OXFORD

Briefings in Bioinformatics, 2023, **24(1)**, 1–10
https://doi.org/10.1093/bib/bbac494
Problem Solving Protocol

Deep generative modeling and clustering of single cell Hi-C data

Qiao Liu †, Wanwen Zeng†, Wei Zhang, Sicheng Wang, Hongyang Chen, Rui Jiang, Mu Zhou and Shaoting Zhang

Corresponding authors. Rui Jiang, Ministry of Education Key Laboratory of Bioinformatics, Research Department of Bioinformatics at the Beijing National Research Center for Information Science and Technology, Center for Synthetic and Systems Biology, Department of Automation, Tsinghua University, Beijing 100084, China. Tel: +86 010-62795578; Fax: 010-62786911; E-mail: ruijiang@tsinghua.edu.cn; Mu Zhou, SenseBrain Research, San Jose, CA 95131, USA. Tel: +1 4086017085; E-mail: muzhou@sensebrain.site; Shaoting Zhang, Shanghai Artificial Intelligence Laboratory, Shanghai 200240, China. Tel: +86 02134204113; E-mail: zhangshaoting@pjlab.org.cn

†Qiao Liu and Wanwen Zeng contributed equally.

Abstract

Deciphering 3D genome conformation is important for understanding gene regulation and cellular function at a spatial level. The recent advances of single cell Hi-C technologies have enabled the profiling of the 3D architecture of DNA within individual cell, which allows us to study the cell-to-cell variability of 3D chromatin organization. Computational approaches are in urgent need to comprehensively analyze the sparse and heterogeneous single cell Hi-C data. Here, we proposed scDEC-Hi-C, a new framework for single cell Hi-C analysis with deep generative neural networks. scDEC-Hi-C outperforms existing methods in terms of single cell Hi-C data clustering and imputation. Moreover, the generative power of scDEC-Hi-C could help unveil the differences of chromatin architecture across cell types. We expect that scDEC-Hi-C could shed light on deepening our understanding of the complex mechanism underlying the formation of chromatin contacts.

Keywords: single cell, 3D genome, deep learning, unsupervised learning

Introduction

The rapid development in single-cell technologies enables us to reliably measure the genomic, transcriptomic and epigenomic features of a particular cellular context at single-cell resolution [1–4]. These powerful technologies provide scientists with the opportunity to study the unique patterns of cell type specificity and gene regulation. One fundamental question regarding the abundant single cell data is how to distinguish different cell types in a heterogeneous cell population based on the measured molecular signatures. A variety of computational approaches have been developed to decipher the heterogeneity across cell types based on transcriptome, methylome and chromatin accessibility [5–11].

The majority of the current single-cell assays, such as RNA sequencing (scRNA-seq) and transposase-accessible chromatin using sequencing (scATAC-seq), ignore the spatial information of the genome, such as 3D chromatin structure, which plays an important role in genome functions, including gene transcription and DNA replication [12–14]. The emerging single cell Hi-C technologies bridge this gap by measuring the 3D chromatin structures in individual cells, which have the potential to comprehensively reveal the diverse genome functions underlying the unique genome structure [15–19].

Several computational methods have been proposed for the single cell Hi-C data analysis. For example, Kim *et al*. [20] used a latent Dirichlet allocation (LDA) topic model for discovering the latent topics given the scHi-C data. scHiCluster [21] introduced a random walk-based strategy for data imputation and used PCA for embedding. HiCRep/MDS [22] used multi-dimensional scaling (MDS) for learning a low-dimensional embedding. Higashi [23] is a recent method that utilized hypergraph representation learning for single cell imputation and embedding. However, all these methods require an additional clustering approach (e.g. *K*-means) for identifying cell types. In addition, choosing the most appropriate clustering approach is sometimes difficult as it is hard for a single clustering approach to perform the best across different datasets.

To overcome the aforementioned limitations, we developed scDEC-Hi-C, a comprehensive end-to-end unsupervised learning framework for single cell Hi-C data embedding, clustering and generation by deep generative neural networks. Unlike existing

Qiao Liu is a postdoctoral researcher in the Department of Statistics at Stanford University. His research interest focuses on machine learning, statistics, and computational biology.
Wanwen Zeng is a postdoctoral researcher in the Department of Statistics at Stanford University. Her research interest focuses on machine learning and bioinformatics.
Wei Zhang is a postdoctoral researcher in the Department of Biomedical Engineering at Shandong University. His research interest focuses on bioinformatics.
Sicheng Wang is a Master student in the Department of Computer Science and Engineering at UCSD. His research interest focuses on machine learning and bioinformatics.
Hongyang Chen is a Reseacher at Zhejiang Lab. His interest focuses on machine learning and communication.
Rui Jiang is an Associate Professor in the Department of Automation at Tsinghua University. His research interest focuses on bioinformatics.
Mu Zhou is a Visiting Professor in the Department of Computer Science at Rutgers University. His research interest focuses on medical image analysis and drug discovery.
Shaoting Zhang is affiliated with Shanghai Artificial Intelligence Laboratory. His research interest focuses on computer vision, deep learning, and medical AI applications.
Received: July 20, 2022. **Revised:** September 28, 2022. **Accepted:** October 18, 2022

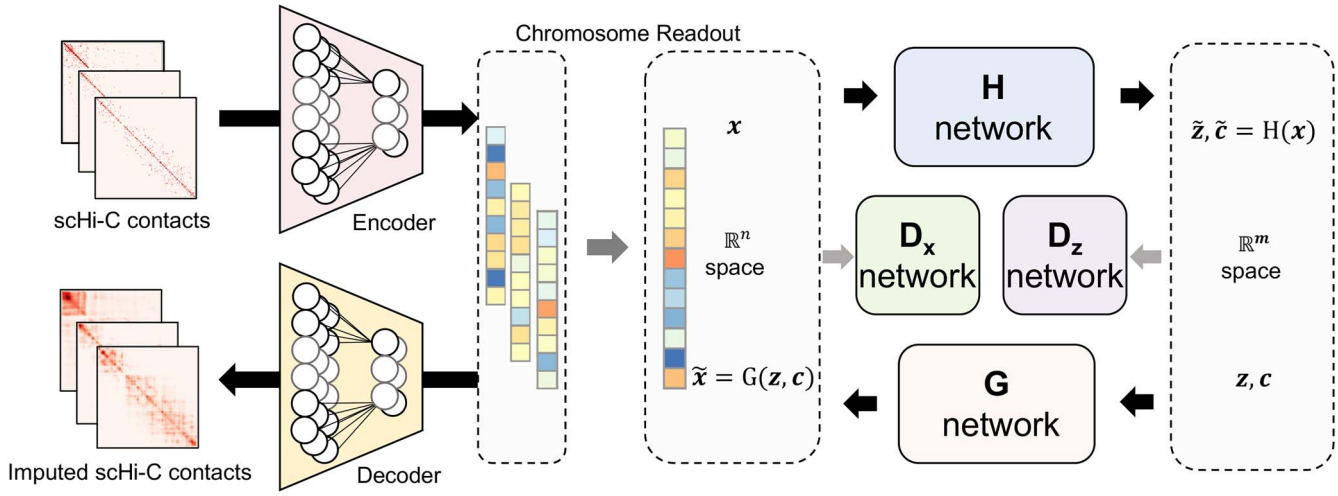

Figure 1. The overview of the proposed scDEC-Hi-C model. scDEC-Hi-C is a multi-scale model, which contains a chromosome-wise convolutional autoencoder (CAE) and a cell-wise single cell deep embedding and clustering model. The intra-chromosome single-cell Hi-C contacts matrices are first fed to a CAE for dimension reduction and latent feature extraction. Then the chromosome readout (e.g. concatenation) is applied to get the cell-wise representation. The cell-wise deep generative neural networks can further learn a low dimensional representation of a cell and cluster each cell simultaneously. In the latent space, latent variables z and c sampled from a Gaussian distribution and a Category distribution, respectively, are fed to the G network. The H network has two outputs of which one corresponds to the latent embedding ($\tilde{z}$) and one corresponds to the estimated cluster label ($\tilde{c}$). The D_x and D_z discriminator networks are used for adversarial training.

methods that treat embedding and clustering as two separated tasks, our approach enables simultaneously learning the low-dimensional embeddings of single cell Hi-C data and clustering the single cell Hi-C data by neural network in an unsupervised manner. From systematical experiments, scDEC-Hi-C demonstrates superiority in various tasks, including clustering the cell types, data imputation for quality enhancement, as well as data generation given a desired cell type. To the best of our knowledge, scDEC-Hi-C is the first computational framework that integrates the data embedding and clustering intrinsically for the single cell Hi-C data analysis.

Results

Overview of scDEC-Hi-C

scDEC-Hi-C consists of two major computational modules, including a convolutional autoencoder module for chromosome-wise representation learning and a deep generative module for cell-wise representation learning and clustering (Figure 1). The autoencoder module aims at extracting the low-dimensional features for each chromosome within a cell. Then the chromosome-wise features are transformed to cell-wise features through a chromosome readout function. We chose global concatenation for the readout function as default. The cell-wise generative model is adopted from our previous work scDEC [24] where G and H networks aim at bidirectional transformation between the m-dimensional latent space and n-dimensional representer space. Note that the latent variables z follows a standard Gaussian distribution $N(0, I)$ and c follows a category distribution $\mathbf{Cat}(K, w)$, which is parameterized by the number of clusters K and the weight w. G network takes z and c as inputs and D_x network was used for matching the distribution of cell-wise representation x and G network output $\tilde{x}$ through adversarial training. Similarly, H network and D_z network also work in an adversarial manner where H network could learn the latent representation ($\tilde{z}$) and infer the cluster ($\tilde{c}$) simultaneously. The detailed hyperparameters of model architecture were provided in Supplementary Table 1 available online at http://bib.oxfordjournals.org/.

scDEC-Hi-C is capable of accurately identifying cell types

A fundamental problem in single cell Hi-C data analysis is to identify different cell types in heterogeneous cell populations. To evaluate the performance of scDEC-Hi-C on this task, we adopted three commonly used benchmark datasets here and systematically compared scDEC-Hi-C to five baseline methods (see section Methods for data preprocessing and Supplementary Table 2 available online at http://bib.oxfordjournals.org/). Three metrics, including NMI, ARI and homogeneity, were introduced for measuring the performance in this unsupervised learning task in order to quantify the ability for distinguishing different cell types in the single cell Hi-C datasets (see section Methods). Note that all baseline methods are only able to learn the embedding for each single cell and require additional clustering methods (e.g. K-means) while scDEC-Hi-C simultaneously learns cell embeddings and assigns clustering labels to each cell. scDEC-Hi-C is capable of learning embeddings, which could separate cells from different cell types with a relatively larger margin than other baseline methods (Figure 2A–C and Supplementary Figure 1 available online at http://bib.oxfordjournals.org/). It is worth mentioning that scDEC-Hi-C exhibits superior performance on Ramani dataset [17] by outperforming other methods with an ARI of 0.845, compared to 0.826 of Higashi, 0.795 of scHiCluster and 0.785 of HiCRep/MDS. The same trend was observed in Nagano dataset where scDEC-Hi-C achieves the highest ARI of 0.411, compared to 0.389 of the second best. In the Dip-C dataset [25] where only annotated labels were available, we treat the annotated labels as surrogate ground truth labels. All methods demonstrate significantly lower clustering performance than the Ramani dataset with ground truth label. Specifically, scDEC-Hi-C only demonstrates slightly lower performance than Higashi, in terms of Homogeneity (Figure 2D). In the readout module in scDEC-Hi-C, the information coming from each chromosome was aggregated. Thus, it is worthy to evaluate the contribution of each chromosome. The experimental results show that chromosome 11 contributed the most in Ramani dataset and scDEC-Hi-C consistently outperformed Higashi in 18 chromosomes out of

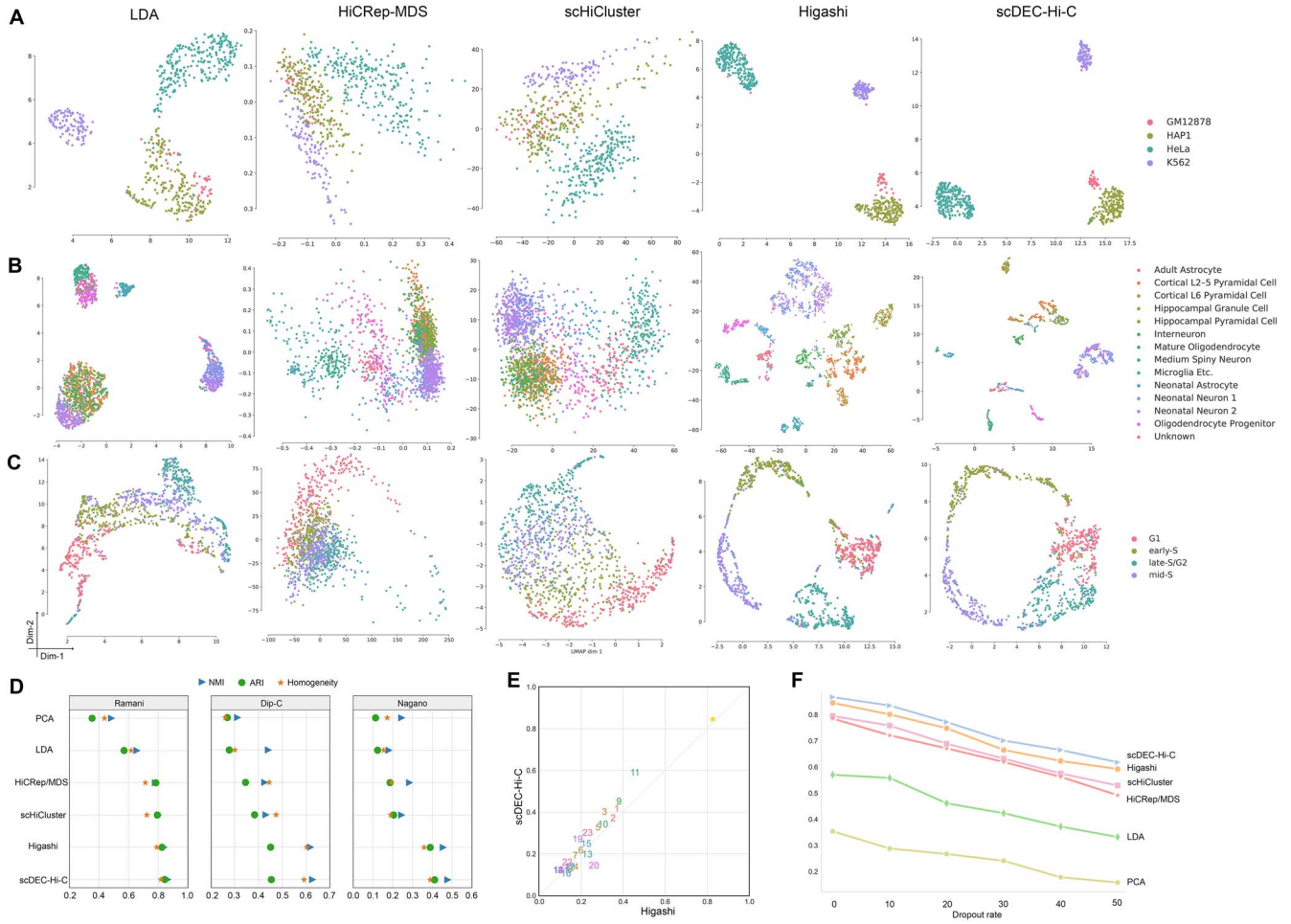

Figure 2. The performance of scDEC-Hi-C method and baseline methods on single cell Hi-C datasets. (**A**) The embeddings visualization of Ramani dataset across five methods. (**B**) The embeddings visualization of Dip-C dataset across five methods. (**C**) The embeddings visualization of Nagano dataset across five methods. (**D**) The clustering performance in terms of NMI, ARI and Homogeneity of six methods across three datasets. (**E**) The performance of scDEC-Hi-C and Higashi using a single chromosome (asterisk denotes the performance using all chromosomes). (**F**) The performance of scDEC-Hi-C and baseline methods under different dropout rates on Ramani dataset.

24 (Figure 2E and Supplementary Figure 2 available online at http://bib.oxfordjournals.org/). To further investigate the effect of sequencing depth on the clustering performance, we randomly dropout the sequencing reads with different rate for each cell. scDEC-Hi-C consistently outperforms all other baseline methods at different dropout rates (Figure 2F).

scDEC-Hi-C enables the identification of structural differences

In single cell Hi-C data analysis, one fundamental question to ask is whether cell type specificity is revealed by the structural difference regions in Hi-C contacts. The cell type specificity in single cell data, such as single cell RNA-seq and single cell ATAC-seq, can be clearly revealed by marker genes or differential peaks [26]. In bulk Hi-C data, it has been validated that cell type specificity is highly associated with the dynamic chromatin loops within topologically associating domains (TADs) [27, 28]. Therefore, it is worthwhile to investigate whether the structural differences also exist in single cell Hi-C data. To explore this, we used the autoencoder from the first stage of scDEC-Hi-C model as an approach for scDEC-Hi-C imputation. In brief, we segmented Hi-C contact matrix of each chromosome per cell into non-overlapping square patches within the range of 1 Mbp. We then treated the output of decoder as the imputed single cell Hi-C data (see section Methods). We designed extensive experiments to evaluate whether the imputed single cell Hi-C data could reveal more biological insights than the raw data. We aggregated single cells of K562 and GM12878 cell lines from Ramani dataset using 10 k as resolution and then merged them as the pseudo-bulk Hi-C data. In the meanwhile, we also downloaded the bulk Hi-C data from GM12878 and K562 cell lines as ground truth for validation. From the Hi-C profile of a genomic region (chr9: 132.9M–134.9M), K562 and GM12878 have significantly different Hi-C contact map. The difference is also emphasized by the imputed pseudo-bulk Hi-C data (Figure 3A). Specifically, the chromatin structural boundaries marked by the rectangle is much clearer by imputed pseudo-bulk Hi-C data than the pseudo-bulk Hi-C data without imputation, which demonstrates the power and effectiveness of scDEC-Hi-C in enhancing the resolution of chromatin structural boundaries. It is also noticeable that the chromatin structural boundaries revealed by bulk Hi-C data have a larger consistency with imputed pseudo-bulk Hi-C data than the pseudo-bulk Hi-C data without imputation. To verify whether these structural differences are statistically significant, we calculated the correlation between merged scHi-C data (with/without imputation) and bulk Hi-C data at different distance (up to 1 Mb). We found that imputed scHi-C data demonstrated a higher Pearson and Spearman correlation than the raw scHi-C data (Supplementary Figure 3 available online at

http://bib.oxfordjournals.org/). To further investigate the regulatory landscape of this genomic region, we downloaded both RNA-seq and histone modification data from ENCODE database [29] and visualized them with the help of WashU Epigenome Browser [30]. It can be seen that both RNA-seq signal and H3K4me1 marker are more enriched in K562 cell line than GM12878 cell line in the bounded region (Figure 3B), which indicates a strong activity of regulatory elements such as enhancer in K562. Next, we designed quantitative experiments to verify whether scDEC-Hi-C can help enhance the signal-to-noise ratio by data imputation. Taking the bulk K562 Hi-C data as ground truth, we calculated the Pearson's correlation of Hi-C interactions of different distances between ground truth and imputed data. It is seen that the interactions at a larger distance are more difficult to impute (Figure 3C). The correlation between bulk Hi-C data and raw single cell Hi-C data is less than 0.25, while the single cell Hi-C data imputed by scDEC-Hi-C, Higashi and scHiCluster are much higher than the baseline. scDEC-Hi-C consistently outperforms scHiCluster and Higashi at different distances ranging from 0 to 1 Mb. To sum up, scDEC-Hi-C enables improving the identification of structural boundaries, which further helps us study the chromatin structure difference across diverse cell types.

scDEC-Hi-C enhances the discovery of chromatin loops

Chromatin loops are defined as a pair of genomic regions that are brought into spatial proximity, which can be inferred from bulk Hi-C data. Chromatin loops have been proved to be highly relevant to gene regulation, cell fates and functions. We then intended to explore whether the chromatin loops can also be identified within single cell Hi-C data. Similarly, we merged single cell Hi-C data of K562 and GM12878 cell lines, respectively. In the meanwhile, we also downloaded the corresponding bulk Hi-C data for comparison. We applied Fit-Hi-C [31], a computational tool for calling chromatin loops from Hi-C data, to bulk Hi-C data and imputed single cell Hi-C data by scDEC-Hi-C, respectively. There are 6478 chromatin loops in GM12878 cell line while 732 (11.3%) chromatin loops are also discovered in imputed single cell Hi-C data (Figure 4A). scDEC-Hi-C additionally identified 294 chromatin loops, which are not contained in the bulk Hi-C chromatin loops. Note that only 196 chromatin loops can be identified from raw single cell Hi-C data and scDEC-Hi-C significantly improves the recall rate from 1.4% to 11.3% by imputation (Supplementary Figure 4 available online at http://bib.oxfordjournals.org/). Additionally, we also compared the precision and recall rate improvement introduced by different methods, scDEC-Hi-C also demonstrates a more powerful imputation ability than baseline methods (Supplementary Figure 5 available online at http://bib.oxfordjournals.org/). We visualized the chromatin loops in a genomic region (chr3:118.2M–120.2M) of bulk Hi-C chromatin loops versus either raw single cell Hi-C data (Figure 4B) or imputed single cell Hi-C data (Figure 4C). In K562 cell line, 12.0% of the chromatin loops from bulk Hi-C data can be also recovered by imputed single cell Hi-C data and 72.5% of the chromatin loops from imputed single cell Hi-C data are also contained in bulk chromatin loops (Figure 4D). In the same genomic region, imputed single cell Hi-C data contains three chromatin loops while two of them were consistent with bulk Hi-C chromatin loops (Figure 4F) while the raw single cell Hi-C data only has one false chromatin loop (Figure 4E). To conclude, scDEC-Hi-C is able to promote the identification of chromatin loops from Hi-C data.

scDEC-Hi-C identifies the cell-type specific TAD-like boundaries

In this section, we showcased an application of using scDEC-Hi-C in identifying the cell-type specific TAD-like boundaries in two cell types, GM12878 and K562. We now focus on the exploring the relationship between domain boundaries inferred from imputed single cell Hi-C data and the implicating functions. We extracted a 1 Mbp genomic region (chr9:36.5–37.5M), which contains Pax5, a master regulator of B cell development [32]. At the same time, we collected several annotation tracks using WashU Epigenome Browser [30]. We note that Pax5 regulator was only expressed in GM12878 cell type according to the RNA-seq annotation track. The imputed pseudo-bulk Hi-C (merged from scHi-C data) shows clear TADs or sub-TADs, while the original pseudo-bulk Hi-C without imputation fails to demonstrate clear boundaries (Figure 5). Interestingly, scDEC-Hi-C identified several promoter-enhancer interactions, which are consistent with the annotations by two histone modifications. Compared to K562 cell type, GM12878 contains more sub-domain boundaries, which are enriched in signals from two histone modifications and ChIA-PET with CTCF (a promoter P and three potential enhancers E1–E3 were denoted in GM12878). More importantly, except for the common contact domain boundaries (yellow dots), we also observed that GM12878 contained cell type specific domain boundaries (blue dots) that were not discovered in K562 cell types. Previous studies have shown that such cell type specific domain boundaries are crucial to the relevant chromatin architecture and the underlying gene regulations [33]. A great portion of the cell type specific contact domain boundaries may not be uncovered in insufficient sequenced scHi-C data. With the powerful scDEC-Hi-C model, one can significantly enhance the resolution of scHi-C data and identity more refined contact domain boundaries.

Ablation analysis

To systematically evaluate the robustness of scDEC-Hi-C, we designed the following ablation studies. We used Ramani dataset for the ablation studies. First, we removed the cell-wise scDEC module and only kept the chromosome-wise convolutional autoencoder module. We directly used K-means for clustering the features from concatenated autoencoder features. The ARI, NMI and Homogeneity decreases by 6.2, 7.2 and 7.1%, respectively. Second, we trained the chromosome-wise autoencoder model first and then fixed the weights in the autoencoder and trained the cell-wise scDEC module. Without joint training of the multi-stage modules, the performance also decreases by 2.4% of ARI, 2.8% of NMI and 2.5% of Homogeneity (Supplementary Table 3 available online at http://bib.oxfordjournals.org/). The model ablation studies demonstrate the significant contribution of both multi-stage model and joint training strategy.

Conclusion and discussion

In this study, we proposed scDEC-Hi-C, a computational tool for comprehensive single cell Hi-C data analysis using deep generative neural network. Unlike previous works that treat dimension reduction and clustering of the single cell Hi-C data as two separated and independent tasks, scDEC-Hi-C intrinsically integrates the task of learning a low-dimensional representation and clustering the single cell by designing a two-stage multi-scale framework, which is composed of a chromosome-wise autoencoder and a cell-wise symmetric GAN model. During the training, the multi-scale models are simultaneously optimized and the results of

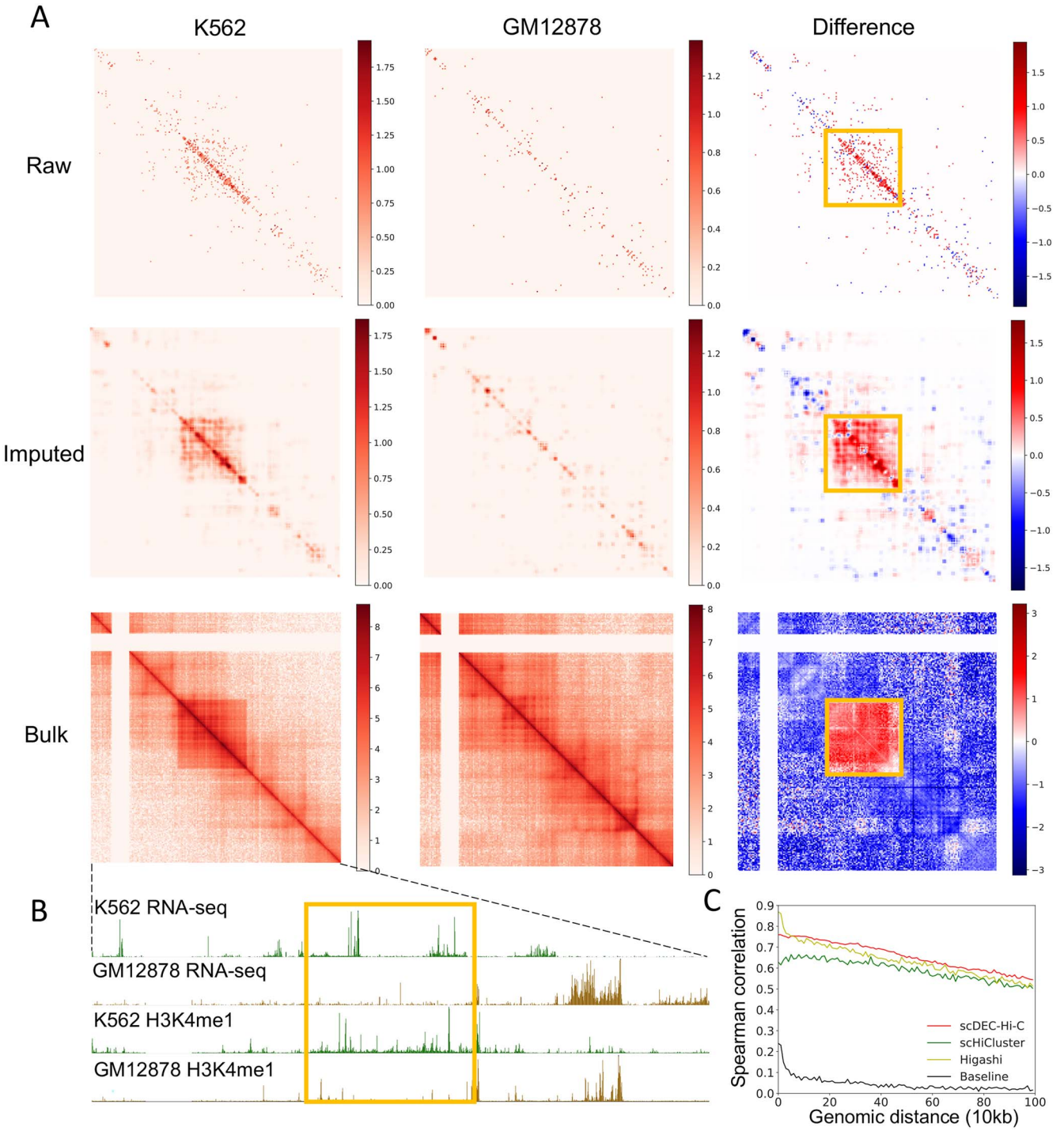

Figure 3. The imputation results of scDEC-Hi-C method. (**A**) The first row denotes merged single cell Hi-C profile of 40 cells of a genomic region (chr9: 132.9M–134.9M) across two diverse cell lines. The middle row denotes the corresponding imputed single cell Hi-C profile with scDEC-Hi-C. The third row denotes the corresponding bulk Hi-C profile of the two cell lines. The differences of the Hi-C profile from two cell lines are illustrated. (**B**) Genome annotation including RNA-seq and H3K4me1 histone marker across two cell lines of the same genomic region. (**C**) The Spearman correlation between bulk K562 Hi-C data and pseudo-bulk Hi-C data with imputation by scDEC-Hi-C (red), scHiCluster (green) and Higashi (yellow). The baseline (black) denotes the Spearman correlation between bulk K562 Hi-C data and pseudo-bulk Hi-C data without imputation.

embedding and clustering are benefitting each other. Based on a series of experiments, scDEC-Hi-C achieves superior or competitive performance compared to state-of-the-art baseline methods. For the downstream analysis, scDEC-Hi-C model demonstrated the excellent ability of imputing the sparse and noisy single cell Hi-C data, which facilitates the identification of chromatin structural differences and chromatin loops. Besides, scDEC-Hi-C also shows the superior power in generating the Hi-C profile of different cell types, which has been confirmed to be consistently with the cell type label (Supplementary Figure 6 available online at http://bib.oxfordjournals.org/).

We also provide several directions for further improving our work. First, the inter-chromosomal interactions, which were ignored by existing methods and scDEC-Hi-C, have been proved

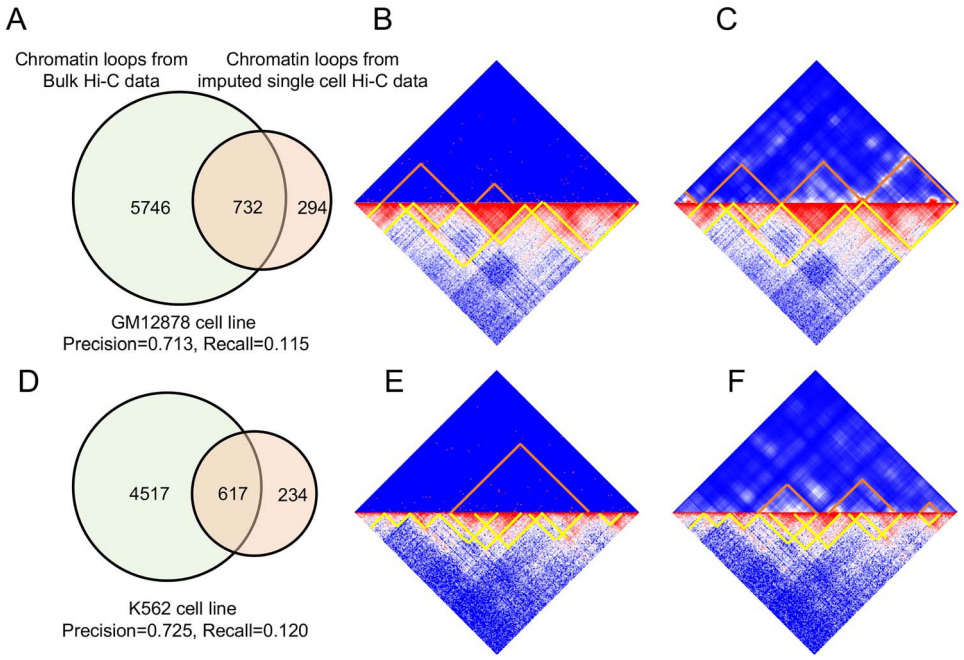

Figure 4. scDEC-Hi-C facilitates the identification of chromatin loops. (**A**) The Venn plot of chromatin loops from bulk Hi-C data and single-cell Hi-C data imputed by scDEC-Hi-C in GM12878 cell line. (**B**) The chromatin loops from raw single cell Hi-C data versus chromatin loops from bulk Hi-C data of a GM12878 cell line genomic region (chr3:118.2M–120.2M). (**C**) The chromatin loops from imputed single cell Hi-C data versus chromatin loops from bulk Hi-C data in GM12878 cell line of the same genomic region. (**D**) The Venn plot of chromatin loops from bulk Hi-C data and single-cell Hi-C data imputed by scDEC-Hi-C in K562 cell line. (**E**) The chromatin loops from raw single cell Hi-C data versus chromatin loops from bulk Hi-C data of a K562 cell line genomic region (chr3:118.2M–120.2M). (**F**) The chromatin loops from imputed single cell Hi-C data versus chromatin loops from bulk Hi-C data in in GM12878 cell line of the same genomic region.

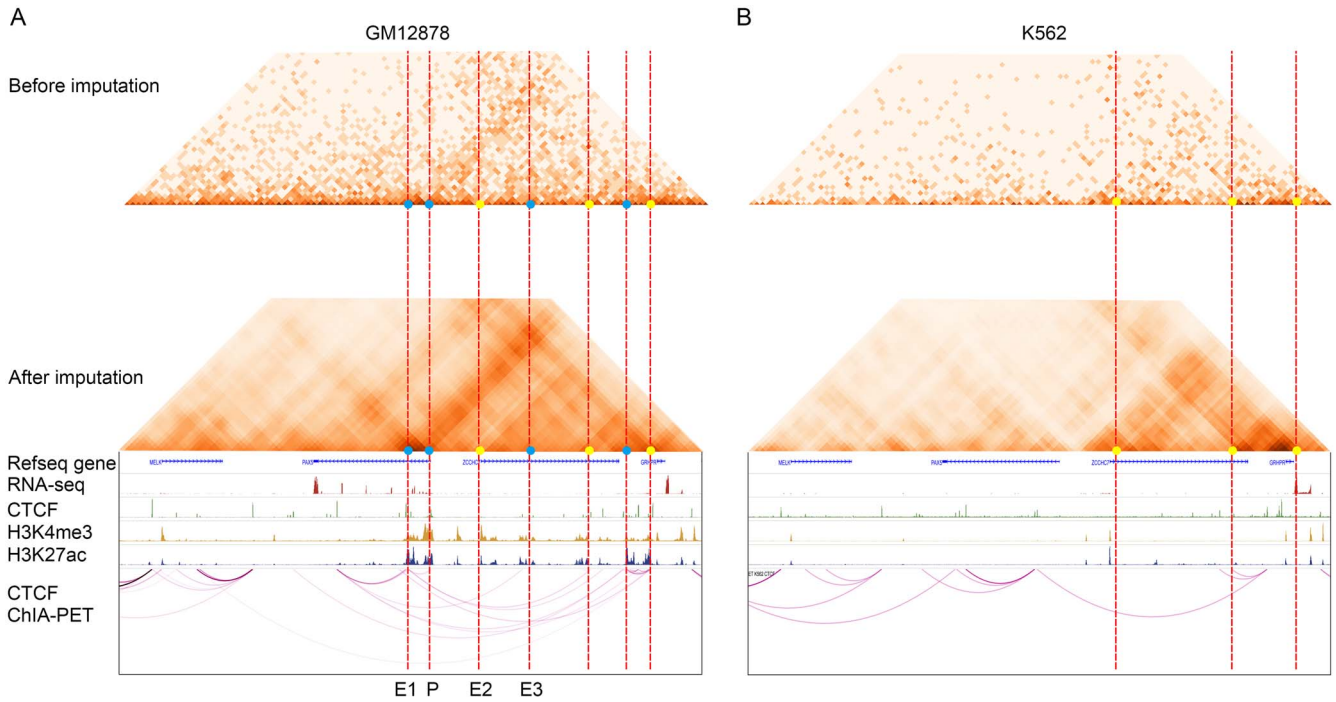

Figure 5. Application of scDEC-Hi-C in identifying the cell-type specific TAD-like boundaries. scHi-C data with/without imputation from a differential genomic region (chr9:36.5–37.5M) between GM12878 and K562 cell type. Several annotation tracks, including RNA-seq, ChIP-seq with CTCF, ChIP-seq with two histone modifications and ChIA-PET with CTCF target across two cell types were also shown below the Hi-C data. We observed that a B cell important regulator, Pax5, only expressed in GM12878 cell type. (**A**) Hi-C maps and annotation tracks in GM12878 cell type. (**B**) Hi-C maps and annotation tracks in K562 cell type.

to regulate gene expression [34]. Second, incorporating multi-omics data, including functional genomic regulatory annotation data [35, 36] and pharmaceutical interaction data [37, 38], could potentially improve the performance. Third, it is worthwhile for applying scDEC-Hi-C to other different types of 3D genome interaction data such as HiChIP [39].

With scDEC-Hi-C, researchers can perform single cell Hi-C experiments of the cell types or tissues with interest. Then one can simultaneously perform unsupervised learning analysis on single cell Hi-C data and uncover biological findings through the imputation and generation power. We expect that scDEC-Hi-C can help unveil the single cell regulation mechanism in 3D genome.

Methods

Data preprocessing

For Ramani dataset, we filtered cells with less than 5000 contacts. Then we collected 624 cells for Ramani dataset. For Dip-C dataset, we used the same QC strategy from the original paper [25] and collected 1954 annotated cells across 14 cell types. For Nagano dataset, we followed the data preprocessing strategy from [22] and collected 1171 cells from 4 cell cycle phases. The details of datasets were summarized in Supplementary Table 2 available online at http://bib.oxfordjournals.org/. The raw Hi-C contact matrices were log-transformed and then resized by spine interpolation so that the Hi-C contact matrix of each chromosome was represented as a 50 by 50 matrix. Then we applied a mean filtering and random walk as suggested by scHiCluster [21]. The chromosome-wise module encodes each chromosome into a 50-dimensional vector and then concatenated across all chromosomes. The cell-wise module further learns a low-dimensional representation of a cell with dimension of latent variable z set to 10. The embedding of each cell was based on the concatenation of reconstructed $\widetilde{z}$ and $\widetilde{c}$ (before softmax).

Adversarial training in scDEC-Hi-C model

The scDEC-Hi-C is multi-scale unsupervised learning model derived from our previous works Roundtrip and scDEC [24, 40] with extensive modifications. scDEC-Hi-C mainly contains a chromosome-wise module convolutional autoencoder (CAE) [41] and a cell-wise model scDEC. The CAE module aims at mapping scHi-C data from the original data space to a representer space, which significantly reduced the data dimension. Specifically, the CAE module takes the single cell Hi-C interaction of each chromosome as a training instance and each intra-chromosomal interaction matrix will be encoded to a fixed dimension vector through encoder. The embedding vectors for intra-chromosomal interaction matrices within each cell are concatenated in the representer space to obtain a fused embedding. The training of the CAE can be formulated as

$$\mathcal{L}_{\mathrm{AE}} = \mathbb{E}\left[\left\|x^{\mathrm{chr}} - D\left(E\left(x^{\mathrm{chr}}\right)\right)\right\|_F^2\right]$$

where x^{chr} denotes an intra-chromosomal Hi-C interaction matrix and $E(\bullet), D(\bullet)$ denote the encoder and decoder in the CAE module, respectively. The chromosome-wise features $E(x^{\mathrm{chr}})$ of each chromosome were concatenated to obtain the cell-wise representation by

$$x = \mathbf{Concat}\left(E\left(x^{\mathrm{chr1}}\right), \ldots, E\left(x^{\mathrm{chrX}}\right)\right)$$

The scDEC module takes cell-wise fused embedding in the representer space as input and learns the low-dimension embedding of a cell in the latent space and clusters the cells simultaneously. scDEC module is composed of a pair of two GAN models. For the forward GAN model, a pair of latent variables z and c are sampled from a Gaussian distribution and a Categorical distribution, respectively. The categorical distribution is updated through an adaptive mechanism (Supplementary Table 4 available online at http://bib.oxfordjournals.org/). G network is used for conditionally generating fake data $\{\tilde{x}_i\}_{i=1}^N$ that have a similar distribution to the real data $\{x_i\}_{i=1}^N$ in the representer space while the discriminator network $\mathbf{D}_x$ tries to discern true data from generated samples in the representer space. In the backward GAN model, the function H and the discriminator $\mathbf{D}_z$ aim at transforming the data from representer space to the latent space. Discriminators can be considered as binary classifiers where any input data point will be asserted to be positive or negative. Besides, we used WGAN-GP [42] as the architecture for the pair of GAN models where the gradient penalties of discriminators were considered as additional loss terms. We then define the objective loss functions of the above four networks ($\mathbf{G}, \mathbf{H}, \mathbf{D}_x$ and $\mathbf{D}_z$) in the training process as

$$\begin{cases} \mathcal{L}_{\mathrm{GAN}}(\mathbf{G}) = -\mathop{\mathbb{E}}_{z\sim p(z), c\sim \mathbf{Cat}(K,w)}\left[\mathbf{D}_x(\mathbf{G}(z,c))\right] \\ \mathcal{L}_{\mathrm{GAN}}(\mathbf{D}_x) = -\mathop{\mathbb{E}}_{x\sim p(x)}\left[\mathbf{D}_x(x)\right] + \mathop{\mathbb{E}}_{z\sim p(z), c\sim \mathbf{Cat}(K,w)}\left[\mathbf{D}_x(\mathbf{G}(z,c))\right] \\ \quad + \lambda \mathop{\mathbb{E}}_{\hat{x}\sim \hat{p}(\hat{x})}\left[\left(\left\|\nabla_{\hat{x}}\mathbf{D}_x(\hat{x})\right\|_2 - 1\right)^2\right] \\ \mathcal{L}_{\mathrm{GAN}}(\mathbf{H}) = -\mathop{\mathbb{E}}_{x\sim p(x)}\left[\mathbf{D}_z(\mathbf{H}(x))\right] \\ \mathcal{L}_{\mathrm{GAN}}(\mathbf{D}_z) = -\mathop{\mathbb{E}}_{z\sim p(z)}\left[\mathbf{D}_z(z)\right] + \mathop{\mathbb{E}}_{x\sim p(x)}\left[\mathbf{D}_z(\mathbf{H}(x))\right] \\ \quad + \lambda \mathop{\mathbb{E}}_{\bar{z}\sim \bar{p}(\bar{z})}\left[\left(\left\|\nabla_{\bar{z}}\mathbf{D}_z(\bar{z})\right\|_2 - 1\right)^2\right] \end{cases}$$

where $p(z)$ and $\mathbf{Cat}(K, w)$ denote the distribution of the continuous variable and discrete variable in the latent space. In practice, sampling$x$ from $p(x)$ can be regarded as a process of randomly sampling from *i.i.d* data in the representer space with replacement. $\hat{p}(\hat{x})$ and $\bar{p}(\bar{z})$ denote a uniformly sampling from the straight line between a pair of points sampled from true data and generated data in the representer and latent space, respectively. λ is a penalty coefficient, which is set to 10 in all experiments.

Roundtrip loss

During the training process, we also intend to minimize the roundtrip loss [40] which is defined as $\rho((z,c), \mathbf{H}(\mathbf{G}(z,c))))$and $\rho(x, \mathbf{G}(\mathbf{H}(x)))$ where z and c are sampled from $p(z)$ and $\mathbf{Cat}(K, w)$, respectively. The basic principle for this loss is to minimize the distance when a data point goes through a roundtrip transformation between two different data domains. Specifically, we applied l_2 loss to the continuous part in roundtrip loss and cross entropy loss to the discrete part in roundtrip loss. We further denoted the roundtrip loss as

$$\mathcal{L}_{\mathrm{RT}}(\mathbf{G},\mathbf{H}) = \alpha\|x - \mathbf{G}(\mathbf{H}(x))\|_2^2 + \alpha\|z - \mathbf{H}_z(\mathbf{G}(z,c))\|_2^2 + \beta \mathbf{CE}(c, \mathbf{H}_c(\mathbf{G}(z,c)))$$

where α and β are two coefficients and are both set to 10 in the experiments. $\mathbf{H}_z(\bullet)$ and $\mathbf{H}_c(\bullet)$ denote the continuous and discrete part of $\mathbf{H}(\bullet)$, respectively. $\mathrm{CE}(\bullet)$ represents the cross-entropy function. The idea of roundtrip loss, which exploits transitivity for regularizing structured data has also been used in previous works [43, 44].

Joint training

Combining the adversarial training loss and roundtrip loss together, we can get the full training loss for the scDEC module as $\mathcal{L}(\mathbf{G},\mathbf{H}) = \mathcal{L}_{\mathrm{GAN}}(\mathbf{G}) + \mathcal{L}_{\mathrm{GAN}}(\mathbf{H}) + \mathcal{L}_{\mathrm{RT}}(\mathbf{G},\mathbf{H})$ and$\mathcal{L}(\mathbf{D}_x,\mathbf{D}_z) = \mathcal{L}_{\mathrm{GAN}}(\mathbf{D}_x) + \mathcal{L}_{\mathrm{GAN}}(\mathbf{D}_z)$, respectively. We iteratively updated the weight parameters in two generative models (G and H) and the two discriminative models ($\mathbf{D}_x$ and $\mathbf{D}_z$), respectively. Thus, the

training of scDEC module can be represented as

$$G^*, D_x^*, H^*, D_z^* = \begin{cases} \arg\min_{G,H} \mathcal{L}(G, H) \\ \arg\min_{D_x, D_z} \mathcal{L}(D_x, D_z) \end{cases}$$

To further achieve joint training of CAE and scDEC modules, we first pretrained the CAE module for 100 epochs. Then we updated the parameters of CAE and scDEC iteratively. The Adam optimizer [45] with a learning rate of 2×10^{-4} was used for optimizing the parameters in neural networks. The whole training process is illustrated in Supplementary Table 5 available online at http://bib.oxfordjournals.org/ in detail.

Data imputation by scDEC-Hi-C model

We use the chromosome-wise model autoencoder for data imputation. Specifically, the reconstructed Hi-C map from the decoder was regarded as the imputed single cell Hi-C data. We used the same strategy in [13] for Hi-C matrices extraction.

Data generation by scDEC-Hi-C model

We generate the intermediate cell state (embeddings) of single cell Hi-C data by interpolating the latent indicator c of two 'neighboring' cell types. Assume that two cell types correspond to the latent indicator c_1 and c_2, respectively. The generated single cell Hi-C profile can be represented as $\mathbf{G}(z, \hat{c})$ where $\hat{c} = \alpha c_1 + (1 - \alpha)c_2$. Note that the α is the coefficient from 0 to 1 and z is sampled from a standard Gaussian distribution.

Network architecture in scDEC-Hi-C

For the CAE module, the encoder contains four convolutional layers and two fully connected layers while the decoder consists of two fully connected layers and four transposed convolutional layers for reconstructing the Hi-C interaction matrices. For the scDEC module. The **G** network contains five fully connected layers and each hidden layer has 512 nodes while the **H** network contains five fully-connected layers and each hidden layer has 256 nodes. $\mathbf{D}_x$ and $\mathbf{D}_z$ both contain two fully connected layers and 128 nodes in the hidden layer. Note that batch normalization [46] was used in discriminator networks.

Updating the Category distribution

The probability parameter w in the Category distribution $\mathbf{Cat}(K, w)$ is adaptively updated every 200 batches of data based on the inferred cluster label (Supplementary Table 4 available online at http://bib.oxfordjournals.org/). K can be estimated by gap statistic [47] if not provided by user.

Evaluation metrics for clustering

We compared different methods for clustering according to three commonly used metrics, normalized mutual information (NMI) [48], adjusted Rand index (ARI) [49] and Homogeneity [50]. Assuming that U and V are true label assignment and predicted label assignment given n observation data points, which have C_U and C_V clusters in total, respectively. NMI is then calculated as

$$\mathrm{NMI} = \frac{\sum_{p=1}^{C_U}\sum_{q=1}^{C_V} |U_p \cap V_q| \log \frac{n|U_p \cap V_q|}{|U_p| \times |V_q|}}{\max\left(-\sum_{p=1}^{C_U} |U_p| \log \frac{|U_p|}{n}, -\sum_{q=1}^{C_V} |V_q| \log \frac{|V_q|}{n}\right)}$$

The Rand index [51] is a measure of agreement between two cluster assignments while ARI corrects lacking a constant value when the cluster assignments are selected randomly. We define the following four quantities: (1) n_1: number of pairs of two objects in the same groups in both U and V, (2) n_2: number of pairs of two objects in different groups in both U and V, (3) n_3: number of pairs of two objects in the same group of U but different group in V, (4) n_4: number of pairs of two objects in the same group of V but different group in $\mathbf{U}$. Then ARI is calculated by

$$\mathrm{ARI} = \frac{\binom{n}{2}(n_1 + n_4) - [(n_1 + n_2)(n_1 + n_3) + (n_3 + n_4)(n_2 + n_4)]}{\binom{n}{2}^2 - [(n_1 + n_2)(n_1 + n_3) + (n_3 + n_4)(n_2 + n_4)]}$$

Homogeneity is calculated by $\mathbf{Homo} = 1 - \frac{H(U|V)}{H(U)}$, where

$$\begin{cases} H(U|V) = -\sum_{p=1}^{C_U}\sum_{q=1}^{C_V} \frac{|U_p \cap V_q|}{n} \log \frac{|U_p \cap V_q|}{\sum_{q=1}^{C_V} |U_p \cap V_q|} \\ H(U) = -\sum_{p=1}^{C_U} \frac{\sum_{q=1}^{C_V} |U_p \cap V_q|}{C_U} \log \frac{\sum_{q=1}^{C_V} |U_p \cap V_q|}{C_U} \end{cases}$$

Baseline methods

We compared scDEC-Hi-C to three comparison methods in our study. scHiCluster is a PCA-based method that could be used for imputing and clustering scHi-C data. LDA was implemented from https://github.com/khj3017/schic-topic-model and the default parameters were used. scHiCluster was implemented from https://github.com/zhoujt1994/scHiCluster and the default parameters were used. HiCRep/MDS used multidimensional scaling to embed scHi-C data into two dimension and was implemented from https://github.com/liu-bioinfo-lab/scHiCTools. Higashi is a hypergraph representation learning framework for embedding scHi-C data. We downloaded Higashi from https://github.com/ma-compbio/Higashi and implemented using the default parameters.

Key Points

- scDEC-Hi-C provides an end-to-end framework based on the combination of autoencoder and deep generative model to comprehensively analyze single cell Hi-C data, including low-dimensional embedding and clustering.
- Through a series of experiments including single cell Hi-C data clustering and structural difference identification, scDEC-Hi-C demonstrates superior performance over existing methods.
- In the downstream analysis of chromatin loops from single cell Hi-C data, scDEC-Hi-C is capable of significantly enhancing the ability for identifying Hi-C chromatin loops by data imputation.

Supplementary Data

Supplementary data are available online at https://academic.oup.com/bib.

Acknowledgement

The authors thank the anonymous reviewers for their valuable and constructive suggestions.

Author contributions statement

Q.L., R.J., M.Z. and S.T.Z. conceived the study. Q.L. designed and implemented scDEC-Hi-C. Q.L. and W.W.Z. performed data

analysis. Q.L. and W.W.Z. interpreted the results. Q.L. wrote the manuscript. Other authors provided editorial support. All authors read and approved the final version of the manuscript.

Funding

This work was partially supported by the National Key Research and Development Program of China (Grant No. 2021YFF1200902, 2021YFF1001000, 2022YFB4500300), the National Natural Science Foundation of China (Grant Nos. 62003178,62273194, 61873141, 61721003, 62271452). H. Chen is with the Research Center for Graph Computing, Zhejiang Lab, Hangzhou 311100, China. He is supported inpart by Key Research Project of Zhejiang Lab (No. 2022PI0AC01).

Code availability

scDEC-Hi-C is an open-source software based on the TensorFlow library [52], which can be downloaded from https://github.com/kimmo1019/scDEC-Hi-C.

Data availability

Three datasets were used in this study. scHi-C dataset of four human cell lines (GM12878, HAP1, HeLa and K562) was collected from Ramani *et al.* (GEO: GSE84920). scHi-C dataset (Dip-C) of mouse brain development was collected from Tan *et al.* (GEO: GSE162511). scHi-C dataset of cell cycle was collected from Nagano *et al.* (GEO: GSE94489). Note that Dip-C only contains annotated labels, which were used as surrogate labels in the clustering experiments.

References

1. Buenrostro JD, Wu B, Litzenburger UM, *et al.* Single-cell chromatin accessibility reveals principles of regulatory variation. *Nature* 2015;**523**:486–90.
2. Gawad C, Koh W, Quake SR. Single-cell genome sequencing: current state of the science. *Nat Rev Genet* 2016;**17**:175–88.
3. Shapiro E, Biezuner T, Linnarsson S. Single-cell sequencing-based technologies will revolutionize whole-organism science. *Nat Rev Genet* 2013;**14**:618–30.
4. Stoeckius M, Hafemeister C, Stephenson W, *et al.* Simultaneous epitope and transcriptome measurement in single cells. *Nat Methods* 2017;**14**:865–8.
5. Stuart T, Butler A, Hoffman P, *et al.* Comprehensive integration of single-cell data. *Cell* 2019;**177**:1888, e1821–902.
6. Butler A, Hoffman P, Smibert P, *et al.* Integrating single-cell transcriptomic data across different conditions, technologies, and species. *Nat Biotechnol* 2018;**36**:411–20.
7. Liu Q, Xia F, Yin Q, *et al.* Chromatin accessibility prediction via a hybrid deep convolutional neural network. *Bioinformatics* 2018;**34**:732–8.
8. Duren Z, Chang F, Naqing F, *et al.* Regulatory analysis of single cell multiome gene expression and chromatin accessibility data with scREG. *Genome Biol* 2022;**23**:114.
9. Yin Q, Wu M, Liu Q, *et al.* DeepHistone: a deep learning approach to predicting histone modifications. *BMC Genomics* 2019;**20**: 11–23.
10. Liu Q, Hua K, Zhang X, *et al.* DeepCAGE: incorporating transcription factors in genome-wide prediction of chromatin accessibility. *Genom Proteom Bioinform* 2022. (In press).
11. Yin Q, Liu Q, Fu Z, *et al.* scGraph: a graph neural network-based approach to automatically identify cell types. *Bioinformatics* 2022;**38**(11):2996–3003. https://doi.org/10.1093/bioinformatics/btac199.
12. Rao SS, Huntley MH, Durand NC, *et al.* A 3D map of the human genome at kilobase resolution reveals principles of chromatin looping. *Cell* 2014;**159**:1665–80.
13. Liu Q, Lv H, Jiang R. hicGAN infers super resolution Hi-C data with generative adversarial networks. *Bioinformatics* 2019;**35**:i99–107.
14. Marchal C, Sima J, Gilbert DM. Control of DNA replication timing in the 3D genome. *Nat Rev Mol Cell Biol* 2019;**20**:721–37.
15. Nagano T, Lubling Y, Stevens TJ, *et al.* Single-cell Hi-C reveals cell-to-cell variability in chromosome structure. *Nature* 2013; **502**:59–64.
16. Flyamer IM, Gassler J, Imakaev M, *et al.* Single-nucleus Hi-C reveals unique chromatin reorganization at oocyte-to-zygote transition. *Nature* 2017;**544**:110–4.
17. Ramani V, Deng X, Qiu R, *et al.* Massively multiplex single-cell Hi-C. *Nat Methods* 2017;**14**:263–6.
18. Stevens TJ, Lando D, Basu S, *et al.* 3D structures of individual mammalian genomes studied by single-cell Hi-C. *Nature* 2017;**544**:59–64.
19. Tan L, Xing D, Chang CH, *et al.* Three-dimensional genome structures of single diploid human cells. *Science* 2018;**361**:924–8.
20. Kim HJ, Yardimci GG, Bonora G, *et al.* Capturing cell type-specific chromatin compartment patterns by applying topic modeling to single-cell Hi-C data. *PLoS Comput Biol* 2020;**16**:e1008173.
21. Zhou J, Ma J, Chen Y, *et al.* Robust single-cell Hi-C clustering by convolution- and random-walk-based imputation. *Proc Natl Acad Sci U S A* 2019;**116**:14011–8.
22. Liu J, Lin D, Yardimci GG, *et al.* Unsupervised embedding of single-cell Hi-C data. *Bioinformatics* 2018;**34**:i96–104.
23. Zhang R, Zhou T, Ma J. Multiscale and integrative single-cell Hi-C analysis with Higashi. *Nat Biotechnol* 2022;**40**:254–61.
24. Liu Q, Chen S, Jiang R, *et al.* Simultaneous deep generative modeling and clustering of single cell genomic data. *Nat Mach Intell* 2021;**3**:536–44.
25. Tan L, Ma W, Wu H, *et al.* Changes in genome architecture and transcriptional dynamics progress independently of sensory experience during post-natal brain development. *Cell* 2021;**184**:741 e717-58.
26. Hao Y, Hao S, Andersen-Nissen E, *et al.* Integrated analysis of multimodal single-cell data. *Cell* 2021;**184**:3573 e3529-87.
27. Ramirez F, Bhardwaj V, Arrigoni L, *et al.* High-resolution TADs reveal DNA sequences underlying genome organization in flies. *Nat Commun* 2018;**9**(1):189. https://doi.org/10.1038/s41467-017-02525-w.
28. Wang R, Chen F, Chen Q, *et al.* MyoD is a 3D genome structure organizer for muscle cell identity. *Nat Commun* 2022;**13**(1):205. https://doi.org/10.1038/s41467-021-27865-6.
29. Consortium EP. An integrated encyclopedia of DNA elements in the human genome. *Nature* 2012;**489**:57–74.
30. Li D, Hsu S, Purushotham D, *et al.* WashU epigenome browser update 2019. *Nucleic Acids Res* 2019;**47**:W158–65.
31. Ay F, Bailey TL, Noble WS. Statistical confidence estimation for Hi-C data reveals regulatory chromatin contacts. *Genome Res* 2014;**24**:999–1011.
32. Medvedovic J, Ebert A, Tagoh H, *et al.* Pax5: a master regulator of B cell development and leukemogenesis. *Adv ImmunolElsevier* 2011;**111**:179–206.
33. Smith EM, Lajoie BR, Jain G, *et al.* Invariant TAD boundaries constrain cell-type-specific looping interactions between promoters

and distal elements around the CFTR locus. *Am J Hum Genet* 2016;**98**:185–201.
34. Xiong K, Ma J. Revealing Hi-C subcompartments by imputing inter-chromosomal chromatin interactions. *Nat Commun* 2019;**10**:5069.
35. Zeng W, Chen S, Cui X, *et al.* SilencerDB: a comprehensive database of silencers. *Nucleic Acids Res* 2021;**49**:D221–8.
36. Chen S, Liu Q, Cui X, *et al.* OpenAnnotate: a web server to annotate the chromatin accessibility of genomic regions. *Nucleic Acids Res* 2021;**49**:W483–90.
37. Xu C, Liu Q, Huang M, *et al.* Reinforced molecular optimization with neighborhood-controlled grammars. *Adv Neural Inf Process Syst* 2020;**33**:8366–77.
38. Liu Q, Hu Z, Jiang R, *et al.* DeepCDR: a hybrid graph convolutional network for predicting cancer drug response. *Bioinformatics* 2020;**36**:i911–8.
39. Mumbach MR, Rubin AJ, Flynn RA, *et al.* HiChIP: efficient and sensitive analysis of protein-directed genome architecture. *Nat Methods* 2016;**13**:919–22.
40. Liu Q, Xu J, Jiang R, *et al.* Density estimation using deep generative neural networks. *Proc Natl Acad Sci* 2021;**118**(15): e2101344118. https://doi.org/10.1073/pnas.2101344118.
41. Masci J, Meier U, Cireşan D, *et al.* Stacked convolutional auto-encoders for hierarchical feature extraction. In: *International Conference on Artificial Neural Networks*. Springer, Berlin, Germany, 2011, p. 52–9.
42. Gulrajani I, Ahmed F, Arjovsky M, *et al.* Improved training of wasserstein gans. In: *Advances in Neural Information Processing Systems*. IEEE, NY, United States, 2017, 5767–77.
43. Yi Z, Zhang H, Tan P, *et al.* Dualgan: Unsupervised dual learning for image-to-image translation. In: *Proceedings of the IEEE International Conference on Computer Vision*. 2017, 2849–57.
44. Zhu J-Y, Park T, Isola P, *et al.* Unpaired image-to-image translation using cycle-consistent adversarial networks. In: *Proceedings of the IEEE International Conference on Computer Vision*. IEEE, NY, United States, 2017, p. 2223–32.
45. Kingma DP, Ba J. Adam: A method for stochastic optimization. In: *Proceedings of the 2015 International Conference on Learning Representations (ICLR)*. 2015.
46. Ioffe S, Szegedy C. Batch normalization: accelerating deep network training by reducing internal covariate shift. In: *Proceedings of the 32nd International Conference on Machine Learning (ICML)*. 2015, p. 448–56.
47. Tibshirani R, Walther G, Hastie T. Estimating the number of clusters in a data set via the gap statistic. *J R Stat Soc Series B Stat Methodology* 2001;**63**(2):411–23. https://doi.org/10.1111/1467-9868.00293.
48. Strehl A, Ghosh J. Cluster ensembles—a knowledge reuse framework for combining multiple partitions. *J Mach Learn Res* 2002;**3**: 583–617.
49. Hubert L, Arabie P. Comparing partitions. *J Classif* 1985;**2**: 193–218.
50. Rosenberg A, Hirschberg J. V-measure: a conditional entropy-based external cluster evaluation measure. In: *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*. Association for Computational Linguistics, PA, United States, 2007, p. 410–20.
51. Rand WM. Objective criteria for the evaluation of clustering methods. *J Am Stat Assoc* 1971;**66**:846–50.
52. Abadi M, Barham P, Chen J, *et al.* Tensorflow: a system for large-scale machine learning. In: *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*. USENIX Association, CA, United States, 2016, p. 265–83.