

Dual Advancement of Representation Learning and Clustering for Sparse and Noisy Images

Anonymous Authors

ABSTRACT

Sparse and noisy images (SNIs), like those in spatial gene expression data, pose significant challenges for effective representation learning and clustering, which are essential for thorough data analysis and interpretation. In response to these challenges, we propose Dual Advancement of Representation Learning and Clustering (*DARLC*), an innovative framework that leverages contrastive learning to enhance the representations derived from masked image modeling. Simultaneously, *DARLC* integrates cluster assignments in a cohesive, end-to-end approach. This integrated clustering strategy addresses the “class collision problem” inherent in contrastive learning, thus improving the quality of the resulting representations. To generate more plausible positive views for contrastive learning, we employ a graph attention network-based technique that produces denoised images as augmented data. As such, our framework offers a comprehensive approach that improves the learning of representations by enhancing their local perceptibility, distinctiveness, and the understanding of relational semantics. Furthermore, we utilize a Student’s t mixture model to achieve more robust and adaptable clustering of SNIs. Extensive evaluation on 12 real-world datasets of SNIs, representing spatial gene expressions, demonstrates *DARLC*’s superiority over current state-of-the-art methods in both image clustering and generating representations that accurately reflect biosemantic content and gene interactions.

CCS CONCEPTS

• Computing methodologies → Computer vision.

KEYWORDS

Representation Learning, Clustering

1 INTRODUCTION

Sparse and noisy images (SNIs), commonly encountered in specialized fields like biomedical sciences, astronomy, and microscopy [33, 39, 48], are characterized by extensive uninformative regions (e.g., voids or background areas), considerable image noise, and severely fragmented visual patterns. These characteristics significantly increase the complexity in analysis and interpretation. A prime example is spatial gene expression Pattern (SGEP) images generated through spatial transcriptomics (ST) technology [35]. As illustrated in Figure 1, the high levels of sparsity and noise of an SGEP image

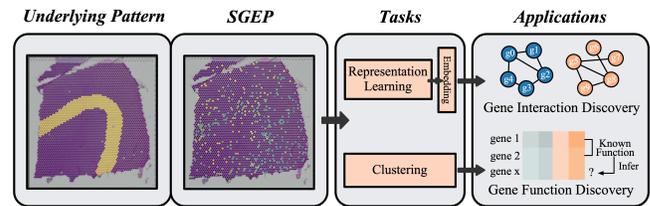


Figure 1: An example of sparse and noisy SGEP image is displayed in the second panel to the left. The gene expression levels across space are represented by pixel brightness, while void expression areas are displayed in purple. The first panel showcases the regions (cortical layer V) where the gene is significantly expressed, namely the major gene expression pattern. The right two panels illustrate the tasks and applications that utilize SGEP images.

complicate the discerning of its underlying major gene expression pattern.

Image clustering can group unlabelled images into distinct clusters, facilitating the exploration of image-implied semantics or functions. For example, clustering SGEP images offers a cost-effective means to identify groups of cofunctional genes and infer gene functions [34]. To obtain informative clustering results, it is essential to learn meaningful image representations, for which self-supervised learning (SSL) is the predominant approach in general scenarios. These include contrastive learning (CL) methods, exemplified by MoCo [20], and masked image modeling (MIM) such as MAE [19]. These methods offer distinct learning perspectives: MIM methods tend to learn local context-aware, holistic features for reconstruction tasks [19], while CL methods focus on learning instance-wise discriminative features [21]. Acknowledging the synergistic advantages of these methodologies, some researchers are endeavoring to utilize CL to refine the representations acquired through MIM [21, 47]. Moreover, many efforts [25, 32, 41] are directed towards guiding the process of learning representations through clustering tasks. This is achieved by jointly learning representations and executing clustering in an integrated and end-to-end manner, yielding image representations that are well-suited for clustering tasks.

However, for SNIs, both representation learning and clustering present significant challenges. Firstly, the widespread presence of uninformative voids or background areas, along with elevated noise levels and extremely fragmented visual patterns, substantially impedes the extraction of semantically meaningful visual features. This challenge has been highlighted in prior studies [30] and is further corroborated by our experiments (see Supplementary Table 1). Secondly, the inherent random noise across pixels induces considerable variability in visual patterns, even among images of

Permission to make digital or hard copies of all or part of this work for personal or professional use, not for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or to publish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

ACM MM, 2024, Melbourne, Australia
© 2024 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 978-x-xxxx-xxxx-x/YY/MM
<https://doi.org/10.1145/nnnnnnn.nnnnnn>

the same category [34], exposing clustering algorithms to a high overfitting risk [1].

Inspired by the aforementioned works, in order to better analyze SNIs, we propose a novel and unified framework, named Joint Learning of Representations and Cluster Assignments (*DARLC*). This framework not only leverages CL to boost the representation learned by MIM but also jointly learns cluster assignments in a self-paced and end-to-end manner, further refining the representation. Nonetheless, our experiments (see Supplementary Table 2) showcase that conventional data augmentation techniques (e.g., cropping and rotating) are ineffective for SNIs, as the augmented images often contain substantial void regions and noise, hampering the extraction of informative visual features. To overcome this limitation, we introduce a data augmentation method based on a graph attention network (GAT) [13, 37] that aggregates information from neighboring pixels to enhance visual patterns, generating smoothed images that act as more plausible positive views so as to improve the effectiveness of contrastive learning.

Additionally, we observed that the clustering algorithms used in many deep clustering methods are either sensitive to outliers, as demonstrated by the Gaussian mixture model (GMM) in DAGMM [49] and manifold clustering in EDESC [3], or lack the flexibility to different data distributions, such as the inflexible Cauchy kernel-based method in DEC [41]. In response, *DARLC* employs a specialized nonlinear projection head to normalize image embeddings, aligning them more closely with a t-distribution. This is followed by modeling with a Student's t mixture model (SMM) for soft clustering. SMM provides a more robust solution by down-weighting extreme values and is more adaptable by altering the degrees of freedom, making it particularly suitable for clustering in the context of SNIs. Furthermore, the clustering loss in *DARLC* also serves to regularize CL, alleviating the "class collision problem" that stems from false negative pairs in CL [4]. Unlike existing regularized CL methods [12, 24, 45], which directly integrate clustering into the CL throughout training, this clustering follows the "warm-up" representation learning, significantly expediting training convergence and enhancing clustering accuracy, as demonstrated in our ablation study. All these features collectively contribute to the finding that *DARLC*-generated image representations not only enhance clustering performance but also exhibit improvements in other semantic distance-based tasks, such as the discovery of functionally interactive genes. In summary, our main contributions are:

- We propose *DARLC*, a novel unified framework for joint representation learning and clustering of SNIs. *DARLC* marks the first endeavor in integrating contrastive learning, MIM and deep clustering into a cohesive process for representation learning. The resultant representations not only enhance image clustering performance but also benefit other semantic distance-based tasks.
- *DARLC* has developed a data augmentation method more suitable for SNIs, using a GAT to generate smoothed images as plausible positive views for CL.
- An SMM-based method is designed to cluster SNIs in a more robust and adaptable manner. Additional features of this clustering method include a novel Laplacian loss for guiding the initial phase of clustering, and a differentiable cross-entropy hinge loss

for controlling cluster sizes. This clustering also addresses the class collision problem by pulling close related instances.

- Extensive experiments have been conducted across 12 real SGEP datasets [28]. Our results show that *DARLC* surpasses the state-of-the-art (SOTA) methods in both image clustering and generating image representations that accurately capture gene interactions.

2 RELATED WORKS

2.1 Self-supervised Representation Learning for Images

Most related SSL studies include CL and MIM methods. In CL, an input instance forms positive pairs with its augmented views, while forms negative pairs with other instances. Paradigmatic CL methods aim to learn instance discriminative representations by maximizing the similarity between positive pairs while minimizing it between negative pairs in a latent space [8, 20]. To address the class collision problem due to false negative samples, several studies [12, 24, 45] regularize CL with clustering, while others [5, 6, 9, 17] bypass the using of negative samples altogether. In contrast, MIM methods focus on learning local context-aware features by restoring raw pixel values from masked image patches [2, 16, 19, 42]. Several researchers are realizing the advantages of integrating these methodologies and endeavoring to utilize CL to refine MIM-generated representations [21, 42]. For instance, iBOT [47] contrasts between the reconstructed tokens of masked and unmasked image patches. Yet, to the best of our knowledge, *DARLC* is the first method that learns image representations from all aspects of discriminability, local perceptibility, and relational semantic structures.

2.2 Deep Image Clustering.

Related deep image clustering studies include deep autoencoder-based methods [32], which couple representation learning with deep embedded clustering in an end-to-end manner, as exemplified by methods like DEC [18, 23, 41]. Subsequent improvements to DEC focus on strategies like overweighing reliable samples (e.g., IDCEC [29]), and replacing Euclidean distance-based clustering with deep subspace clustering (e.g., EDESC [3]) or GMM-based clustering [38, 49]. Recent studies, including CC [25], DCP [27], CVCL [7], and CCES [43], directly integrate CL into the clustering process by contrasting at both instance and cluster levels across views, generating a soft cluster assignment matrix as deep embeddings for iterative refinement. Compared to these methods, *DARLC* offers a more comprehensive and potent mechanism for learning deep embeddings, a more robust and adaptable clustering algorithm, and a warm-up representation learning phase for accelerating clustering convergence.

3 METHODOLOGY

3.1 Overview

The framework of *DARLC*, as illustrated in Figure 2 and Algorithm 1, comprises two modules: a self-supervised representation learning and a deep clustering. The self-supervised representation learning module unifies CL and MIM, encompassing three encoders: an on-line encoder and a target momentum encoder for the contrastive

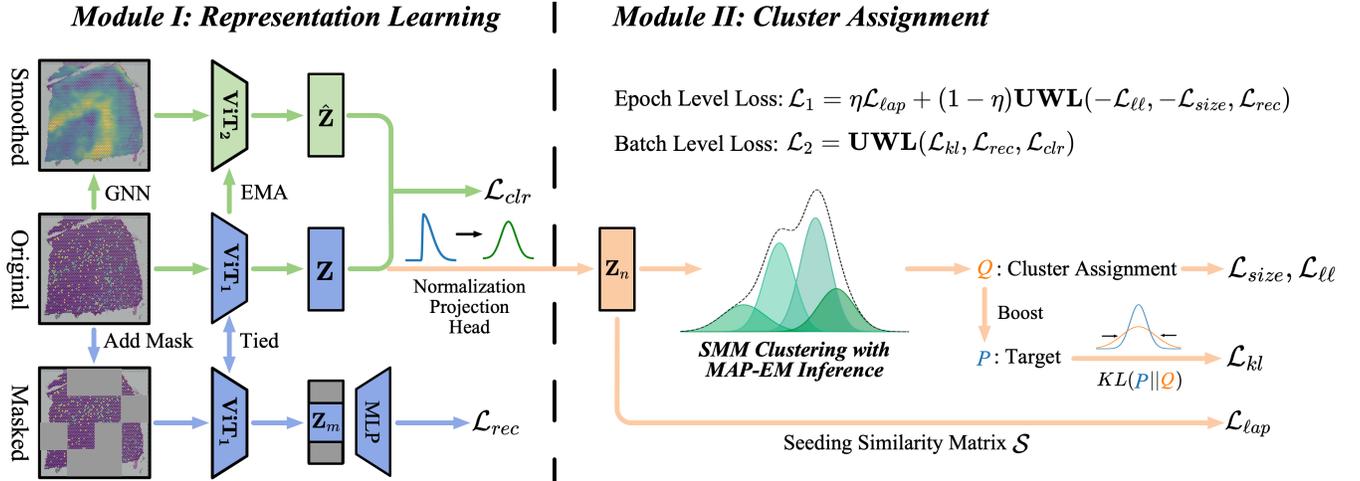


Figure 2: The framework of DARLC consists of two components: the representation learning and the deep clustering. The representation learning component integrates MIM and CL to generate image embeddings, which are then normalized through a non-linear projection head. Normalized representations are modeled by an SMM to derive their soft cluster assignments, which are used to construct various loss functions. With these loss functions, the two components are jointly optimized in a self-paced and end-to-end manner.

branch, and a masked encoder for the MIM branch. All three encoders adopt the identical vision transformer (ViT) [14] architecture, with shared parameters between the online encoder and masked encoder. The parameters of the target momentum encoder are updated using exponential moving average (EMA), as suggested by BYOL [17]. The initial phase of the unified representation learning involves a warm-up pretraining to generate preliminary embeddings, which are then normalized by a nonlinear projection head. This normalization aligns the embeddings more closely with t-distributions, setting the stage for subsequent t-distribution-based clustering. The deep clustering module utilizes a SMM to cluster the normalized embeddings, generating soft cluster assignment scores involved in the calculation of various loss functions. There are two types of loss functions: \mathcal{L}_1 , an epoch-level loss function for maximizing the empirical likelihood of observed instances, and \mathcal{L}_2 , a batch-level loss function for discriminatively boosted clustering optimization [41]. The two loss functions work in tandem, enabling the joint refinement of image embeddings and cluster assignments in a self-paced and end-to-end manner.

3.2 Unified Self-supervised Representation Learning (Module I)

3.2.1 Denoising-based Data Augmentation. We train a graph attention autoencoder \mathcal{G} to generate smoothed images, serving as augmented positive instances [13, 37]. Initially, for each image, we construct an undirected and unweighted graph by treating pixels as nodes connected to their k-nearest neighbors. Specifically, for a given image $X \in \mathbb{R}^{C \times H \times W}$, where C is the number of channels, H and W are the height and width of the image, respectively. Let $N_{pix} = H \times W$ denote the number of pixels, $v_i \in \mathbb{R}^C$ denote the pixel vector at location i , $\forall i \in \{1, 2, \dots, N_{pix}\}$. The encoder in \mathcal{G} comprises L layers. For each layer $t \in \{1, 2, \dots, L-1\}$, with the initial

value $\mathbf{h}_i^{(0)} = v_i$, the output $\mathbf{h}_i^{(t)} \in \mathbb{R}^{d_p}$ is calculated as follows:

$$\mathbf{h}_i^{(t)} = \text{LeakyReLU}\left(\sum_{v \in S_i} \text{att}_{iv}(\mathbf{W}^{(t)} \mathbf{h}_v^{(t-1)})\right), \quad (1)$$

where $\mathbf{W}^{(t)}$ represents the trainable weights of the t -th autoencoder layer, S_i the set of node i 's neighbors within a pre-specified radius r . The attention score, att_{iv} , between nodes i and v are computed as follows:

$$\alpha_{iv}^{(t)} = \mathbf{w}_{att}^{(t)} \text{LeakyReLU}(\mathbf{W}^{(t)} [\mathbf{h}_i^{(t-1)}] \|\mathbf{h}_v^{(t-1)}]), \quad (2)$$

$$\text{att}_{iv}^{(t)} = \frac{\exp(\alpha_{iv}^{(t)})}{\sum_{i \in S_i} \exp(\alpha_{iv}^{(t)})}, \quad (3)$$

where $\mathbf{w}_{att}^{(t)}$ represents the trainable attention weights. The decoder of \mathcal{G} mirrors the encoder with tied weights. The total loss function is defined as:

$$L_{denoise} = \sum_{i=1}^{N_{pix}} \|\mathbf{v}_i - \tilde{\mathbf{h}}_i^{(0)}\|_2, \quad (4)$$

where $\tilde{\mathbf{h}}_i^{(0)}$ denotes the reconstructed \mathbf{v}_i output by the decoder. Once trained, \mathcal{G} is applied to any given image X_i , to generate a smoothed image \bar{X}_i for a given image X_i as:

$$\bar{X}_i = \mathcal{G}(X_i, \mathbf{W}, \mathbf{w}_{att}) \quad (5)$$

3.2.2 Unified Self-supervised Image Representation Learning. This SSL model encompasses two branches: a MIM branch \mathcal{M} and a contrastive branch \mathcal{C} . \mathcal{M} is an adapted version of MAE, specifically designed for generating image patch embeddings in the context of SNIs. In this adaption, the standard MAE encoder is replaced with a lightweight ViT encoder, denoted as M_E , with four transformer blocks, four attention heads, and a higher masking ratio (80%). Meanwhile, the original transformer-based MAE decoder,

Algorithm 1 Algorithm for Dual Advancement of Representations Learning and Clustering (DARLC).

Input: Images $\mathcal{X} \in \mathbb{R}^{N \times C \times H \times W}$; Seeding similarity matrix $\mathcal{S} \in \mathbb{R}^{N \times N}$; Denoise GAT \mathcal{G} ; Maximum epochs E_{max} ; Number of images N ; Number of clusters K .

Definition: Parametes of \cdot are denoted as $\Pi(\cdot)$; MIM branch \mathcal{M} ; CL branch \mathcal{C} ; Projection head \mathcal{H} ; UWL (Uncertain Weight Loss function); EMA (Exponential Moving Average function).

Output: Representations $Z \in \mathbb{R}^{N \times D}$; Soft clustering $Q \in \mathbb{R}^{N \times K}$.

```

1: Compute smoothed image  $\tilde{\mathcal{X}}$  by Eq. (5).
2: while  $epoch < E_{max}$  do
3:   for  $X_b, \bar{X}_b$  in  $\mathcal{X}, \tilde{\mathcal{X}}$  do
4:     Compute  $\hat{X}_b$  by Eq. (6)  $\rightarrow \mathcal{L}_{rec}(X_b, \hat{X}_b)$  by Eq. (7).
5:     Compute  $e_b, \bar{e}_b$  by Eq. (9)  $\rightarrow \mathcal{L}_{clr}(e_b, \bar{e}_b)$  by Eq. (10).
6:      $\mathcal{L}_{ssl} = \text{UWL}(\mathcal{L}_{rec}, \mathcal{L}_{clr})$ .
7:     Update  $\Pi(\mathcal{M})$  using  $\mathcal{L}_{ssl}$ .
8:     Update  $\Pi(\mathcal{C})$  with EMA( $\Pi(\mathcal{M})$ ).
9:   end for
10:  end while
11: while (not converged) & ( $epoch < E_{max}$ ) do
12:   Compute  $\tilde{\mathcal{X}}$  by Eq. (6)  $\rightarrow \mathcal{L}_{rec}(\mathcal{X}, \tilde{\mathcal{X}})$  by Eq. (7).
13:   Compute  $Z$  by Eqs. (9), (12)  $\rightarrow \mathcal{L}_{lap}(Z, \mathcal{S})$  by Eq. (19).
14:   SMM parameters inference using MAP-EM  $\rightarrow \Theta$ .
15:    $Q = \text{SMM}(Z|\Theta) \rightarrow \mathcal{L}_{size}(Q), \mathcal{L}_{\ell\ell}(Q)$  by Eqs. (22), (20).
16:    $\mathcal{L}_1 = \eta \mathcal{L}_{lap} + (1 - \eta) \text{UWL}(-\mathcal{L}_{\ell\ell}, -\mathcal{L}_{size}, \mathcal{L}_{rec})$ .
17:   Update  $\Pi(\mathcal{M}), \Pi(\mathcal{C}), \Pi(\mathcal{H})$  using  $\mathcal{L}_1$ .
18:   for  $X_b, \bar{X}_b$  in  $\mathcal{X}, \bar{\mathcal{X}}$  do
19:     Compute  $P$  by Eq. (24)  $\rightarrow \mathcal{L}_{kl}(P, Q)$  by Eq. (23).
20:     Compute  $\hat{X}_b$  by Eq. (6)  $\rightarrow \mathcal{L}_{rec}(X_b, \hat{X}_b)$  by Eq. (7).
21:     Compute  $e_b, \bar{e}_b$  by Eq. (9)  $\rightarrow \mathcal{L}_{clr}(e_b, \bar{e}_b)$  by Eq. (10).
22:      $\mathcal{L}_2 = \text{UWL}(\mathcal{L}_{kl}, \mathcal{L}_{rec}, \mathcal{L}_{clr})$ .
23:     Update  $\Theta$  and  $\Pi(\mathcal{M}), \Pi(\mathcal{C}), \Pi(\mathcal{H})$  using  $\mathcal{L}_2$ .
24:   end for
25: end while
26: return  $Z, Q$ 

```

is substituted with a fully-connected linear decoder \mathcal{M}_D . For any given image X_i , the regenerated image \hat{X}_i is as follows:

$$\hat{X}_i = \mathcal{M}_D(\mathcal{M}_E(X_i, \mathbf{W}_E), \mathbf{W}_D) \quad (6)$$

The MIM branch loss for the current batch are:

$$\mathcal{L}_{rec} = \frac{1}{N_b} \sum_{i=1}^{N_b} \frac{1}{N_{masked}} \sum_{j \in S_{masked}} (\mathbf{p}_{i,j} - \hat{\mathbf{p}}_{i,j})^T (\mathbf{p}_{i,j} - \hat{\mathbf{p}}_{i,j}), \quad (7)$$

where N_b is the batch size, \mathbf{W}_E and \mathbf{W}_D represent the parameters of the encoder and decoder, respectively. N_{masked} denotes the number of masked patches, S_{masked} the set of masked patches. $\mathbf{p}_{i,j}$ and $\hat{\mathbf{p}}_{i,j}$ represent the original and regenerated j -th image patch of X_i , respectively.

For the contrastive branch \mathcal{C} , the N_b raw images form positive pairs with their respective smoothed images, and negative pairs with the other $2N_b - 2$ images in the same batch. \mathcal{C} is structured around a pseudo-siamese network with two encoders: an online encoder \mathcal{C}_O and a target momentum encoder \mathcal{C}_T . Both encoders

share the identical network architecture as \mathcal{P}_E , with \mathcal{C}_O and \mathcal{P}_E having tied parameters. The parameters of \mathcal{C}_T are updated using EMA. Concretely, let $\tilde{\mathbf{W}}_E$ and $\bar{\mathbf{W}}_E$ denote the parameters of the \mathcal{C}_O and \mathcal{C}_T , respectively. Then we have:

$$\tilde{\mathbf{W}}_E = \mathbf{W}_E, \bar{\mathbf{W}}_E = m\bar{\mathbf{W}}_E + (1 - m)\tilde{\mathbf{W}}_E \quad (8)$$

Here, m represents the momentum, fixed at 0.999. For each image X_i and its augmented counterpart \bar{X}_i , their respective embedding vectors, \mathbf{e}_i and $\bar{\mathbf{e}}_i \in \mathbb{R}^D$, are obtained as: $\mathbf{e}_i = \tilde{g}(\mathcal{C}_O(X_i, \tilde{\mathbf{W}}_E))$, $\bar{\mathbf{e}}_i = \bar{g}(\mathcal{C}_T(\bar{X}_i, \bar{\mathbf{W}}_E))$,

$$\mathbf{e}_i = \tilde{g}(\mathcal{C}_O(X_i, \tilde{\mathbf{W}}_E)), \bar{\mathbf{e}}_i = \bar{g}(\mathcal{C}_T(\bar{X}_i, \bar{\mathbf{W}}_E)), \quad (9)$$

where \tilde{g} and \bar{g} are linear mapping functions with trainable weights, and the weights of \bar{g} are updated using EMA as well. The contrastive loss $\mathcal{L}_{clr,i}$ is computed as :

$$\mathcal{L}_{clr,i} = -\log \frac{s(\mathbf{e}_i, \bar{\mathbf{e}}_i)}{\sum_{k=1, k \neq i}^{N_b} s(\mathbf{e}_i, \mathbf{e}_k) + \sum_{k=1}^{N_b} s(\mathbf{e}_i, \bar{\mathbf{e}}_k)} - \log \frac{s(\bar{\mathbf{e}}_i, \mathbf{e}_i)}{\sum_{k=1}^{N_b} s(\bar{\mathbf{e}}_i, \mathbf{e}_k) + \sum_{k=1, k \neq i}^{N_b} s(\bar{\mathbf{e}}_i, \bar{\mathbf{e}}_k)}, \quad (10)$$

where $s(\cdot, \cdot) = \exp(\cos(\cdot, \cdot)/\tau)$, and τ is a temperature coefficient, defaulting to 0.5. Consequently, the loss function \mathcal{L}_{clr} is defined as $\mathcal{L}_{clr} = \frac{1}{N_b} \sum_{i=1}^{N_b} \mathcal{L}_{clr,i}$. Finally, \mathcal{L}_{rec} and \mathcal{L}_{clr} are dynamically integrated into the total SSL loss function \mathcal{L}_{ssl} using the uncertain weights loss (UWL) function [26]:

$$\begin{aligned} \mathcal{L}_{ssl} &= \text{UWL}(\mathcal{L}_{rec}, \mathcal{L}_{clr}) \\ &= \frac{1}{2\sigma_1^2} \mathcal{L}_{rec} + \frac{1}{2\sigma_2^2} \mathcal{L}_{clr} + \log(1 + \sigma_1^2) + \log(1 + \sigma_2^2), \end{aligned} \quad (11)$$

where σ_1 and σ_2 are trainable noise parameters.

3.3 Self-paced Deep Image Clustering (Module II)

3.3.1 Student's t mixture model. Let \mathbf{e}_i denote the Module I-generated embedding vector for the i -th original image. We first map \mathbf{e}_i to $\mathbf{z}_i \in \mathbb{R}^D$ in a latent space wherein it is more conformed to a t -distribution. This mapping is achieved through a nonlinear projection head with batch normalization and scaled exponential linear unit (SELU) activation function:

$$\mathbf{z}_i = \text{SELU}(\text{BN}(\mathbf{W}_p, \mathbf{e}_i)), \quad (12)$$

The set of these vectors, $Z = \{\mathbf{z}_i\}_{i=1}^N \in \mathbb{R}^{N \times D}$, is modeled using an SMM, whose components correspond to image clusters. Since extreme values are downweighed by SMM during parameter inference, this clustering is more robust to outliers and variances. The SMM is parameterized by $\Theta = \{\theta_k, \pi_k, \mu_k, \Sigma_k, v_k, \forall k \in K\}$, where K represents the number of components and is assumed to be known or can be automatically inferred (see Section 4.1). Here, $\pi_k, \mu_k, \Sigma_k, v_k$ denote the weight, mean, covariance matrix, and degree of freedom of the k -th component, respectively. The density function of \mathbf{z}_i is expressed as:

$$p(\mathbf{z}_i|\Theta) = \sum_{k=1}^K \pi_k \phi(\mathbf{z}_i|\mu_k, \Sigma_k, v_k) \quad (13)$$

For robust model inference, we use the maximum a posterior-expectation maximization (MAP-EM) algorithm by applying a conjugate Dirichlet prior on $\Pi = \{\pi_k, \forall k \in [1, K]\}$ and a normal inverse Wishart (NIW) prior on μ_k, Σ_k :

$$\Pi \sim \text{Dir}(\Pi|\alpha^0), \quad (14)$$

$$\mu_k, \Sigma_k \sim \text{NIW}(\mu_k, \Sigma_k | m_0, \kappa_0, S_0, \rho_0), \forall k \in [1, K] \quad (15)$$

To simplify the inference, we rewrite the Student's t density function ϕ as a Gaussian scale mixture by introducing an "artificial" hidden variable $\zeta_{i,k}$, $\forall i \in [1, N]$, $\forall k \in [1, K]$ that follows a Gamma distribution parameterized by v_k :

$$\phi(\mathbf{z}_i | \mu_k, \Sigma_k, v_k) = \int \mathcal{N}\left(\mathbf{z}_i | \mu_k, \frac{\Sigma_k}{\zeta_{i,k}}\right) \Gamma\left(\zeta_{i,k} \left| \frac{v_k}{2}, \frac{v_k}{2} \right.\right) d\zeta_{i,k} \quad (16)$$

We further introduce a missing variable ξ_i to represent the component membership of \mathbf{z}_i . The posterior complete data log likelihood is then expressed as:

$$\begin{aligned} \ell_c(\Theta) &= \log P(\mathbf{Z}, \zeta, \xi | \Theta) \\ &= \log \text{Dir}(\Pi | \alpha^0) + \\ &\quad \sum_k \log \text{NIW}(\mu_k, \Sigma_k | m_0, \kappa_0, S_0, \rho_0) + \\ &\quad \sum_i \sum_k [II(\xi_i = k)(\log \pi_k + \log \Phi(\mathbf{z}_i, \zeta_{i,k} | \mu_k, \Sigma_k, v_k))] \end{aligned} \quad (17)$$

In the t -th iteration of the E-step, the expected sufficient statistics $\overline{\xi_{i,k}}^{(t)}$ and $\overline{\zeta_{i,k}}^{(t)}$ are derived based on $\Theta^{(t-1)}$. In the subsequent M-step, $\Theta^{(t-1)}$ is updated to $\Theta^{(t)}$ by maximizing the auxiliary function $Q(\Theta, \Theta^{(t-1)}) = E(\ell_c(\Theta) | \Theta^{(t-1)})$. These two steps are alternated until either convergence is reached or a predefined maximum number of iterations is attained. Refer to Supplementary 1.1 for details of the model inference.

3.3.2 Self-paced Joint Optimization of Image Embeddings and Cluster Assignments. Two loss functions, \mathcal{L}_1 and \mathcal{L}_2 , are calculated based on clustering results for updating parameters of both *Module I* and the SMM through loss gradient backpropagation. This iterative process progressively improves the clustering-oriented image embeddings and clustering results. Upon completing the inference of SMM parameters $\tilde{\Theta}$ in each epoch, an epoch-level loss \mathcal{L}_1 is calculated for updating parameters of *Module I*:

$$\mathcal{L}_1 = \eta \mathcal{L}_{lap} + (1 - \eta) \text{UWL}(-\mathcal{L}_{\ell\ell}, -\mathcal{L}_{size}, \mathcal{L}_{rec}). \quad (18)$$

Here, \mathcal{L}_{lap} is a Laplacian regularization term that promotes the similarities among image embeddings \mathbf{Z} to be consistent with a seeding image-image similarity matrix \mathcal{S} , informing the initial training phase. The derivation of \mathcal{S} is detailed in Supplementary 1.2. \mathcal{L}_{lap} is defined as follows:

$$\mathcal{L}_{lap} = \text{Tr}\left(\mathbf{Z}^T \left(I - \mathcal{D}^{-\frac{1}{2}} \mathcal{S} \mathcal{D}^{-\frac{1}{2}}\right) \mathbf{Z}\right), \quad (19)$$

where \mathcal{D} is the degree matrix of \mathcal{S} , and η , initially set at 0.5, decays over the training course so that the influence of \mathcal{S} is gradually reduced. $\mathcal{L}_{\ell\ell}$ represents the log likelihood of the embeddings given

the estimated SMM parameters $\tilde{\Theta}$:

$$\mathcal{L}_{\ell\ell} = \sum_{i=1}^N \log \left[\sum_k q_{i,k} \right], \quad (20)$$

$$q_{i,k} = \pi_k \phi(\mathbf{z}_i | \mu_k, \Sigma_k, v_k), \forall i \in [1, N], \forall k \in [1, K]. \quad (21)$$

\mathcal{L}_{size} penalizes empty and tiny clusters, while exempting those whose size exceeds a predefined threshold v so that image assignments is not overly uniform:

$$\mathcal{L}_{size} = \sum_{k=1}^K -J_k \log J_k, J_k = \begin{cases} \frac{\sum_i q_{i,k}}{N} & , \text{if } J_k \leq v \\ 1 & .\text{otherwise} \end{cases} \quad (22)$$

\mathcal{L}_{rec} , defined in Equation 7, aims to enhance the local-context awareness of embeddings. Subsequently, within the same epoch, a batch-level loss $\mathcal{L}_2 = \text{UWL}(\mathcal{L}_{kl}, \mathcal{L}_{rec}, \mathcal{L}_{clr})$ is utilized to update *Module I* and SMM parameters across successive batches. Here, \mathcal{L}_{rec} and \mathcal{L}_{clr} remains same as in Equations 7 and ?? except being calculated on the batch-level. \mathcal{L}_{kl} boosts high-confidence images, incrementally grouping similar instances while separating dissimilar ones:

$$\mathcal{L}_{kl} = \text{KL}(\mathcal{P} | \mathcal{Q}) = \sum_i \sum_j \mathbb{P}_{i,j} \log \frac{\mathbb{P}_{i,j}}{\mathbb{Q}_{i,j}}, \quad (23)$$

$$\text{where } \mathbb{Q}_{i,k} = \frac{q_{i,k}}{\sum_c q_{i,c}}, \mathbb{P}_{i,k} = \frac{\mathbb{Q}_{i,k}^2 / \sum_i \mathbb{Q}_{i,c}}{\sum_c (\mathbb{Q}_{i,c}^2 / \sum_i \mathbb{Q}_{i,c})} \quad (24)$$

Here, $q_{i,k}$ is same as in Equation 21, $\mathbb{Q}_{i,k}$ represents the probability of assigning i -th image to the k -th SMM component, and $\mathbb{P}_{i,k}$ an auxiliary target distribution that boosts up high-confidence images. After this joint optimization, the training progresses to the next epoch, iterating until the end of the training process. The mathematical derivations of gradients of \mathcal{L}_1 and \mathcal{L}_2 with respect to \mathbf{W}_E , \mathbf{W}_D and Θ are detailed in Supplementary 1.3.

4 EXPERIMENTS

4.1 Experimental Settings

4.1.1 Datasets. To comprehensively evaluate *DARLC*, we utilize 12 real ST datasets derived from different tissue slices of the human dorsolateral prefrontal cortex (hDLPFC). These datasets display spatial expression levels of the whole genome across the hDLPFC tissue (approximately 18700 75x75 SGEF images per dataset). These datasets are named by tissue slice numbers (151507-151510, 151669-151676), and accessible through the spatialLIBD package [31] at <http://spatial.libd.org/spatialLIBD>.

4.1.2 Data quality control and preprocessing. We conform to the conventional procedure for preprocessing ST data, as implemented in the SCANPY package [40]. Specifically, we first remove mitochondrial and External RNA Controls Consortium spike-in genes. Then, genes detected in fewer than 10 spots are excluded. To preserve the spatial data integrity, we do not perform quality control on spatial spots. Finally, the gene expression counts are normalized by library size, followed by log-transformation.

Table 1: Clustering performance of *DARLC* and the benchmark methods across 12 datasets of spatial gene expression images are quantified using DBIE and DBIP scores. Lower scores indicate better performance, the best score for each dataset is bolded, and the second-best score is underlined. The score standard deviation is subscripted.

Method	DBIE↓											
	151507	151508	151509	151510	151669	151670	151671	151672	151673	151674	151675	151676
DEC	10.88 _{0.11}	<u>10.59</u> _{0.31}	11.54 _{0.14}	10.88 _{0.55}	13.23 _{1.87}	11.15 _{0.55}	12.32 _{0.97}	11.05 _{0.59}	11.58 _{1.20}	11.22 _{1.08}	11.44 _{0.70}	11.42 _{0.65}
DAGMM	46.18 _{27.97}	47.42 _{28.27}	30.70 _{18.39}	19.02 _{8.21}	30.60 _{21.35}	24.81 _{7.74}	46.71 _{42.62}	54.80 _{42.75}	26.07 _{5.72}	39.47 _{12.57}	20.64 _{17.48}	42.83 _{19.63}
EDESC	12.55 _{0.95}	14.24 _{1.39}	11.81 _{0.97}	10.79 _{0.88}	12.48 _{0.50}	<u>10.71</u> _{1.22}	11.94 _{1.08}	13.53 _{1.53}	<u>11.18</u> _{0.99}	10.17 _{1.70}	<u>9.94</u> _{1.44}	<u>10.04</u> _{1.12}
IDCEC	15.65 _{1.73}	14.09 _{0.88}	13.55 _{0.90}	14.35 _{1.31}	<u>11.18</u> _{0.13}	11.85 _{0.14}	<u>10.51</u> _{0.34}	10.83 _{0.49}	11.23 _{0.50}	10.60 _{0.34}	12.30 _{0.94}	12.95 _{1.10}
CC	16.66 _{0.02}	18.66 _{0.01}	20.03 _{0.04}	16.38 _{0.02}	18.31 _{0.04}	18.74 _{0.04}	17.05 _{0.03}	19.05 _{0.05}	18.77 _{0.04}	19.53 _{0.05}	20.26 _{0.02}	17.41 _{0.03}
DCP	12.48 _{0.47}	12.07 _{0.27}	<u>11.07</u> _{0.32}	11.65 _{0.24}	11.45 _{0.32}	11.73 _{0.10}	11.61 _{0.11}	12.28 _{0.39}	11.92 _{0.27}	11.33 _{0.86}	12.09 _{0.66}	11.27 _{0.03}
CVCL	18.04 _{7.64}	33.14 _{3.16}	20.24 _{4.30}	28.77 _{1.70}	33.38 _{4.85}	55.02 _{11.21}	39.52 _{11.33}	31.00 _{0.51}	30.77 _{1.70}	31.15 _{3.20}	39.85 _{7.30}	30.96 _{2.35}
iBOT-C	<u>10.75</u> _{0.01}	10.88 _{0.02}	11.87 _{0.01}	<u>10.23</u> _{0.03}	13.15 _{0.03}	13.21 _{0.02}	12.69 _{0.04}	<u>10.76</u> _{0.05}	12.81 _{0.02}	<u>10.11</u> _{0.01}	10.71 _{0.03}	11.31 _{0.01}
<i>DARLC</i>	8.09 _{0.33}	8.16 _{0.13}	8.31 _{0.28}	7.98 _{0.74}	7.69 _{0.16}	8.22 _{0.31}	7.65 _{0.40}	8.06 _{0.37}	7.90 _{0.37}	7.78 _{0.33}	7.41 _{0.36}	8.09 _{0.55}

Method	DBIP↓											
	151507	151508	151509	151510	151669	151670	151671	151672	151673	151674	151675	151676
DEC	<u>2.38</u> _{0.06}	<u>2.18</u> _{0.02}	2.68 _{0.21}	<u>2.44</u> _{0.11}	2.55 _{0.10}	<u>2.49</u> _{0.09}	3.04 _{0.24}	<u>2.52</u> _{0.08}	3.71 _{0.53}	3.32 _{0.19}	3.52 _{0.59}	<u>2.42</u> _{0.08}
DAGMM	16.19 _{5.26}	14.01 _{4.23}	12.35 _{0.98}	12.11 _{2.09}	15.65 _{1.38}	13.51 _{3.34}	31.03 _{18.08}	22.75 _{12.10}	27.17 _{13.81}	20.06 _{2.27}	16.14 _{13.59}	20.14 _{9.84}
EDESC	2.49 _{0.17}	2.66 _{0.28}	2.68 _{0.32}	2.57 _{0.20}	2.68 _{0.36}	2.81 _{0.41}	2.70 _{0.36}	2.66 _{0.38}	<u>2.85</u> _{0.40}	3.11 _{0.14}	<u>2.75</u> _{0.30}	2.77 _{0.30}
IDCEC	2.70 _{0.05}	2.52 _{0.11}	2.75 _{0.07}	2.66 _{0.06}	2.60 _{0.28}	2.65 _{0.28}	2.76 _{0.26}	2.69 _{0.32}	3.13 _{0.23}	2.99 _{0.20}	2.98 _{0.26}	3.07 _{0.19}
CC	3.18 _{0.01}	3.12 _{0.01}	4.28 _{0.01}	4.07 _{0.03}	3.13 _{0.04}	3.14 _{0.01}	4.12 _{0.02}	4.35 _{0.01}	3.18 _{0.00}	3.58 _{0.00}	2.93 _{0.01}	3.96 _{0.04}
DCP	2.46 _{0.11}	2.29 _{0.03}	<u>2.54</u> _{0.12}	2.46 _{0.16}	<u>2.55</u> _{0.06}	2.54 _{0.10}	<u>2.59</u> _{0.06}	2.55 _{0.06}	3.37 _{0.03}	<u>2.96</u> _{0.16}	3.00 _{0.13}	2.87 _{0.16}
CVCL	5.28 _{1.35}	5.68 _{0.65}	5.28 _{0.70}	5.72 _{0.76}	7.00 _{2.07}	8.26 _{2.00}	8.92 _{2.02}	6.69 _{0.81}	7.13 _{0.84}	5.45 _{0.63}	6.32 _{1.07}	4.97 _{0.81}
iBOT-C	3.39 _{0.02}	3.62 _{0.04}	4.77 _{0.01}	3.27 _{0.03}	4.90 _{0.04}	3.45 _{0.02}	3.48 _{0.03}	2.96 _{0.02}	4.87 _{0.01}	3.38 _{0.01}	3.57 _{0.03}	3.47 _{0.02}
<i>DARLC</i>	2.08 _{0.02}	2.04 _{0.01}	2.11 _{0.03}	2.10 _{0.01}	2.22 _{0.05}	2.24 _{0.04}	2.24 _{0.06}	2.19 _{0.03}	2.23 _{0.00}	2.14 _{0.08}	2.45 _{0.11}	2.11 _{0.20}

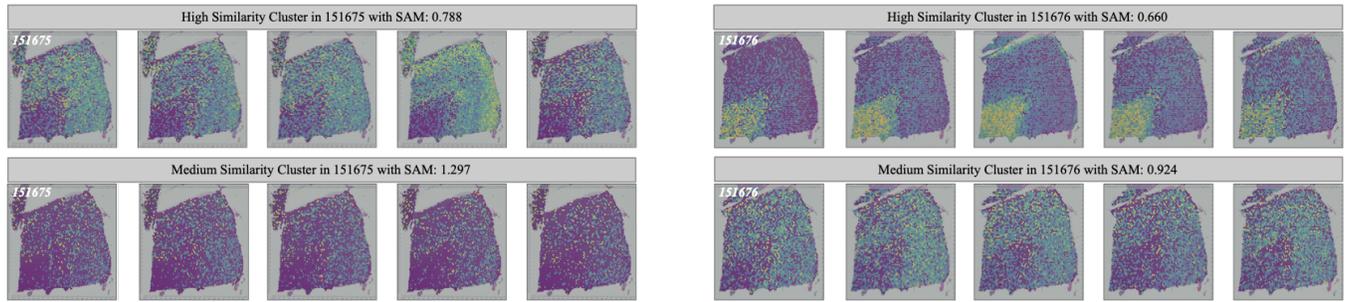


Figure 3: SGEPs from clusters generated by *DARLC* in dataset (151675, 151676) with high and medium intra-cluster similarity.

4.1.3 Cluster Number Inference. Given the number of image clusters is not known a priori, *DARLC* can estimate this number using a seeding similarity matrix $\mathcal{S} \in \mathbb{R}^{N \times N}$, where N represents the number of images [44]. \mathcal{S} is transformed to a graph Laplacian matrix (refer to Supplementary 1.2), $\mathbb{L} \in \mathbb{R}^{N \times N}$ as follows:

$$\mathcal{S}' = \mathcal{S} + \mathcal{S}^2, \quad (25)$$

$$\mathbb{L} = \mathbb{D}^{-\frac{1}{2}} \mathcal{S}' \mathbb{D}^{-\frac{1}{2}}. \quad (26)$$

Here, $\mathbb{D} \in \mathbb{R}^{N \times N}$ represents the degree matrix of \mathcal{S} , and \mathcal{S}' aims to enhance the similarity structure. The eigenvalues of \mathbb{L} are then ranked as $\lambda_{(1)} \leq \lambda_{(2)} \leq \dots \leq \lambda_{(N)}$. The number of clusters, denoted as K , is inferred as:

$$K = \operatorname{argmax}_i \{ \lambda_{(i)} - \lambda_{(i-1)} \}, \quad i = 2, 3, \dots, N. \quad (27)$$

Table 2: Gene-gene interaction prediction results based on the three types of gene image embeddings. The superior method is bolded.

Case	ACC (%)					
	151671	151672	151673	151674	151675	151676
iBOT	66.58	68.73	68.53	65.88	68.03	67.48
DARLC-C1	76.18	77.20	74.71	73.64	77.85	71.50
DARLC-full	78.11	78.97	77.66	78.74	78.65	73.59

4.1.4 Baselines. The image clustering benchmark methods include two classic deep clustering method, DEC [41] and DAGMM [49], as well as five SOTA methods that either improve DEC (e.g., EDESC [3] and IDCEC [29]) or incorporate CL (e.g., CC [25], DCP [27], and CVCL [7]). In addition, we include a benchmark method consisting of iBOT [47], which integrates CL and MIM for representation learning, and a boosted GMM for clustering, denoted as iBOT-C.

4.1.5 Implementation Details. The GAT for data augmentation adopts an encoder including a single attention head with C-512-30 network structure, and a symmetric decoder. In *Module I*, the shared encoder structure is a ViT comprising four transformer blocks, each having four attention heads, for processing 75×75 input images segmented into 4×4 patches in our case. The MIM decoder follows a D-128-256-512-1024-75*75 residual network. The *Module I* is pre-trained for 50 epochs with a learning rate of 0.001. The nonlinear projection head that bridges *Module I* and *II* is a two-layer MLP for normalizing image representations to a dimension size of 32. The iterative joint optimization of representation learning and clustering continues for 50 epochs using Adam optimizer. Given the absence of ground truth in gene cluster labels, we heuristically determine the number of gene clusters for our experiments to achieve an average cluster size of 30 genes, approximating the typical size of a gene pathway [46].

4.1.6 Evaluation Metrics. Without ground truth cluster labels, we evaluate the clustering results using the Davies-Bouldin index (DBI) metric [10]:

$$DBI = \frac{1}{K} \sum_{i=1}^K \max_{i \neq j} \frac{d_i + d_j}{d_{(i,j)}}, \quad d_i = \frac{|C_i|}{|C_i|} \sum_{j=1}^{|C_i|} \delta_{c_i,j}, \quad (28)$$

where K is the number of clusters, C_i the samples in cluster i , $\delta_{i,j}$ the distance between instances i and j , c_i the centroid of cluster i . Cluster width d_i is the mean intra-cluster distance to c_i , and $d_{(i,j)} = \delta_{c_i,c_j}$ measures the distance between clusters i and j . DBI quantifies the clustering efficiency by measuring the ratio of intra-cluster compactness to inter-cluster separation, with lower scores indicating better clustering. To evaluate clustering from different perspectives, we use two DBI metrics, DBIP and DBIE, based on Pearson and Euclidean distances, respectively [34].

Table 3: Spatial cofunctional gene clustering results using three image embeddings, with the best method in bolded

Case	NMI (%)					
	151671	151672	151673	151674	151675	151676
iBOT	20.56	15.98	15.41	27.05	23.17	25.62
DARLC-C1	66.81	67.81	70.91	65.13	74.62	75.85
DARLC-full	78.11	71.28	79.23	74.34	75.20	75.91
Case	ARI (%)					
	151671	151672	151673	151674	151675	151676
iBOT	-2.66	-5.06	-5.64	4.29	-0.80	2.92
DARLC-C1	53.41	51.61	55.58	45.88	48.97	59.83
DARLC-full	59.44	53.02	65.82	36.25	59.83	54.50

4.2 Clustering Sparse, Noisy Images of Spatial Gene Expressions

Table 1 showcase the performance of *DARLC*, compared to eight benchmark methods, in clustering gene spatial expression images of 12 real ST datasets, evaluated by the DBIE and DBIP scores, respectively. For each dataset, the experiment is repeated ten times to obtain the mean and standard deviation of each method's scores. *DARLC* consistently achieves the lowest scores in both DBIE and DBIP across all datasets, highlighting its superiority in generating clusters consisting of spatially similar and coherent images. This superiority can be attributed to *DARLC*'s features in integrating MIM and CL, generating more plausible augmented data, and the robust and adaptable clustering algorithm, as substantiated in our ablation study. In contrast, benchmark methods adopt varied strategies: IDCEC and EDESC leverage a convolutional autoencoder for extracting visual features; iBOT+boosted GMM adopts conventional data augmentation; CC, CVCL and DCP rely solely on CL for representation learning; DAGMM employs an outlier-sensitive GMM for clustering. However, these methods generally demonstrate unstable and suboptimal performance. Overall, compared to the best-performing benchmark method, *DARLC* achieves an average reduction of approximately 24.89% in DBIE and 14.39% in DBIP across all datasets. Finally, *DARLC* clusters are divided into high and medium-quality groups using spectral angle mapper (SAM) metric scores [22], with lower SAM indicating greater intra-cluster similarity. To provide a visual illustration of *DARLC*'s clustering performance, we randomly select one cluster from both the high- and medium-quality groups within each of the 151675 and 151676 datasets. From each of the selected clusters, we then randomly select five genes to be displayed in Figure 3. Figure 3 clearly demonstrates that *DARLC* effectively groups images with similar expression patterns into the same cluster.

4.3 Evaluating *DARLC*-generated Gene Image Representations

In this section, we present a comprehensive evaluation of gene image representations produced by the fully implemented *DARLC*

model (denoted as *DARLC*-full). First, we assess whether representations generated by *DARLC*-full capture corresponding biosemantics, particularly through pathway enrichment analysis, compared to original gene expressions. Furthermore, we extend our investigation to specific critical downstream tasks: predicting interactions between genes and clustering genes based on spatial cofunctionality. These evaluations are conducted across six distinct datasets (151671-151676). Additionally, gene image embeddings generated by iBOT and a variant of *DARLC* (denoted as *DARLC*-C1), which is deprived of *Module II*, serve as baselines.

4.3.1 Gene-gene Interaction Prediction. We employ an MLP-based classifier for predicting gene pair interactions. This is achieved by linear probing using gene image representations generated by *DARLC* and baseline methods (see Supplementary 1.4). We follow the methodology in [15], which is based on the Gene Ontology, to acquire the gene-gene interaction ground truth. Theoretically, gene image representations with richer semantic meanings should yield more accurate predictions. As shown by the accuracy scores in Table 2, the classifier yields the most accurate predictions ($77.62\% \pm 1.85\%$) using embeddings generated by *DARLC*-full, and the second most accurate predictions ($75.18\% \pm 2.17\%$) using embeddings generated by *DARLC*-C1, followed by the predictions using iBOT-generated embeddings ($67.54\% \pm 1.03\%$).

4.3.2 Spatially Cofunctional Gene Clustering. Spatially cofunctional genes are those belonging to the same gene family and exhibit similar spatial expression patterns [11]. Their family identities can serve as labels to evaluate the quality of gene image embeddings via clustering. Specifically, our evaluation involves five spatially cofunctional genes from each of the HLA, GABR, RPL, and MT gene families (see Supplementary Figure 1). The Leiden algorithm [36] is used to cluster gene image embeddings generated by *DARLC*-full and the baseline methods. The clustering results, evaluated using the normalized mutual information (NMI) and adjusted rand index (ARI) scores, as shown in Table 3, demonstrate that Leiden yields the most accurate clustering with *DARLC*-full and the second most accurate with *DARLC*-C1.

In summary, these results collectively highlight the effectiveness of the joint clustering (i.e., *DARLC*-full surpasses *DARLC*-C1) and GAT-based data augmentation (i.e., *DARLC*-C1 surpasses iBOT) in enhancing the quality of gene image representations.

4.4 Ablation Study

Here, we conduct a series of ablation studies on six ST datasets (151671-151676) to investigate the contributions of *DARLC*'s components in image clustering. The results, detailed in DBIE and DBIP scores, are presented in Table 4. Notably, *DARLC*'s performance declines most with the CL branch removal ("w/o CLR"), followed by the elimination of *Module II* ("w/o SMM") and the substitution of the robust SMM with an outlier-sensitive GMM ("SMM \rightarrow GMM"). We also observed that employing traditional image smoothing methods such as Gaussian Kernel Smoothing (GKS) instead of GAT ("GAT \rightarrow GKS") on SNIs leads to a decline in *DARLC*'s performance. Additionally, removing either \mathcal{L}_{lap} ("w/o \mathcal{L}_{lap} ") or \mathcal{L}_{size} ("w/o \mathcal{L}_{size} ") from the optimization decreases *DARLC*'s performance, indicated

Table 4: Ablation study results across six datasets for components in *Module I & II* and regularization terms. The best result is bolded.

Case	DBIE↓					
	151671	151672	151673	151674	151675	151676
w/o CLR	14.35	16.99	8.70	9.23	10.36	15.12
GAT \rightarrow GKS	9.35	10.37	8.98	9.08	8.90	8.95
w/o SMM	11.84	12.19	11.33	11.81	11.49	12.35
SMM \rightarrow GMM	11.49	11.36	11.16	11.31	11.05	11.37
w/o \mathcal{L}_{lap}	8.32	10.76	9.02	8.10	9.22	8.12
w/o \mathcal{L}_{size}	10.38	8.60	8.11	9.34	8.00	8.28
w/o Pretraining	8.22	8.74	8.38	8.18	7.87	9.09
<i>DARLC</i> -full	7.65	8.06	7.90	7.78	7.41	8.09

Case	DBIP↓					
	151671	151672	151673	151674	151675	151676
w/o CLR	4.21	4.21	3.51	3.97	3.82	3.69
GAT \rightarrow GKS	2.35	2.29	2.62	2.61	2.63	2.37
w/o SMM	2.49	2.35	2.29	2.66	2.66	2.36
SMM \rightarrow GMM	2.53	2.26	2.31	2.51	2.84	2.26
w/o \mathcal{L}_{lap}	2.33	2.27	2.26	2.31	2.66	2.17
w/o \mathcal{L}_{size}	2.29	2.22	2.25	2.25	2.31	2.19
w/o Pretraining	2.36	2.40	2.30	2.65	2.53	2.42
<i>DARLC</i> -full	2.24	2.19	2.23	2.14	2.45	2.11

by higher DBIE and DBIP scores. Lastly, "w/o Pretraining" shows *DARLC*'s performance at the same clustering training epoch as the complete model but without the initial "warm up" pretraining of *Module I*. The relative underperformance in the "w/o Pretraining" scenario suggests a slower training convergence compared to the complete model.

Overall, all key components of *DARLC* have proven effective, particularly the CL module. This is due to: a) The discontinuous and sparse nature of SNIs, which greatly benefits from the enhanced discriminability provided by CL. b) CL's use of smoothed images as augmented data, which serve as effective priors for guiding *DARLC* to discern the primary visual patterns of SNIs.

5 CONCLUSION

In this study, we introduce *DARLC*, a novel algorithm for joint learning of representations and cluster assignments for SNIs. *DARLC* features in its enhanced data augmentation technique, comprehensive and potent representation learning approach that integrates MIM, CL and clustering, as well as robust and adaptable clustering algorithm. These features collectively contribute to *DARLC*'s superiority in both image representation learning and clustering, as evidenced by our extensive benchmarks over multiple real datasets and comprehensive ablation studies.

REFERENCES

- [1] Salar Askari. 2021. Fuzzy C-Means clustering algorithm for data with unequal cluster sizes and contaminated with noise and outliers: Review and development. *Expert Systems with Applications* 165 (2021), 113856.
- [2] Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. 2021. Beit: Bert pre-training of image transformers. *arXiv preprint arXiv:2106.08254* (2021).
- [3] Jinyu Cai, Jicong Fan, Wenzhong Guo, Shiping Wang, Yunhe Zhang, and Zhao Zhang. 2022. Efficient deep embedded subspace clustering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 1–10.
- [4] Tiffany Tianhui Cai, Jonathan Frankle, David J Schwab, and Ari S Morcos. 2020. Are all negatives created equal in contrastive instance discrimination? *arXiv preprint arXiv:2010.06682* (2020).
- [5] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. 2020. Unsupervised learning of visual features by contrasting cluster assignments. *Advances in neural information processing systems* 33 (2020), 9912–9924.
- [6] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. 2021. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*. 9650–9660.
- [7] Jie Chen, Hua Mao, Wai Lok Woo, and Xi Peng. 2023. Deep Multiview Clustering by Contrasting Cluster Assignments. *arXiv preprint arXiv:2304.10769* (2023).
- [8] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*. PMLR, 1597–1607.
- [9] Xinlei Chen and Kaiming He. 2021. Exploring simple siamese representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 15750–15758.
- [10] David L Davies and Donald W Bouldin. 1979. A cluster separation measure. *IEEE transactions on pattern analysis and machine intelligence* 2 (1979), 224–227.
- [11] Jeffery P Demuth, Tijl De Bie, Jason E Stajich, Nello Cristianini, and Matthew W Hahn. 2006. The evolution of mammalian gene families. *PLoS one* 1, 1 (2006), e85.
- [12] Julien Denize, Jaonary Rabarisoa, Astrid Orcesi, Romain Hérault, and Stéphane Canu. 2023. Similarity contrastive estimation for self-supervised soft contrastive learning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 2706–2716.
- [13] Kangning Dong and Shihua Zhang. 2022. Deciphering spatial domains from spatially resolved transcriptomics with an adaptive graph attention auto-encoder. *Nature communications* 13, 1 (2022), 1739.
- [14] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiuhua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929* (2020).
- [15] Jingcheng Du, Peilin Jia, Yulin Dai, Cui Tao, Zhongming Zhao, and Degui Zhi. 2019. Gene2vec: distributed representation of genes based on co-expression. *BMC genomics* 20 (2019), 7–15.
- [16] Peng Gao, Teli Ma, Hongsheng Li, Ziyi Lin, Jifeng Dai, and Yu Qiao. 2022. Convmae: Masked convolution meets masked autoencoders. *arXiv preprint arXiv:2205.03892* (2022).
- [17] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. 2020. Bootstrapping your own latent—a new approach to self-supervised learning. *Advances in neural information processing systems* 33 (2020), 21271–21284.
- [18] Xifeng Guo, Long Gao, Xinwang Liu, and Jianping Yin. 2017. Improved deep embedded clustering with local structure preservation. In *Ijcai*, Vol. 17. 1753–1759.
- [19] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. 2022. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 16000–16009.
- [20] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. 2020. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 9729–9738.
- [21] Zhicheng Huang, Xiaojie Jin, Chengze Lu, Qibin Hou, Ming-Ming Cheng, Dongmei Fu, Xiaohui Shen, and Jiashi Feng. 2023. Contrastive masked autoencoders are stronger vision learners. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2023).
- [22] Fred A Kruse, AB Lefkoff, JW Boardman, KB Heidebrecht, AT Shapiro, PJ Barloon, and AFH Goetz. 1993. The spectral image processing system (SIPS)—interactive visualization and analysis of imaging spectrometer data. *Remote sensing of environment* 44, 2-3 (1993), 145–163.
- [23] Fengfu Li, Hong Qiao, and Bo Zhang. 2018. Discriminatively boosted image clustering with fully convolutional auto-encoders. *Pattern Recognition* 83 (2018), 161–173.
- [24] Junnan Li, Pan Zhou, Caiming Xiong, and Steven CH Hoi. 2020. Prototypical contrastive learning of unsupervised representations. *arXiv preprint arXiv:2005.04966* (2020).
- [25] Yunfan Li, Peng Hu, Zitao Liu, Dezhong Peng, Joey Tianyi Zhou, and Xi Peng. 2021. Contrastive clustering. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 35. 8547–8555.
- [26] Lukas Liebel and Marco Körner. 2018. Auxiliary tasks in multi-task learning. *arXiv preprint arXiv:1805.06334* (2018).
- [27] Yijie Lin, Yuanbiao Gou, Xiaotian Liu, Jinfeng Bai, Jiancheng Lv, and Xi Peng. 2022. Dual contrastive prediction for incomplete multi-view representation learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45, 4 (2022), 4447–4461.
- [28] Yahui Long, Kok Siong Ang, Mengwei Li, Kian Long Kelvin Chong, Raman Sethi, Chengwei Zhong, Hang Xu, Zhiwei Ong, Karishma Sachaphibulkij, Ao Chen, et al. 2023. Spatially informed clustering, integration, and deconvolution of spatial transcriptomics with GraphST. *Nature Communications* 14, 1 (2023), 1155.
- [29] Hu Lu, Chao Chen, Hui Wei, Zhongchen Ma, Ke Jiang, and Yingquan Wang. 2022. Improved deep convolutional embedded clustering with re-selectable sample training. *Pattern Recognition* 127 (2022), 108611.
- [30] Yadong Lu, Julian Collado, Daniel Whiteson, and Pierre Baldi. 2021. Sparse autoregressive models for scalable generation of sparse images in particle physics. *Physical Review D* 103, 3 (2021), 036012.
- [31] Brenda Pardo, Abby Spangler, Lukas M Weber, Stephanie C Page, Stephanie C Hicks, Andrew E Jaffe, Keri Martinovich, Kristen R Maynard, and Leonardo Collado-Torres. 2022. spatialLBD: an R/Bioconductor package to visualize spatially-resolved transcriptomics data. *BMC genomics* 23, 1 (2022), 434.
- [32] Yazhou Ren, Jingyu Pu, Zhimeng Yang, Jie Xu, Guofeng Li, Xiaorong Pu, Philip S Yu, and Lifang He. 2022. Deep clustering: A comprehensive survey. *arXiv preprint arXiv:2210.04142* (2022).
- [33] Scott Sloka. 2023. Image denoising in astrophotography—an approach using recent network denoising models. *Journal of the British Astronomical Association* 133, 4 (2023).
- [34] Tianci Song, Kathleen K Markham, Zhulu Li, Kristen E Muller, Kathleen Greenham, and Rui Kuang. 2022. Detecting spatially co-expressed gene clusters with functional coherence by graph-regularized convolutional neural network. *Bioinformatics* 38, 5 (2022), 1344–1352.
- [35] Patrik L Ståhl, Fredrik Salmén, Sanja Vickovic, Anna Lundmark, José Fernández Navarro, Jens Magnusson, Stefania Giacomello, Michaela Asp, Jakob O Westholm, Mikael Huss, et al. 2016. Visualization and analysis of gene expression in tissue sections by spatial transcriptomics. *Science* 353, 6294 (2016), 78–82.
- [36] Vincent A Traag, Ludo Waltman, and Nees Jan Van Eck. 2019. From Louvain to Leiden: guaranteeing well-connected communities. *Scientific reports* 9, 1 (2019), 5233.
- [37] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2017. Graph attention networks. *arXiv preprint arXiv:1710.10903* (2017).
- [38] Jinghua Wang and Jianmin Jiang. 2021. Unsupervised deep clustering via adaptive GMM modeling and optimization. *Neurocomputing* 433 (2021), 199–211.
- [39] Yuxing Wang, Wenguan Wang, Dongfang Liu, Wenpin Hou, Tianfei Zhou, and Zhicheng Ji. 2023. GeneSegNet: a deep learning framework for cell segmentation by integrating gene expression and imaging. *Genome Biology* 24, 1 (2023), 235.
- [40] F Alexander Wolf, Philipp Angerer, and Fabian J Theis. 2018. SCANPY: large-scale single-cell gene expression data analysis. *Genome biology* 19 (2018), 1–5.
- [41] Junyuan Xie, Ross Girshick, and Ali Farhadi. 2016. Unsupervised deep embedding for clustering analysis. In *International conference on machine learning*. PMLR, 478–487.
- [42] Zhenda Xie, Zheng Zhang, Yue Cao, Yutong Lin, Jianmin Bao, Zhuliang Yao, Qi Dai, and Han Hu. 2022. Simmim: A simple framework for masked image modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 9653–9663.
- [43] Jun Yin, Haowei Wu, and Shiliang Sun. 2023. Effective sample pairs based contrastive learning for clustering. *Information Fusion* 99 (2023), 101899.
- [44] Xiaokang Yu, Xinyi Xu, Jingxiao Zhang, and Xiangjie Li. 2023. Batch alignment of single-cell transcriptomics data using deep metric learning. *Nature Communications* 14, 1 (2023), 960.
- [45] Mingkai Zheng, Fei Wang, Shan You, Chen Qian, Changshui Zhang, Xiaogang Wang, and Chang Xu. 2021. Weakly supervised contrastive learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 10042–10051.
- [46] Hufeng Zhou, Jingjing Jin, Haojun Zhang, Bo Yi, Michal Wozniak, and Limsoon Wong. 2012. IntPath—an integrated pathway gene relationship database for model organisms and important pathogens. In *BMC systems biology*, Vol. 6. BioMed Central, 1–17.
- [47] Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan Yuille, and Tao Kong. 2021. ibot: Image bert pre-training with online tokenizer. *arXiv preprint arXiv:2111.07832* (2021).
- [48] Jiaqiang Zhu, Shiquan Sun, and Xiang Zhou. 2021. SPARK-X: non-parametric modeling enables scalable and robust detection of spatial expression patterns for large spatial transcriptomic studies. *Genome biology* 22, 1 (2021), 1–25.
- [49] Bo Zong, Qi Song, Martin Renqiang Min, Wei Cheng, Cristian Lumezanu, Daeki Cho, and Haifeng Chen. 2018. Deep autoencoding gaussian mixture model for

929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044

1045	unsupervised anomaly detection. In <i>International conference on learning representations</i> .	1103
1046		1104
1047		1105
1048		1106
1049		1107
1050		1108
1051		1109
1052		1110
1053		1111
1054		1112
1055		1113
1056		1114
1057		1115
1058		1116
1059		1117
1060		1118
1061		1119
1062		1120
1063		1121
1064		1122
1065		1123
1066		1124
1067		1125
1068		1126
1069		1127
1070		1128
1071		1129
1072		1130
1073		1131
1074		1132
1075		1133
1076		1134
1077		1135
1078		1136
1079		1137
1080		1138
1081		1139
1082		1140
1083		1141
1084		1142
1085		1143
1086		1144
1087		1145
1088		1146
1089		1147
1090		1148
1091		1149
1092		1150
1093		1151
1094		1152
1095		1153
1096		1154
1097		1155
1098		1156
1099		1157
1100		1158
1101		1159
1102		1160