction rather than clustering.

For this analysis, we used two scHi-C datasets, both at a 500-kb resolution: (1) a mouse cell-cycle dataset from Nagano et al. [20], comprising 1171 cells with approximately 350 million total sequencing reads, including 320 million intra-chromosomal reads; and (2) a developing mouse brain dataset from Tan et al. [25], comprising 1954 cells with approximately 780 million total sequencing reads, including 620 million intra-chromosomal reads. The performance of each method was evaluated using the Adjusted Rand Index (ARI) and Normalized Mutual Information (NMI) scores, which were computed based on the clustering results obtained from the Kmeans++ algorithm applied to the final 2D embeddings produced by each method. ARI measures the similarity between the predicted clustering and the ground truth, where a score of 1 indicates perfect clustering and a score of 0 indicates random clustering. NMI assesses clustering quality by measuring the mutual information between the predicted clusters and the ground truth, normalized to yield a score between 0 and 1, with 1 indicating perfect alignment.

For the Nagano et al. dataset, we were particularly interested in assessing whether a low-dimensional embedding could capture the circular dynamics of the cell cycle. Fig. 2 shows that BandNorm, FastHigashi, Higashi, and scHiCTools clearly presented the circular cell-cycle pattern. In contrast, scHiCluster, scVI-3D, and scHi-C Topics depicted the cell-cycle trajectory, but without the distinct circular structure. However, scHiCExplorer failed to exhibit a discernible cell-cycle manifold.

Next, we evaluated whether the embeddings can effectively differentiate the four cell-cycle stages. Using both ARI and NMI scores, we assessed the clustering performance for these stages. Fig. 4 demonstrates that scHiCExplorer performed poorly in terms of ARI and NMI scores, while the remaining eight methods performed well and yielded comparable results. Notably, scVI-3D and scHiCluster emerge as the top two methods in terms of clustering the Nagano et al. dataset, followed by FastHigashi, scHiCTools, and Higashi.

In addition to using the Nagano et al. dataset, which featured a pronounced cell-cycle pattern, we also analyzed the Tan et al. dataset, consisting of 13 cell types from the developing mouse brain. Due to the presence of numerous neuron subtypes, this dataset presented greater challenges for clustering, resulting in generally lower scores than those observed with the Nagano et al. dataset. Fig. 3 shows that all methods, except for scHiCExplorer, achieved effective cell-type separation. It is important to note that some cells lacking cell-type annotations and are labeled as “Unknown,” and these cells were excluded from the clustering performance evaluation. Among the tested methods, BandNorm, FastHigashi, Higashi, scHiCTools, and scVI-3D delivered particularly competitive clustering results (Fig. 4).

Lastly, we recorded the runtime for each method to assess and compare their computational efficiency (Table 2). Three methods—BandNorm, FastHigashi, and scHiCExplorer—demonstrated significantly faster runtime than the other methods across both datasets. Among these, FastHigashi delivered the fastest performance when assisted with GPUs, while BandNorm and scHiCExplorer provided competitive performance with only CPUs used.

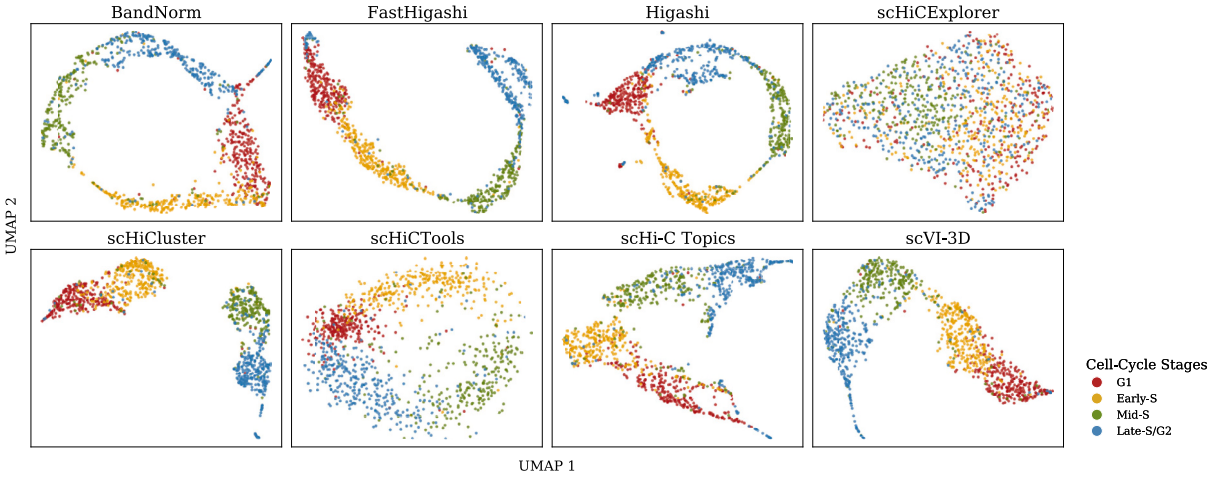

Fig. 2. Visualization and clustering of Nagano et al. dataset. This set of scatterplots provides 2D visualizations of the embeddings from the Nagano et al. dataset, obtained using UMAP with two components. Each dot represents an individual cell, with different colors indicating four cell-cycle stages.

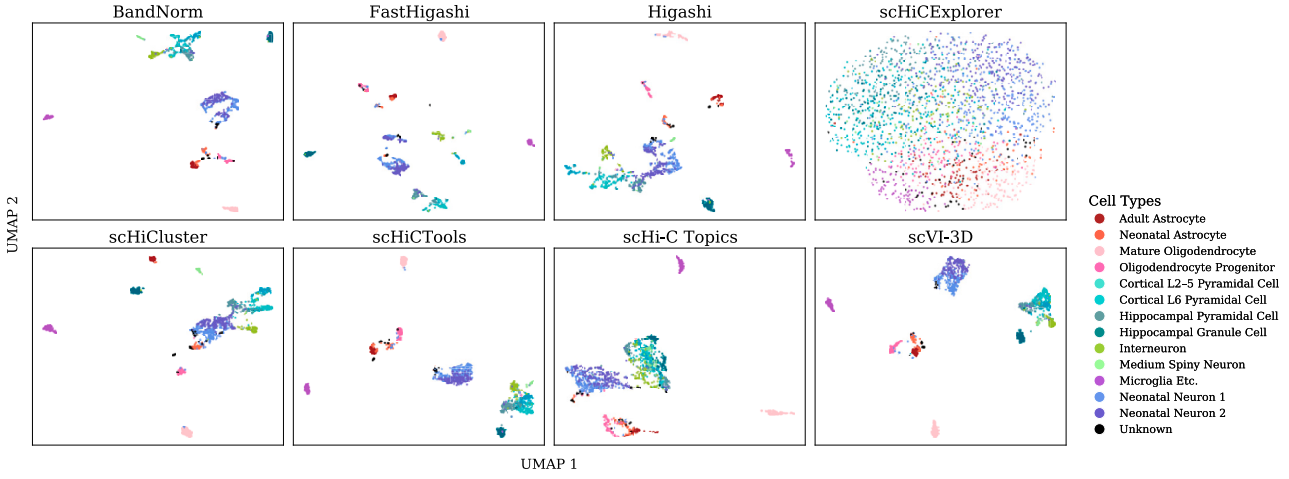

Fig. 3. Visualization and clustering of Tan et al. dataset. This set of scatterplots provides 2D visualizations of the embeddings from the Tan et al. dataset, obtained using UMAP with two components. Each dot represents an individual cell, with different colors indicating 13 cell subtypes.

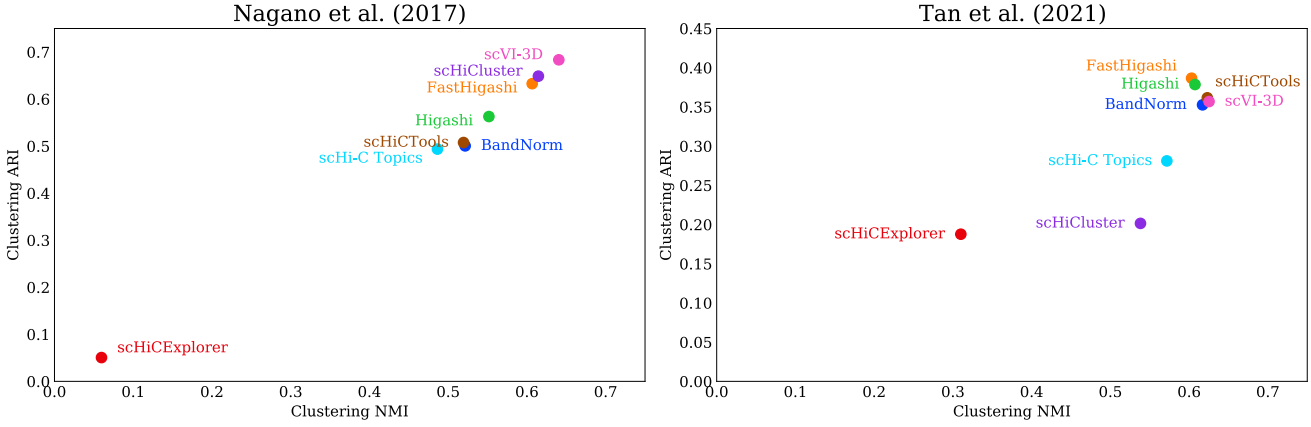

Fig. 4. Clustering performances of scHi-C embedding methods. The clustering scores were derived from the 2D UMAP embeddings of (A) the mouse cell-cycle dataset (Nagano et al.) and (B) the mouse developmental brain dataset (Tan et al.). The x-axis represents NMI scores and the y-axis represents ARI scores. Each point represents the results of a scHi-C embedding method, with different colors and labels indicating the specific method used.

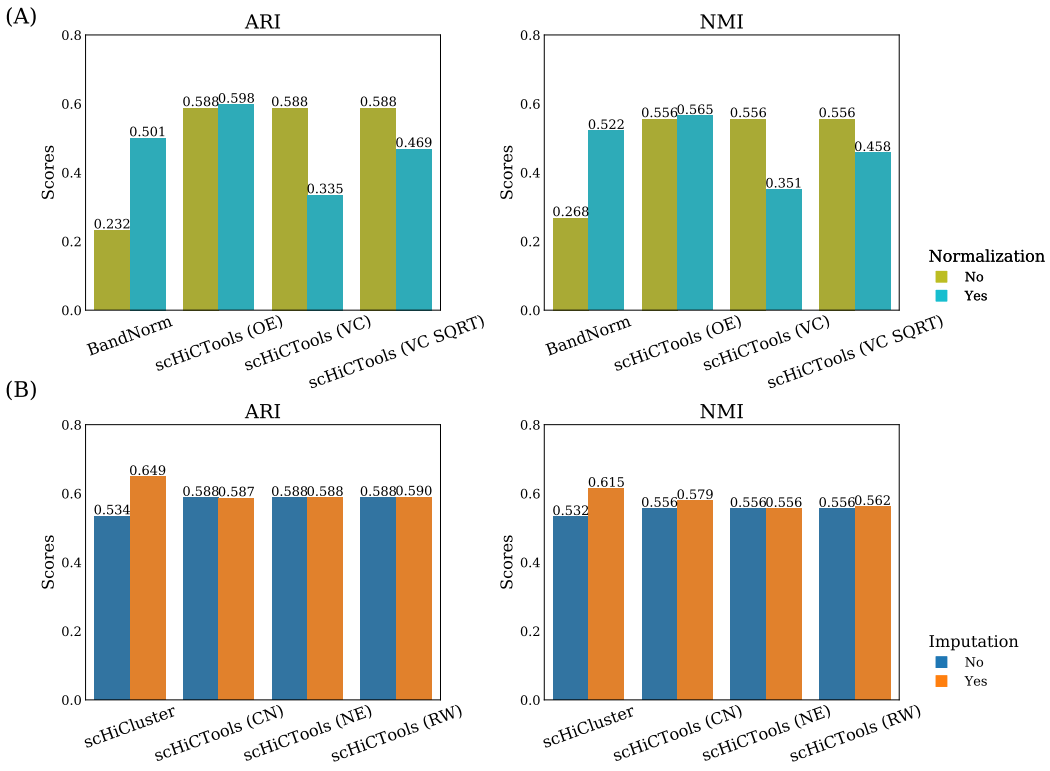

Fig. 5. Clustering performances with and without normalization/imputation. (A) Comparison of the clustering results with and without normalization. (B) Comparison of the clustering results with and without imputation. Each panel includes two barplots (left: ARI; right: NMI), displaying clustering scores based on 2D embeddings derived from the following methods: BandNorm, scHiCluster, and scHiCTools. Note that scHiCTools offers three normalization options: observed/expected (OE), Vanilla coverage (VC), and Knight-Ruiz (KR), as well as three imputation options: linear convolution (CN), random walk (RW), and network enhancing (NE). For each method, scores for non-normalized/imputed and normalized/imputed data are shown side-by-side. Rounded scores are annotated above each bar for clarity.

4. Effects of normalization and imputation on scHi-C embedding and clustering

Normalization and imputation play crucial roles in improving the quality and interpretability of scHi-C data. Due to the inherent sparsity of scHi-C contact matrices and heterogeneous sequencing depths across different cells, these pre-processing steps are essential for accurate downstream analyses. Normalization techniques aim to account for coverage biases and variability in library sizes across cells and experiments, resulting in more balanced contact matrices. This process helps ensure that contact frequencies are comparable within and across different datasets. In Section 2, we reviewed various embedding methods that incorporate different normalization techniques during pre-processing. Some methods use standard normalization techniques. For example, scVI-3D normalizes scHi-C contacts per million within each cell, followed by log transformation, while Higashi and FastHigashi normalize contacts by the total read count (i.e., coverage). Other methods adapt normalization techniques from bulk Hi-C analysis. For example, scHiCTools provides three normalization options—OE normalization, VC normalization, and KR normalization—while scHiCExplorer only uses KR normalization. Notably, BandNorm employs a scHi-C-specific band normalization approach to address unique challenges in scHi-C data.

Imputation methods, on the other hand, are designed to address sparsity in scHi-C data by recovering low-frequency or missed contacts in scHi-C matrices. The aforementioned scHi-C embedding methods employ various imputation strategies, including linear convolution (used by scHiCluster and scHiCTools), random walk (used by scHiCluster, scHiCTools, and Fast-Higashi), network enhancing (used by scHiCTools), and scHi-C-specific neural networks (used by Higashi and scVI-3D). While these imputation methods can significantly enhance downstream embedding and clustering performance, over-imputation may result in over-smoothed contact matrices, potentially obscuring important structural features, such as chromatin loops.

Our evaluation results in Section 3 showed that the embedding methods incorporating normalization and/or imputation, such as BandNorm, FastHigashi, and Higashi, demonstrated improved clustering performance and more robust embeddings compared to the methods that do not include these pre-processing steps (e.g., scHi-C Topics). However, the extent of this improvement depends on the dataset, the specific characteristics of the embedding method, and the choice of normalization and imputation techniques.

Among these scHi-C embedding tools, three allow users to choose whether to include normalization or imputation in pre-processing. BandNorm offers a band normalization option; scHiCluster incorporates both linear convolution and random work imputation; and scHiCTools provides a comprehensive list of normalization options (OE, VC, VC SQRT) and imputation options (linear convolution, random work, and network enhancing).

To further illustrate the effects of normalization and imputation, we applied these three methods, BandNorm, scHiCluster, and scHiCTools, to the Nagano et al. dataset and compared their clustering performances with and without the normalization/imputation steps. As shown in Fig. 5A, the band normalization strategy in BandNorm significantly improved clustering performance. On the other hand, in scHiCTools, only the OE-normalized data produced competitive results compared to the raw data. The other bulk Hi-C normalization techniques (VC and VC SQRT) surprisingly yield worse performances.

As for imputation, the strategies used in scHiCluster (linear convolution and random work) notably enhanced the clustering performance. However, the three imputation approaches integrated into scHiCTools yielded only marginal improvements. This can be attributed to the quality of the Nagano et al. dataset, which was already being sufficient for the “innerproduct” similarity approach in scHiCTools. This observation

aligns with the findings in the scHiCTools paper, where the authors noted that the "innerproduct" approach is robust across various downsampling and dropout levels [38].

5. Discussion

ScHi-C techniques have been widely utilized to study 3D genome organization, uncovering the spatial and dynamic patterns within the cell nuclei. Embedding methods have emerged as powerful tools for biologists to cluster and annotate scHi-C data, promoting the investigation of cell-type-specific characteristics. However, due to the technical limitations of scHi-C techniques, the data are often extremely sparse, posing challenges in revealing genome architecture. Additionally, unlike 1D genomics data, scHi-C data are in a 2D format with a complex, hierarchical structure presented in contact maps, which further complicates the data analyses and requires substantial computational resources and time. To facilitate cell clustering and other downstream analyses, various methods have been developed to handle the embedding of scHi-C data and address these challenges. These methods employ diverse strategies to reduce the high dimensionality of scHi-C data, including transformation, decomposition, neural networks, and graph-based approaches. In addition, they often incorporate pre-processing techniques such as contact matrix normalization, contact imputation, and batch effect removal, which greatly help extract important and meaningful features from the scHi-C data.

Given the large-scale nature of scHi-C datasets, many methods prioritize computational efficiency and memory usage over previously published methods. For researchers seeking fast embedding results, FastHigashi and BandNorm are particularly recommended. Our comprehensive benchmarking demonstrated that BandNorm and FastHigashi excel in time efficiency while achieving competitive clustering performance. Both methods complete their tasks within half an hour for the two selected datasets (Table 2). Moreover, Fast-Higashi was notably faster than other deep learning-based methods on GPU implementation [43], and our runtime analysis has shown its impressive time efficiency, achieving results for approximately 2000 cells at 500-kb resolution in just 10 minutes. For CPU implementations, BandNorm can process a large-scale dataset of over 4000 cells at 1-Mb resolution in under 15 minutes on a single-core CPU [42]. Furthermore, our evaluation further demonstrated that BandNorm completes a dataset of 2000 cells at 500-kb resolution in less than half an hour; in contrast, scHiCluster requires at least 2 hours on a 23-core CPU, and scHi-C Topics needs at least 7 hours on a single-core CPU.

Despite the advancements, current scHi-C embedding methods still have limitations. Normalization strategies, although effective in addressing coverage discrepancies, may introduce additional biases. Imputation techniques may lead to over-smoothing that affects the extraction of structural features due to inaccurate estimation of chromatin interactions [17]. For example, Zheng et al. [42] demonstrated that Higashi and scHiCluster face issues of over-smoothing and blurriness, which obscure chromatin structures, compared to scVI-3D and BandNorm. Furthermore, while current models excel at distinguishing between cell subtypes, they fall short of differentiating rare cell populations at a finer scale. Therefore, the development of more advanced tools for processing and analyzing scHi-C data is critical to address these limitations.

By summarizing current embedding methods for scHi-C data, we aim to make this review a valuable resource for researchers studying the 3D genome architecture as well as those developing new embedding techniques. Our review provides a comprehensive overview of existing embedding methods, detailing their underlying strategies, strengths, and limitations, as well as their ability to address challenges such as data sparsity, high dimensionality, and the complex hierarchical structures inherent in scHi-C contact maps. We hope to assist researchers in selecting the most suitable techniques for their specific needs, thereby fostering further advancements in the study of 3D genome architecture.

CRediT authorship contribution statement

Rui Ma: Writing – review & editing, Writing – original draft, Visualization, Investigation, Formal analysis, Conceptualization. **Jingong Huang:** Writing – original draft, Investigation, Formal analysis. **Tao Jiang:** Writing – review & editing, Supervision, Funding acquisition. **Wenxiu Ma:** Writing – review & editing, Supervision, Funding acquisition, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work was supported by the National Institute of Health (R35GM133678, R01NS125018) and the National Science Foundation (DBI-1751317).

Appendix A. Supplementary material

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.csbj.2024.11.002.

References

[1] Cremer T, Cremer C. Chromosome territories, nuclear architecture and gene regulation in mammalian cells. Nat Rev Genet 2001;2(4):292–301.
[2] Kumaran RI, Thakar R, Spector DL. Chromatin dynamics and gene positioning. Cell 2008;132(6):929–34.
[3] Dekker J, Marti-Renom MA, Mirny LA. Exploring the three-dimensional organization of genomes: interpreting chromatin interaction data. Nat Rev Genet 2013;14(6):390–403.
[4] Bonev B, Cavalli G. Organization and function of the 3D genome. Nat Rev Genet 2016;17(11):661–78.
[5] Misteli T. The self-organizing genome: principles of genome architecture and function. Cell 2020;183(1):28–45.
[6] Dekker J, Rippe K, Dekker M, Kleckner N. Capturing chromosome conformation. Science 2002;295:1306–11. https://doi.org/10.1126/science.1067799.
[7] Simonis M, Klous P, Splinter E, Moshkin Y, Willemsen R, De Wit E, et al. Nuclear organization of active and inactive chromatin domains uncovered by chromosome conformation capture–on-chip (4C). Nat Genet 2006;38(11):1348–54.
[8] Dostie J, Richmond TA, Arnaout RA, Selzer RR, Lee WL, Honan TA, et al. Chromosome conformation capture carbon copy (5C): a massively parallel solution for mapping interactions between genomic elements. Genome Res 2006;16(10):1299–309.
[9] Lieberman-Aiden E, Berkum N, Williams L, Imakaev M, Ragoczy T, Telling A, et al. Comprehensive mapping of long-range interactions reveals folding principles of the human genome. Science 2009;326:289–93. https://doi.org/10.1126/science.1181369.
[10] Rao SS, Huntley MH, Durand NC, Stamenova EK, Bochkov ID, Robinson JT, et al. A 3D map of the human genome at kilobase resolution reveals principles of chromatin looping. Cell 2014;159(7):1665–80.
[11] Hsieh T-HS, Cattoglio C, Slobodyanyuk E, Hansen AS, Rando OJ, Tjian R, et al. Resolving the 3D landscape of transcription-linked mammalian chromatin folding. Mol Cell 2020;78(3):539–53.
[12] Krietenstein N, Abraham S, Venev SV, Abdennur N, Gibcus J, Hsieh T-HS, et al. Ultrastructural details of mammalian chromosome architecture. Mol Cell 2020;78(3):554–65.
[13] Fullwood MJ, Liu MH, Pan YF, Liu J, Xu H, Mohamed YB, et al. An oestrogen-receptor-α-bound human chromatin interactome. Nature 2009;462(7269):58–64.
[14] Tang WW, Dietmann S, Irie N, Leitch HG, Floros VI, Bradshaw CR, et al. A unique gene regulatory network resets the human germline epigenome for development. Cell 2015;161(6):1453–67.
[15] Mumbach MR, Rubin AJ, Flynn RA, Dai C, Khavari PA, Greenleaf WJ, et al. HiChIP: efficient and sensitive analysis of protein-directed genome architecture. Nat Methods 2016;13(11):919–22.
[16] Kempfer R, Pombo A. Methods for mapping 3D chromosome architecture. Nat Rev Genet 2020;21(4):207–26.
[17] Zhou T, Zhang R, Ma J. The 3D genome structure of single cells. Annu Rev Biomed Data Sci 2021;4:21–41.
[18] Finn EH, Pegoraro G, Brandão HB, Valton A-L, Oomen ME, Dekker J, et al. Extensive heterogeneity and intrinsic variation in spatial genome organization. Cell 2019;176(6):1502–15.

[19] Nagano T, Lubling Y, Stevens TJ, Schoenfelder S, Yaffe E, Dean W, et al. Single-cell hi-c reveals cell-to-cell variability in chromosome structure. Nature 2013;502(7469):59–64.
[20] Nagano T, Lubling Y, Várnai C, Dudley C, Leung W, Baran Y, et al. Cell-cycle dynamics of chromosomal organization at single-cell resolution. Nature 2017;547(7661):61–7.
[21] Flyamer IM, Gassler J, Imakaev M, Brandão HB, Ulianov SV, Abdennur N, et al. Single-nucleus Hi-C reveals unique chromatin reorganization at oocyte-to-zygote transition. Nature 2017;544(7648):110–4.
[22] Ramani V, Deng X, Qiu R, Gunderson KL, Steemers FJ, Disteche CM, et al. Massively multiplex single-cell Hi-C. Nat Methods 2017;14(3):263–6.
[23] Stevens TJ, Lando D, Basu S, Atkinson LP, Cao Y, Lee SF, et al. 3D structures of individual mammalian genomes studied by single-cell Hi-C. Nature 2017;544(7648):59–64.
[24] Tan L, Xing D, Chang C-H, Li H, Xie XS. Three-dimensional genome structures of single diploid human cells. Science 2018;361(6405):924–8.
[25] Tan L, Ma W, Wu H, Zheng Y, Xing D, Chen R, et al. Changes in genome architecture and transcriptional dynamics progress independently of sensory experience during post-natal brain development. Cell 2021;184(3):741–58.
[26] Lee D-S, Luo C, Zhou J, Chandran S, Rivkin A, Bartlett A, et al. Simultaneous profiling of 3d genome structure and dna methylation in single human cells. Nat Methods 2019;16(10):999–1006.
[27] Li G, Liu Y, Zhang Y, Kubo N, Yu M, Fang R, et al. Joint profiling of DNA methylation and chromatin architecture in single cells. Nat Methods 2019;16(10):991–3.
[28] Heffel MG, Zhou J, Zhang Y, Lee D-S, Hou K, Alonso OP, et al. Epigenomic and chromosomal architectural reconfiguration in developing human frontal cortex and hippocampus. bioRxiv 2022.
[29] Dekker J, Belmont AS, Guttman M, Leshyk VO, Lis JT, Lomvardas S, et al. The 4D nucleome project. Nature 2017;549(7671):219–26.
[30] Nguyen HQ, Chattoraj S, Castillo D, Nguyen SC, Nir G, Lioutas A, et al. 3D mapping and accelerated super-resolution imaging of the human genome using in situ sequencing. Nat Methods 2020;17(8):822–32.
[31] Takei Y, Yun J, Zheng S, Ollikainen N, Pierson N, White J, et al. Integrated spatial genomics reveals global architecture of single nuclei. Nature 2021;590(7845):344–50.
[32] Arrastia MV, Jachowicz JW, Ollikainen N, Curtis MS, Lai C, Quinodoz SA, et al. Single-cell measurement of higher-order 3D genome organization with scsprite. Nat Biotechnol 2022;40(1):64–73.
[33] Liu Z, Chen Y, Xia Q, Liu M, Xu H, Chi Y, et al. Linking genome structures to functions by simultaneous single-cell Hi-C and RNA-seq. Science 2023;380(6649):1070–6.
[34] Zhou T, Zhang R, Jia D, Doty RT, Munday AD, Gao D, et al. GAGE-seq concurrently profiles multiscale 3D genome organization and gene expression in single cells. Nat Genet 2024:1–11.
[35] Liu J, Lin D, Yardımcı GG, Noble WS. Unsupervised embedding of single-cell hi-c data. Bioinformatics 2018;34(13):i96–104.
[36] Zhou J, Ma J, Chen Y, Cheng C, Bao B, Peng J, et al. Robust single-cell Hi-C clustering by convolution-and random-walk–based imputation. Proc Natl Acad Sci 2019;116(28):14011–8.
[37] Kim H-J, Yardımcı GG, Bonora G, Ramani V, Liu J, Qiu R, et al. Capturing cell type-specific chromatin compartment patterns by applying topic modeling to single-cell Hi-C data. PLoS Comput Biol 2020;16(9):e1008173.
[38] Li X, Feng F, Pu H, Leung WY, Liu J. scHiCTools: a computational toolbox for analyzing single-cell Hi-C data. PLoS Comput Biol 2021;17(5):e1008978.
[39] Zhang R, Zhou T, Ma J. Multiscale and integrative single-cell Hi-C analysis with Higashi. Nat Biotechnol 2022;40(2):254–61.
[40] Wolff J, Backofen R, Grüning B. Robust and efficient single-cell Hi-C clustering with approximate k-nearest neighbor graphs. Bioinformatics 2021;37(22):4006–13.
[41] Liu T, Wang Z. scHiCEmbed: bin-specific embeddings of single-cell Hi-C data using graph auto-encoders. Genes 2022;13(6):1048.
[42] Zheng Y, Shen S, Keleş S. Normalization and de-noising of single-cell Hi-C data with BandNorm and scVI-3D. Genome Biol 2022;23(1):222.
[43] Zhang R, Zhou T, Ma J. Ultrafast and interpretable single-cell 3D genome analysis with Fast-Higashi. Cell Syst 2022;13(10):798–807.
[44] Zhang Y, Boninsegna L, Yang M, Misteli T, Alber F, Ma J. Computational methods for analysing multiscale 3D genome organization. Nat Rev Genet 2024;25(2):123–41.
[45] Liu T, Wang Z. scHiCNorm: a software package to eliminate systematic biases in single-cell Hi-C data. Bioinformatics 2018;34(6):1046–7.
[46] McInnes L, Healy J, Saul N, Großberger L. UMAP: uniform manifold approximation and projection. J Open Sour Softw 2018;3(29):861.
[47] Van der Maaten L, Hinton G. Visualizing data using t-SNE. J Mach Learn Res 2008;9(11).
[48] Yang T, Zhang F, Yardımcı GG, Song F, Hardison RC, Noble WS, et al. HiCRep: assessing the reproducibility of Hi-C data using a stratum-adjusted correlation coefficient. Genome Res 2017;27(11):1939–49.
[49] Ursu O, Boley N, Taranova M, Wang YR, Yardimci GG, Stafford Noble W, et al. GenomeDISCO: a concordance score for chromosome conformation capture experiments using random walks on contact map graphs. Bioinformatics 2018;34(16):2701–7.
[50] Yan K-K, Yardımcı GG, Yan C, Noble WS, Gerstein M. HiC-spector: a matrix library for spectral and reproducibility analysis of Hi-C contact maps. Bioinformatics 2017;33(14):2199–201.
[51] Kruskal JB. Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. Psychometrika 1964;29(1):1–27.
[52] Pan J-Y, Yang H-J, Faloutsos C, Duygulu P. Automatic multimedia cross-modal correlation discovery. In: Proceedings of the tenth ACM SIGKDD international conference on knowledge discovery and data mining; 2004. p. 653–8.
[53] Korsunsky I, Millard N, Fan J, Slowikowski K, Zhang F, Wei K, et al. Fast, sensitive and accurate integration of single-cell data with harmony. Nat Methods 2019;16(12):1289–96.
[54] González-Blas C Bravo, Minnoye L, Papasokrati D, Aibar S, Hulselmans G, Christiaens V, et al. cisTopic: cis-regulatory topic modeling on single-cell ATAC-seq data. Nat Methods 2019;16(5):397–400.
[55] Knight PA, Ruiz D. A fast algorithm for matrix balancing. IMA J Numer Anal 2013;33(3):1029–47.
[56] Wang B, Pourshafeie A, Zitnik M, Zhu J, Bustamante CD, Batzoglou S, et al. Network enhancement as a general method to denoise weighted biological networks. Nat Commun 2018;9(1):3108.
[57] Ardakany AR, Ay F, Lonardi S. Selfish: discovery of differential chromatin interactions via a self-similarity measure. Bioinformatics 2019;35(14):i145–53.
[58] Zhang R, Zou Y, Ma J. Hyper-SAGNN: a self-attention based graph neural network for hypergraphs. preprint. arXiv:1911.02613, 2019.
[59] Broder AZ. On the resemblance and containment of documents. In: Proceedings. Compression and complexity of SEQUENCES 1997 (cat. no. 97TB100171). IEEE; 1997. p. 21–9.
[60] Gayoso A, Lopez R, Xing G, Boyeau P, Pour Amiri V Valiollah, Hong J, et al. A Python library for probabilistic analysis of single-cell omics data. Nat Biotechnol 2022;40(2):163–6.
[61] Van Benthem MH, Keller TJ, Gillispie GD, DeJong SA. Getting to the core of PARAFAC2, a nonnegative approach. Chemom Intell Lab Syst 2020;206:104127.