Generalized Probabilistic Graphical Modeling for Multi-View Bipartite Graph Clustering

Liang Li, *Graduate Student Member, IEEE*, Yuangang Pan, Yinghua Yao, Junpu Zhang, Moyun Liu, Xueling Zhu, Xinwang Liu, *Senior Member, IEEE*, Kenli Li, *Senior Member, IEEE*, Ivor W. Tsang, *Fellow, IEEE*, and Keqin Li, *Fellow, IEEE*

***Abstract*—Multi-view bipartite graph clustering (MVBGC) is an active pipeline in unsupervised learning to tackle the limited scalability issue of traditional graph clustering. Despite improved performance, numerous variants still fall under conventional modeling that plugs additional modules, which however induces increasingly intricate models and fails to reveal the inherent variable relationship. We make the first attempt to introduce probabilistic graphical models for modeling the multi-view bipartite graph clustering task, reformulating it as a maximum likelihood estimation (MLE) problem. Such a setting uncovers the underlying probabilistic correlations among the commonality, view-specific variables, and noisy components. By pruning redundancy and disturbance collectively referred to as noise, we prove that minimizing the total noise is an approximation of the lower bound of MLE for multi-view data observations. We further generalize the MLE setting with clustering-suited constraints, deriving a Generalized Probabilistic Graphical Modeling framework (GProM), achieving an interpretable, concise, and flexible MVBGC framework. Extensive experiments verify the effectiveness of our framework. Furthermore, statistical significance analysis reveals the effectiveness of different distribution assumptions, providing valuable insights for model design.**

Received 20 December 2024; revised 1 July 2025; accepted 3 August 2025. Date of publication 7 August 2025; date of current version 5 November 2025. This work was supported in part by the National Natural Science Foundation of China under Grant 62325604, Grant 62276271, Grant 62473380, and Grant 62272481, in part by the National Key Research and Development Program of China under Grant 2021YFB0300101, in part by the Science and Technology Innovation Program of Hunan Province under Grant 2023RC1029, and in part by China Scholarship Council under Grant 202306110001. Recommended for acceptance by J. Wu. *(Corresponding authors: Xinwang Liu; Xueling Zhu.)*

Liang Li, Junpu Zhang, and Xinwang Liu are with the College of Computer Science and Technology, National University of Defense Technology, Changsha 410073, China (e-mail: liliang1037@gmail.com; zhangjunpu@nudt.edu.cn; xinwangliu@nudt.edu.cn).

Yuangang Pan, Yinghua Yao, and Ivor W. Tsang are with the Center for Frontier AI Research, Agency for Science, Technology and Research (A*STAR), Singapore 138632 (e-mail: yuangang.pan@gmail.com; Yao_Yinghua@cfar.a-star.edu.sg; ivor.tsang@gmail.com).

Moyun Liu is with the Mechanical Science and Engineering, Huazhong University of Science and Technology, Wuhan 430074, China (e-mail: lmomoy@hust.edu.cn).

Xueling Zhu is with the Department of Radiology, National Clinical Research Center for Geriatric Disorders, Xiangya Hospital, Central South University, Changsha 410073, China (e-mail: zhuxueling@csu.edu.cn).

Kenli Li is with the College of Computer Science and Electronic Engineering, Supercomputing and Cloud Computing Institute, Hunan University, Changsha 410073, China (e-mail: lkl@hnu.edu.cn).

Keqin Li is with the Department of Computer Science, State University of New York, New Paltz, NY 12561 USA (e-mail: lik@newpaltz.edu).

The code is provided at https://github.com/liliangnudt/GProM.

This article has supplementary downloadable material available at https://doi.org/10.1109/TPAMI.2025.3596764, provided by the authors.

Digital Object Identifier 10.1109/TPAMI.2025.3596764

***Index Terms*—Bipartite graph clustering, multi-view learning, probabilistic graphical models.**

I. Introduction

With dramatically growing data from diverse sources, such as in self-driving scenarios, cars sense the surroundings through cameras, lidar, and radar, annotating massive data is cost- and labor-intensive, which is an urgent need to develop unsupervised learning [1], [2], [3], [4]. Typically, graph clustering [5], [6], [7] has emerged as a fast-growing pipeline in multi-view clustering (MVC) [8], [9], [10], widely employed in social networks, bioinformatics, and recommendation [11], [12], [13].

Early multi-view graph clustering (MVGC) [14], [15], [16] required building pairwise memberships, resulting in quadratic space complexity and cubic time complexity w.r.t. instances, degrading scalability for large-scale applications. To alleviate this limitation, multi-view bipartite graph clustering (MVBGC) [17], [18], [19], [20], [21] was proposed by constructing memberships of prototype-instance pairs, showing promising scalability with linear complexity.

A typical MVBGC model includes three procedures: prototype selection, bipartite graph construction, and multi-graph fusion. Numerous variants [20], [22], [23], [24] have been proposed to enhance these components by coupling various techniques. For example, prototype enhancement strategy [25] augments the discrimination of pre-sampled prototypes from k-means; graph filtering [26] refines the bipartite graph construction by smoothing graph signals; tensor-based [27] or diversity-induced [28] concatenation strengthens multi-graph fusion by concatenating bipartite graphs. Despite the pleasing performance, these methods still fall under conventional design, wherein plugging more and more modules directly results in increasingly intricate models. More critically, these manners are unable to uncover the underlying variable relationship.

Orthogonal to the conventional MVBGC routine, this paper introduces probabilistic graphical models [29] to explore the probabilistic correlations among variables, reformulating MVBGC as a maximum likelihood estimation (MLE) problem. Specifically, we revisit the classical linear reconstruction backbone [18] from the perspective of probabilistic graphical model and find that it overlooks the impact of redundancy and disturbance within the data observations and latent variable, thereby hindering the discovery of inherent structural information. By

uncoupling these undesired components, identified as noise, we derive a generalized linear reconstruction model. Using Gaussian and rotational invariant Laplacian distribution [30] as examples for likelihood instantiation, we reveal the connection between the MLE of multi-view data and the minimization of noise under distribution-related penalties. We further generalize the MLE problem with appropriate constraints, deriving a generalized probabilistic graphical modeling framework, termed GProM, which is well-suited for clustering tasks.

Our contributions are summarized as follows:

1) We pioneer the use of probabilistic graphical models to formalize the multi-view bipartite graph clustering task and reformulate it as a MLE problem, which reveals the probabilistic dependencies among commonality, view-specific variables, and noisy components, distinguishing our work from conventional designs.
2) We theoretically verify that minimizing the sum of feature- and structure-level noise approximates the lower bound of MLE for multi-view data observations. We further generalize the MLE setting with appropriate constraints, enabling compatibility with clustering tasks and formulating a generalized probabilistic modeling framework for MVBGC.
3) Extensive experiments verify the effectiveness of our model. The empirical results indicate that the penalty derived from the rotational invariant Laplacian distribution outperforms Gaussian distribution-based counterpart for modeling feature-level noise, while both distributions exhibit comparable significance for structural-level noise.

II. Related Research

This section briefly outlines typical techniques of MVBGC, including (1) prototype selection, (2) bipartite graph construction, (3) multi-graph fusion.

Prototype Selection: Early research typically samples prototypes from all instances. Random sampling is intuitive and the representative is Nyström [31]. k-means is a widely used method, but it is sensitive to initialization [18], [32]. DPP sampling [33] is another popular method, particularly in recommendation. However, it needs to build a Gram matrix with enormous complexity. Some hybrid methods seek a trade-off between the above methods [34], [35]. Recently, DAS [17] alternately generated prototypes to cover the entire point cloud, and BKHK [36] introduced binary tree to improve efficiency. A recent popular strategy is to learn prototypes. Pioneering works [37], [38], [39] incorporated prototypes into the optimization, which enhances flexibility and avoids the one-shot approximation problem of sampling methods.

Bipartite Graph Construction: Linear and locally linear reconstructions [17], [18] are two commonly employed backbones. The former assumes that input features can be linearly reconstructed from prototypes via similarity coefficient [40], whereas the latter assumes that high-dimensional features lie on a mixture of low-dimensional submanifolds [41]. Most variants [28], [42] are based on them.

Multi-graph Fusion: Stage-wise fusion ensembles pre-generated candidate bipartite graphs. Two typical strategies are linear-combination [17] and concatenation [18]. However, such manners may yield suboptimal solutions. Conversely, collaborative fusion unifies graph construction and commonality fusion, considering dependencies among variables. Pioneering work [38] projected multiple views into a latent space through orthogonal mapping, directly building consensus information, which avoids separate optimization. However, as pointed out in [43], mapping all features into low-dimensional space via orthogonal projection easily induces information loss with degraded performance.

Numerous variants have been developed. For instance, prototype enhancement [25] augments the diversity of prototypes initially generated by k-means; graph filtering [26], manifold learning [44], sparse [45] and low-rank [46] regularizations refine the bipartite graph construction; diversity-induced [28], multi-scale [47], contrastive learning [48], or tensor-based concatenation [49], [50] enhance multi-graph fusion. These techniques have also been extended to tackle incomplete data [51], [52], [53]. Despite pleasing results, these variants increasingly introduce extra modules, inducing intricate models, limiting their practical deployment.

TABLE I
Notations

Notation	Explanation
n, v, k, m	Instances, views, clusters, prototypes
d_l	Feature dimension for the l-th view
λ, γ, ρ	ALM parameters, scaling parameter
Θ	Model parameters
$\theta_{\mathcal{F}}$, $\theta_{\mathcal{G}}$	Distribution parameters of $\mathbf{e}^{\mathcal{F}}_{l[j]}$ and $\mathbf{e}^{\mathcal{G}}_{l[j]}$
$\mathbb{E}(\cdot)$, $\Phi(\cdot)$	Expectation term, regularization term
$\mathbf{h}_l \in \mathbb{R}^m$	The latent variable of the l-th view
$\mathbf{X}_l \in \mathbb{R}^{d_l \times n}$	The l-th input view
$\mathbf{O}_l \in \mathbb{R}^{d_l \times m}$	The l-th prototype matrix
$\mathbf{E}^{\mathcal{F}}_l \in \mathbb{R}^{d_l \times n}$	The l-th feature-level noise matrix
$\mathbf{E}^{\mathcal{G}}_l$, $\widetilde{\mathbf{E}}^{\mathcal{G}}_l \in \mathbb{R}^{m \times n}$	The l-th structure-level noise matrix, auxiliary variable
$\mathbf{U} \in \mathbb{R}^{m \times n}$	Refined bipartite graph
$\mathbf{R} \in \mathbb{R}^{(n+m) \times k}$	Graph embedding
$\mathbf{S} \in \mathbb{R}^{(n+m) \times (n+m)}$	Augmented affinity matrix
$\widetilde{\mathbf{L}} \in \mathbb{R}^{(n+m) \times (n+m)}$	Normalized Laplacian matrix
$\mathbf{\Lambda}_l \in \mathbb{R}^{d_l \times n}$, $\mathbf{\Upsilon}_l \in \mathbb{R}^{m \times n}$	ALM multipliers
$\mathcal{N}(\cdot)$	Gaussian distribution
$\mathcal{L}(\cdot)$	Rotational invariant Laplacian distribution

III. Methodology

This section presents the technical roadmap for modeling the MVBGC problem using probabilistic graphical models. Table I lists the notations used throughout this work.

A. Interpretable Multi-View Linear Reconstruction Model From a Probability Perspective

Let us consider multiple view raw features $\{\mathbf{X}_l\}_{l=1}^v = \{[\mathbf{x}_{l[1]}, \mathbf{x}_{l[2]}, \ldots, \mathbf{x}_{l[n]}]\}_{l=1}^v$ and view-specific prototypes $\{\mathbf{O}_l\}_{l=1}^v = \{[\mathbf{o}_{l[1]}, \mathbf{o}_{l[2]}, \ldots, \mathbf{o}_{l[m]}]\}_{l=1}^v$, where v, n, m denote number of views, instances, and prototypes, respectively. The classical linear reconstruction backbone [18] assumes that each input instance $\mathbf{x}_{l[j]}$ can be reconstructed linearly by prototypes $\mathbf{O}_l$ via coefficient $\mathbf{h}_{l[j]}$, which is expressed by:

$$\mathbf{x}_{l[j]} = \mathbf{O}_l \mathbf{h}_{l[j]}. \tag{1}$$

For the lth view, $\mathbf{h}_{l[j]} = [h_{l[1j]}, h_{l[2j]}, \ldots, h_{l[mj]}]^\top$ represents the transition probability from the jth instance to m prototypes

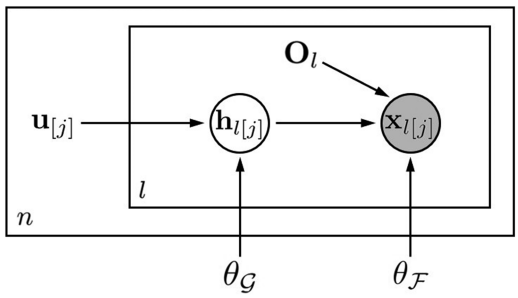


Fig. 1. Diagram for the generalized linear reconstruction model from a probabilistic graphical perspective. $\mathbf{x}_{l[j]}$ is the data observation, $\mathbf{h}_{l[j]}$ is the latent variable, $\mathbf{u}_{[j]}$ is an instance-specific parameter, and $\mathbf{O}_l$ is a view-specific parameter. $\theta_{\mathcal{F}}$ and $\theta_{\mathcal{G}}$ are remaining parameters.

in a one-step stationary Markov random-walk [52]. The normalized probability is given by $p(\mathbf{x}_{l[j]} \mapsto \mathbf{o}_{l[r]}) = \frac{h_{l[rj]}}{\sum_{r=1}^{m} h_{l[rj]}}$ [54], where $\mapsto$ denotes the transitioning. If we relax the requirement that the sum of probabilities $\sum_{r=1}^{m} h_{l[rj]} = 1$, (1) actually describes a basic probabilistic graphical model [29], where $\{\mathbf{x}_{l[j]}\}_{j=1}^{n}$ denote independent and identically distributed (i.i.d) multi-view data observations, $\mathbf{h}_{l[j]}$ is the latent variable, and $\mathbf{O}_l$ is the parameter.

However, the deployment of classical linear reconstruction model in real-world scenarios often deviates from the ideal conditions in (1) due to the presence of internal or external noisy information. For instance, observations $\mathbf{x}_{l[j]}$ are inevitably contaminated by sensor errors or resolution limitations, and these adverse factors further degrade the quality and reliability of the latent variable $\mathbf{h}_{l[j]}$. For simplicity, these redundancy and disturbance are referred to as noise and should be disentangled.

Definition 1: Feature Noise $\mathbf{e}_{l[j]}^{\mathcal{F}}$. It refers to disturbances and errors inherent in the data observations, which should be pruned at the feature level, i.e., $\mathbf{x}_{l[j]} = \mathbf{O}_l \mathbf{h}_{l[j]} + \mathbf{e}_{l[j]}^{\mathcal{F}}$.

Definition 2: Structure Noise $\mathbf{e}_{l[j]}^{\mathcal{G}}$. It refers to the deviation of the view-specific latent variable $\mathbf{h}_{l[j]}$ from the consensus parameter $\mathbf{u}_{[j]}$, which should be pruned at the structure level, i.e., $\mathbf{h}_{l[j]} = \mathbf{u}_{[j]} + \mathbf{e}_{l[j]}^{\mathcal{G}}$.

Based on above definitions, a general multi-view linear reconstruction model is derived as follows:

$$\mathbf{x}_{l[j]} = \mathbf{O}_l \left(\mathbf{u}_{[j]} + \mathbf{e}_{l[j]}^{\mathcal{G}} \right) + \mathbf{e}_{l[j]}^{\mathcal{F}}. \tag{2}$$

Fig. 1 depicts the probabilistic relationship of (2) based on the probabilistic graph model. Further, Theorem 1 gives the lower bound of the log-likelihood of the multi-view data observations. For simplicity, we assume that multiple views share the same parameters $\theta_{\mathcal{F}}$ and $\theta_{\mathcal{G}}$.

Theorem 1: Given i.i.d instances $\{\mathbf{x}_{l[j]}\}_{j=1}^{n}$, the lower bound of the log-likelihood of observations within the general linear reconstruction model can be derived as follows:

$$\begin{aligned}
&\log \prod_{l=1}^{v} \prod_{j=1}^{n} p(\mathbf{x}_{l[j]}; \Theta) \\
&\overset{(i)}{=} \sum_{l=1}^{v} \sum_{j=1}^{n} \log \mathbb{E}_{q(\mathbf{h}_{l[j]})} \left[\frac{p(\mathbf{x}_{l[j]}, \mathbf{h}_{l[j]}; \Theta)}{q(\mathbf{h}_{l[j]})} \right] \\
&= \sum_{l=1}^{v} \sum_{j=1}^{n} \log \mathbb{E}_{q(\mathbf{h}_{l[j]})} \left[\frac{p(\mathbf{x}_{l[j]} | \mathbf{h}_{l[j]}; \mathbf{O}_l, \theta_{\mathcal{F}}) p(\mathbf{h}_{l[j]}; \mathbf{u}_{[j]}, \theta_{\mathcal{G}})}{q(\mathbf{h}_{l[j]})} \right] \\
&\overset{(ii)}{\geq} \sum_{l=1}^{v} \sum_{j=1}^{n} \mathbb{E}_{q(\mathbf{h}_{l[j]})} \left[\log p(\mathbf{x}_{l[j]} | \mathbf{h}_{l[j]}; \mathbf{O}_l, \theta_{\mathcal{F}}) \right. \\
&\qquad \left. + \log p\left(\mathbf{h}_{l[j]}; \mathbf{u}_{[j]}, \theta_{\mathcal{G}} \right) \right] + H(q(\mathbf{h}_{l[j]})) \\
&\overset{(iii)}{=} \sum_{l=1}^{v} \sum_{j=1}^{n} \mathbb{E}_{q(\mathbf{e}_{l[j]}^{\mathcal{G}})} \left[\log p(\mathbf{x}_{l[j]} | \mathbf{h}_{l[j]}; \mathbf{O}_l, \theta_{\mathcal{F}}) \right. \\
&\qquad \left. + \log p\left(\mathbf{h}_{l[j]}; \mathbf{u}_{[j]}, \theta_{\mathcal{G}} \right) \right] + \underbrace{H(q(\mathbf{e}_{l[j]}^{\mathcal{G}}))}_{\Phi(\cdot)}
\end{aligned} \tag{3}$$

where Θ denotes all model parameters; the conditional likelihood $p(\mathbf{x}_{l[j]} | \mathbf{h}_{l[j]}; \mathbf{O}_l, \theta_{\mathcal{F}})$ parameterized by $\mathbf{O}_l$ and $\theta_{\mathcal{F}}$, describes how the latent variable generates data observations; $p(\mathbf{h}_{l[j]}; \mathbf{u}_{[j]}, \theta_{\mathcal{G}})$ parameterized by $\mathbf{u}_{[j]}$ and $\theta_{\mathcal{G}}$, defines the distribution of the latent variable; $\theta_{\mathcal{F}}$ and $\theta_{\mathcal{G}}$ denote the parameters controlling the distributions of $\mathbf{e}_{l[j]}^{\mathcal{F}}$ and $\mathbf{e}_{l[j]}^{\mathcal{G}}$, respectively, and these parameters are endowed with physical meanings in instantiation, as illustrated in Section III-B1.

Equality (i) is straightforward that introduces the approximate posterior $q(\mathbf{h}_{l[j]})$ to both the numerator and denominator. Inequality (ii) holds because of Jensen's inequality. Equality (iii) is valid by substituting the expectation of $\mathbf{h}_{l[j]}$ with that of $\mathbf{e}_{l[j]}^{\mathcal{G}}$, based on the linear assumption about the latent variable in Definition 2, i.e., $\mathbf{h}_{l[j]} = \mathbf{u}_{[j]} + \mathbf{e}_{l[j]}^{\mathcal{G}}$.

Equality (iii) includes an expectation term $\mathbb{E}(\cdot)$ and an entropy term $H(\cdot)$. Note that $H(\cdot)$ is associated with $\mathbf{e}_{l[j]}^{\mathcal{G}}$. Actually, this term can be generalized into a broader regularization term $\Phi(\cdot)$ with respect to the model parameters Θ, i.e., $\Phi(\Theta)$. This generalization enables a more flexible configuration tailored to specific practical problems, rather than being limited to $\mathbf{e}_{l[j]}^{\mathcal{G}}$.

Remark 1: No specific assumptions are imposed on $p(\mathbf{x}_{l[j]} | \mathbf{h}_{l[j]}; \mathbf{O}_l, \theta_{\mathcal{F}})$ or $p(\mathbf{h}_{l[j]}; \mathbf{u}_{[j]}, \theta_{\mathcal{G}})$. Therefore, (3) depicts a general probabilistic graphical model that encodes probabilistic dependencies of observations $\mathbf{x}_{l[j]}$, latent variables $\mathbf{h}_{l[j]}$, and parameters $\mathbf{O}_l$ and $\mathbf{u}_{[j]}$.

B. Probabilistic Graphical Modeling for MVBGC Task

This section instantiates the expectation term $\mathbb{E}(\cdot)$ and the regularization term $\Phi(\cdot)$ within the lower bound in (3).

1) Instantiation of the Expectation Term $\mathbb{E}(\cdot)$: This section instantiates the first term $\mathbb{E}(\cdot)$ in (3). Considering the challenges of solving expectations analytically, we first employ Monte Carlo approximation to reformulate $\mathbb{E}(\cdot)$ as a statistical estimation problem, i.e.,

$$\begin{aligned}
\mathbb{E}(\cdot) \Rightarrow \sum_{l=1}^{v} \sum_{j=1}^{n} \sum_{t=1}^{T} \frac{1}{T} \left[\log p\left(\mathbf{x}_{l[j]} | \mathbf{h}_{l[j]}^{t}; \mathbf{O}_l, \theta_{\mathcal{F}} \right) \right. \\
\left. + \log p\left(\mathbf{h}_{l[j]}^{t}; \mathbf{u}_{[j]}, \theta_{\mathcal{G}} \right) \right],
\end{aligned} \tag{4}$$

where T is set to 1 for simplicity. It allows us to focus on instantiating $p(\mathbf{x}_{l[j]}|\mathbf{h}_{l[j]};\mathbf{O}_l,\theta_{\mathcal{F}})$ and $p(\mathbf{h}_{l[j]};\mathbf{u}_{[j]},\theta_{\mathcal{G}})$ more effectively, where we omit the superscripts for the sampling index.

Based on (4), we derive two specific instantiations under different distribution assumptions.

The first assumption is Gaussian distribution, where the sample mean is an unbiased estimator of the population mean. Gaussian mixture model [55] is a typical density-based clustering method developed on it. Inspired by this, Proposition 1 gives the instantiated (4) by Gaussian distribution. For simplicity, we define $\mathcal{O}=\{\mathbf{O}_l\}_{l=1}^{v}\cup\{\mathbf{u}_{[j]}\}_{j=1}^{n}$, $\mathcal{E}^{\mathcal{F}}=\{\mathbf{e}_{l[j]}^{\mathcal{F}}\}_{l,j=1}^{v,n}$, $\mathcal{E}^{\mathcal{G}}=\{\mathbf{e}_{l[j]}^{\mathcal{G}}\}_{l,j=1}^{v,n}$, where l denotes the view index and j denotes the instance index.

Proposition 1: When we instantiate $p(\mathbf{x}_{l[j]}|\mathbf{h}_{l[j]};\mathbf{O}_l,\theta_{\mathcal{F}})$ and $p(\mathbf{h}_{l[j]};\mathbf{u}_{[j]},\theta_{\mathcal{G}})$ by Gaussian distribution $\mathcal{N}(\mathbf{x}_{l[j]};\boldsymbol{\mu}_{\mathcal{F}},\sigma_{\mathcal{F}}^2\mathbf{I})$ and $\mathcal{N}(\mathbf{h}_{l[j]};\boldsymbol{\mu}_{\mathcal{G}},\sigma_{\mathcal{G}}^2\mathbf{I})$, respectively, the mean and variance are endowed with physical meanings, i.e., $\boldsymbol{\mu}_{\mathcal{F}}=\mathbf{O}_l\mathbf{h}_{l[j]}$, $\boldsymbol{\mu}_{\mathcal{G}}=\mathbf{u}_{[j]}$, $\sigma_{\mathcal{F}}^2\mathbf{I}=\theta_{\mathcal{F}}$, $\sigma_{\mathcal{G}}^2\mathbf{I}=\theta_{\mathcal{G}}$. It is equivalent to $\mathbf{e}_{l[j]}^{\mathcal{F}}$ and $\mathbf{e}_{l[j]}^{\mathcal{G}}$ following zero-mean $\mathcal{N}(\mathbf{e}_{l[j]}^{\mathcal{F}};\mathbf{0},\theta_{\mathcal{F}})$ and $\mathcal{N}(\mathbf{e}_{l[j]}^{\mathcal{G}};\mathbf{0},\theta_{\mathcal{G}})$, respectively. (4) is converted into:

$$\max_{\mathcal{O}}\sum_{l=1}^{v}\sum_{j=1}^{n}\left(-\frac{\left\|\mathbf{x}_{l[j]}-\mathbf{O}_l\mathbf{h}_{l[j]}\right\|_2^2}{2\sigma_{\mathcal{F}}^2}-\frac{\left\|\mathbf{h}_{l[j]}-\mathbf{u}_{[j]}\right\|_2^2}{2\sigma_{\mathcal{G}}^2}\right)$$
$$\Rightarrow\min_{\mathcal{E}^{\mathcal{F}},\mathcal{E}^{\mathcal{G}}}\sum_{l=1}^{v}\sum_{j=1}^{n}\left(\left\|\mathbf{e}_{l[j]}^{\mathcal{F}}\right\|_2^2+\eta\left\|\mathbf{e}_{l[j]}^{\mathcal{G}}\right\|_2^2\right),\tag{5}$$

where $\eta=\frac{\sigma_{\mathcal{F}}^2}{\sigma_{\mathcal{G}}^2}$.

Note that two levels of noise are measured by the squared ℓ_2-norm in (5), making the objective sensitive to outliers that may be caused by sensor errors. To alleviate this, we introduce another rotational invariant Laplacian distribution [30] to instantiate (4).

Proposition 2: When we instantiate $p(\mathbf{x}_{l[j]}|\mathbf{h}_{l[j]};\mathbf{O}_l,\theta_{\mathcal{F}})$ and $p(\mathbf{h}_{l[j]};\mathbf{u}_{[j]},\theta_{\mathcal{G}})$ by rotational invariant Laplacian distribution $\mathcal{L}(\mathbf{x}_{l[j]};\boldsymbol{\mu}_{\mathcal{F}},b_{\mathcal{F}}\mathbf{I})$ and $\mathcal{L}(\mathbf{h}_{l[j]};\boldsymbol{\mu}_{\mathcal{G}},b_{\mathcal{G}}\mathbf{I})$, respectively, the location parameter and scale parameter are endowed with physical meanings, i.e., $\boldsymbol{\mu}_{\mathcal{F}}=\mathbf{O}_l\mathbf{h}_{l[j]}$, $\boldsymbol{\mu}_{\mathcal{G}}=\mathbf{u}_{[j]}$, $b_{\mathcal{F}}\mathbf{I}=\theta_{\mathcal{F}}$, $b_{\mathcal{G}}\mathbf{I}=\theta_{\mathcal{G}}$. It is equivalent to $\mathbf{e}_{l[j]}^{\mathcal{F}}$ and $\mathbf{e}_{l[j]}^{\mathcal{G}}$ following $\mathcal{L}(\mathbf{e}_{l[j]}^{\mathcal{F}};\mathbf{0},\theta_{\mathcal{F}})$ and $\mathcal{L}(\mathbf{e}_{l[j]}^{\mathcal{G}};\mathbf{0},\theta_{\mathcal{G}})$, respectively. Eq. (4) is converted into:

$$\max_{\mathcal{O}}\sum_{l=1}^{v}\sum_{j=1}^{n}\left(-\frac{\left\|\mathbf{x}_{l[j]}-\mathbf{O}_l\mathbf{h}_{l[j]}\right\|_2}{2b_{\mathcal{F}}}-\frac{\left\|\mathbf{h}_{l[j]}-\mathbf{u}_{[j]}\right\|_2}{2b_{\mathcal{G}}}\right)$$
$$\Rightarrow\min_{\mathcal{E}^{\mathcal{F}},\mathcal{E}^{\mathcal{G}}}\sum_{l=1}^{v}\sum_{j=1}^{n}\left(\left\|\mathbf{e}_{l[j]}^{\mathcal{F}}\right\|_2+\eta\left\|\mathbf{e}_{l[j]}^{\mathcal{G}}\right\|_2\right),\tag{6}$$

where $\eta=\frac{b_{\mathcal{F}}}{b_{\mathcal{G}}}$.

In (6), two levels of noise are measured by ℓ_2-norm, thereby enhancing robustness against outliers. Details of Propositions 1–2 are provided in supplementary materials (Section 1-2).

Remark 2: We use $\mathcal{N}(\cdot)$ and $\mathcal{L}(\cdot)$ to instantiate the log-likelihoods to (4), which reveals that minimizing the sum of two levels of noise (i.e., (5) and (6)) approximates the lower bound of MLE of multi-view data observations. Importantly, (4) is a flexible framework that is compatible with different distribution combinations and assumptions.

2) Instantiation of the Regularization Term $\Phi(\Theta)$: This section instantiates the second term $\Phi(\Theta)$ in (3). Given that practical clustering tasks often incorporate domain-specific knowledge to reduce the search space [17], we impose task-oriented constraints on the model parameters.

Constraint on $\mathbf{O}_l$: The prototypes play a crucial role in capturing the underlying data distribution. To promote diversity, we impose a Stiefel manifold constraint $\phi(\mathbf{O}_l)=\{\mathbf{O}_l|\mathbf{O}_l^\top\mathbf{O}_l=\mathbf{I}_m\}$, which encourages the prototypes to span the entire point cloud, thereby enhancing variety.

Constraint on $\{\mathbf{h}_{l[j]},\mathbf{u}_{[j]},\mathbf{e}_{l[j]}^{\mathcal{G}}\}$: To normalize the instance-prototype transition probability, the view-specific $\mathbf{h}_l$ and the consistent $\mathbf{u}$ should hold $\{\mathbf{h}_l|\mathbf{h}_l^\top\mathbf{1}_m=1,\ \mathbf{h}_l\geq 0\}$ and $\phi(\mathbf{u})=\{\mathbf{u}|\mathbf{u}^\top\mathbf{1}_m=1,\ \mathbf{u}\geq 0\}$, respectively. Naturally, $\mathbf{e}_l^{\mathcal{G}}$ should also adhere to such a constraint. However, the above constraints are inconsistent with Definition 2. To address this issue, we introduce hyperparameters $0<\xi,\beta<1$ to balance their relationship, such that the Definition 2 is relaxed to $\mathbf{h}_l=\xi\mathbf{u}+\beta\mathbf{e}_l^{\mathcal{G}}$ with $\xi+\beta=1$. Note that ξ and β cannot be modeled by probabilistic graphical models but are instead manually specified. It is straightforward to derive their matrix form $\phi(\mathbf{U})$ and $\phi(\mathbf{E}_l^{\mathcal{G}})$ with $\mathbf{U}=[\mathbf{u}_{[1]},\mathbf{u}_{[2]},\ldots,\mathbf{u}_{[n]}]$ and $\mathbf{E}_l^{\mathcal{G}}=[\mathbf{e}_{l[1]}^{\mathcal{G}},\mathbf{e}_{l[2]}^{\mathcal{G}},\ldots,\mathbf{e}_{l[n]}^{\mathcal{G}}]$. The details are omitted for brevity.

Besides, a discriminative $\mathbf{U}$ comprises disjoint k-connected components, with each one corresponding to a distinct cluster. Drawing inspiration from [17], we incorporate a connectivity constraint $\phi(\widetilde{\mathbf{L}})=\{\widetilde{\mathbf{L}}|rank(\widetilde{\mathbf{L}})=n+m-k\}$, where $\widetilde{\mathbf{L}}=\mathbf{I}-\boldsymbol{\Pi}^{-\frac{1}{2}}\mathbf{S}\boldsymbol{\Pi}^{-\frac{1}{2}}$ is the normalized Laplacian matrix, $\mathbf{S}=\begin{bmatrix}\mathbf{0}&\mathbf{U}^\top\\\mathbf{U}&\mathbf{0}\end{bmatrix}$, $\boldsymbol{\Pi}=\begin{bmatrix}\boldsymbol{\Omega}_n&\mathbf{0}\\\mathbf{0}&\boldsymbol{\Omega}_m\end{bmatrix}$, $\boldsymbol{\Omega}_n=diag(\mathbf{U}^\top\mathbf{1})$, and $\boldsymbol{\Omega}_m=diag(\mathbf{U}\mathbf{1})$.

Constraint on $\mathbf{e}_{l[j]}^{\mathcal{F}}$: As stated in Definition 1, $\mathbf{e}_{l[j]}^{\mathcal{F}}$ is the disturbances detached from the unconstrained observations. Therefore, $\phi(\mathbf{E}_l^{\mathcal{F}})=\{\mathbf{E}_l^{\mathcal{F}}\mid\mathbf{E}_l^{\mathcal{F}}\in\mathbb{R}^{d_l\times n}\}$ with $\mathbf{E}_l^{\mathcal{F}}=[\mathbf{e}_{l[1]}^{\mathcal{F}},\mathbf{e}_{l[2]}^{\mathcal{F}},\ldots,\mathbf{e}_{l[n]}^{\mathcal{F}}]$ remains unconstrained.

3) The Proposed GProM Framework: Consequently, GProM is formulated as:

$$\min_{\Theta}\sum_{l=1}^{v}\sum_{j=1}^{n}\left(\left\|\mathbf{E}_l^{\mathcal{F}}\right\|_{\mathcal{X}}+\eta\left\|\mathbf{E}_l^{\mathcal{G}}\right\|_{\mathcal{X}}\right)+\Phi(\Theta),\tag{7}$$

where Θ includes $\{\mathbf{O}_l,\mathbf{U},\widetilde{\mathbf{L}},\mathbf{E}_l^{\mathcal{G}},\mathbf{E}_l^{\mathcal{F}}\}$, $\|\cdot\|_{\mathcal{X}}$ denotes the penalty term corresponding to the distribution assumptions of either $\mathcal{N}(\cdot)$ or $\mathcal{L}(\cdot)$.

We summarize the superiority of the GProM framework:

1) *Interpretable*: From the perspective of probabilistic graphical models, it elucidates the dependencies among view-related variables, commonality, and two levels of noises separated from observations and latent variables.
2) *Concise*: It directly models cross-view consistency by relying solely on latent variables associated with multiple views, without requiring additional components.

3) *Flexible*: It accommodates various distribution assumptions and combinations, enabling flexible transformation into a data-driven noise pruning mechanism.

Remark 3: Our GProM offers four options, 1) GProM-GG: both $\mathbf{e}_l^{\mathcal{F}}$ and $\mathbf{e}_l^{\mathcal{G}}$ follow $\mathcal{N}(\cdot)$. 2) GProM-LG: $\mathbf{e}_l^{\mathcal{F}}$ follows $\mathcal{L}(\cdot)$ while $\mathbf{e}_l^{\mathcal{G}}$ follows $\mathcal{N}(\cdot)$. 3) GProM-GL: $\mathbf{e}_l^{\mathcal{F}}$ follows $\mathcal{N}(\cdot)$ while $\mathbf{e}_l^{\mathcal{G}}$ follows $\mathcal{L}(\cdot)$. 4) GProM-LL: both $\mathbf{e}_l^{\mathcal{F}}$ and $\mathbf{e}_l^{\mathcal{G}}$ follow $\mathcal{L}(\cdot)$.

IV. Tailor-Made ADMM Optimization

For the first expectation term of (7), existing gradient-based solutions, such as Stochastic Gradient Descent (SGD) can solve it. What makes our problem difficult or even intractable lies in the second regularization term $\Phi(\Theta)$. Conventional MLE solutions will suffer failures as they may cause variables beyond the constraint boundaries. These difficulties prompt us to design a tailor-made but efficient solution. Considering the coupling relationship of variables and constraints, we adopt Alternating Direction Method of Multipliers (ADMM) [56] to optimize our problem. Without loss of generality, we take GProM-LL as an example, where the penalty $\|\cdot\|_{\chi}$ is specified as the $\|\cdot\|_{2,1}$ norm. The solutions for the remaining variants follow a similar procedure.

In the ADMM setting, we introduce auxiliary $\widetilde{\mathbf{E}}_l^{\mathcal{G}}$ for $\mathbf{E}_l^{\mathcal{G}}$ to decouple its constraint and distribution assumptions. For $\phi(\widetilde{\mathbf{L}})$, Ky Fan's Theorem [57] implies that the solution is to enforce the rank-k smallest $\sigma_{\varrho}(\widetilde{\mathbf{L}})$ equal to zero, namely $\sum_{\varrho=1}^{k}\sigma_{\varrho}(\widetilde{\mathbf{L}}) = \min_{\mathbf{R}^\top\mathbf{R}=\mathbf{I}_k}\mathrm{Tr}(\mathbf{R}^\top\widetilde{\mathbf{L}}\mathbf{R})$, where $\mathbf{R}$ is the graph embedding. Therefore, (7) is transformed into:

$$\begin{aligned}\min_{\substack{\{\mathbf{E}_l^{\mathcal{F}},\mathbf{E}_l^{\mathcal{G}},\widetilde{\mathbf{E}}_l^{\mathcal{G}},\mathbf{O}_l\}_{l=1}^v,\\ \mathbf{U},\mathbf{R}}}\ &\sum_{l=1}^{v}\left(\left\|\mathbf{E}_l^{\mathcal{F}}\right\|_{2,1}+\eta\left\|\widetilde{\mathbf{E}}_l^{\mathcal{G}}\right\|_{2,1}\right)\\ &+\frac{\gamma}{2}\sum_{l=1}^{v}\left\|\mathbf{E}_l^{\mathcal{G}}-\widetilde{\mathbf{E}}_l^{\mathcal{G}}+\frac{1}{\gamma}\boldsymbol{\Upsilon}_l\right\|_F^2+\vartheta\mathrm{Tr}\left(\mathbf{R}^\top\widetilde{\mathbf{L}}\mathbf{R}\right)\\ &+\frac{\lambda}{2}\sum_{l=1}^{v}\left\|\mathbf{X}_l-\mathbf{O}_l(\xi\mathbf{U}+\beta\mathbf{E}_l^{\mathcal{G}})-\mathbf{E}_l^{\mathcal{F}}+\frac{1}{\lambda}\boldsymbol{\Lambda}_l\right\|_F^2,\\ \text{s.t.}\ &\begin{cases}\mathbf{O}_l^\top\mathbf{O}_l=\mathbf{I}_m;\ \mathbf{R}^\top\mathbf{R}=\mathbf{I}_k,\\ \mathbf{E}_l^{\mathcal{G}\top}\mathbf{1}_m=\mathbf{1}_n,\ \mathbf{E}_l^{\mathcal{G}}\geq 0,\\ \mathbf{U}^\top\mathbf{1}_m=\mathbf{1}_n,\ \mathbf{U}\geq 0,\end{cases}\end{aligned}\tag{8}$$

where $\boldsymbol{\Lambda}_l$ and $\boldsymbol{\Upsilon}_l$ are Augmented Lagrangian Method (ALM) multipliers to penalize the discrepancy between the auxiliary variables and original objective, λ and γ are ALM parameters, the distribution-related parameter is set to $\eta = 1$ for simplicity, ϑ is a penalty parameter that iteratively increases to enforce the rank-k smallest $\sigma_{\varrho}(\widetilde{\mathbf{L}})$ infinitely approach 0.

Eq. (8) can be optimized by a block-coordinate descent strategy [58]. Algorithm 1 outlines the overall workflow.

Update $\mathbf{E}_l^{\mathcal{F}}$: Each $\mathbf{E}_l^{\mathcal{F}}$ is independently solved by:

$$\min_{\mathbf{E}_l^{\mathcal{F}}}\ \frac{1}{\lambda}\left\|\mathbf{E}_l^{\mathcal{F}}\right\|_{2,1}+\frac{1}{2}\left\|\mathbf{Q}_l-\mathbf{E}_l^{\mathcal{F}}\right\|_F^2,\tag{9}$$

where $\mathbf{Q}_l = \mathbf{X}_l - \mathbf{O}_l(\xi\mathbf{U}+\beta\mathbf{E}_l^{\mathcal{G}})+\frac{1}{\lambda}\boldsymbol{\Lambda}_l$. The solution [59] is $\mathbf{E}_{l[:,j]}^{\mathcal{F}} = \begin{cases}\left(1-\frac{1}{\lambda\|\mathbf{Q}_{l[:,j]}\|_2}\right)\mathbf{Q}_{l[:,j]}, & if\ \|\mathbf{Q}_{l[:,j]}\|_2>\frac{1}{\lambda},\\ 0, & otherwise.\end{cases}$

Update $\widetilde{\mathbf{E}}_l^{\mathcal{G}}$: Each $\widetilde{\mathbf{E}}_l^{\mathcal{G}}$ is separately optimized by:

$$\min_{\widetilde{\mathbf{E}}_l^{\mathcal{G}}}\ \frac{\eta}{\gamma}\left\|\widetilde{\mathbf{E}}_l^{\mathcal{G}}\right\|_{2,1}+\frac{1}{2}\left\|\mathbf{N}_l-\widetilde{\mathbf{E}}_l^{\mathcal{G}}\right\|_F^2,\tag{10}$$

where $\mathbf{N}_l = \mathbf{E}_l^{\mathcal{G}}+\frac{1}{\gamma}\boldsymbol{\Upsilon}_l$. Similarly, we can obtain the solution: $\widetilde{\mathbf{E}}_l^{\mathcal{G}} = \begin{cases}(1-\frac{\eta}{\gamma\|\mathbf{N}_{l[:,j]}\|_2})\mathbf{N}_{l[:,j]}, & if\ \|\mathbf{N}_{l[:,j]}\|_2>\frac{\eta}{\gamma},\\ 0, & otherwise.\end{cases}$

Update $\mathbf{O}_l$: The problem w.r.t. $\mathbf{O}_l$ is: $\max_{\mathbf{O}_l}\ \mathrm{Tr}(\mathbf{O}_l^\top\mathbf{A}_l)$, s.t. $\mathbf{O}_l^\top\mathbf{O}_l=\mathbf{I}_m$, where $\mathbf{A}_l = (\mathbf{X}_l-\mathbf{E}_l^{\mathcal{F}}+\frac{1}{\lambda}\boldsymbol{\Lambda}_l)(\xi\mathbf{U}+\beta\mathbf{E}_l^{\mathcal{G}})^\top$. This problem can be solved by SVD [42].

Update $\mathbf{E}_l^{\mathcal{G}}$: Each $\mathbf{E}_{l[:,j]}^{\mathcal{G}}$ is individually updated via: $\min_{\mathbf{E}_{l[:,j]}^{\mathcal{G}}}\frac{1}{2}\|\mathbf{E}_{l[:,j]}^{\mathcal{G}}-\widehat{\mathbf{E}}_{l[:,j]}^{\mathcal{G}}\|_2^2$, s.t. $\mathbf{E}_{l[:,j]}^{\mathcal{G}\top}\mathbf{1}=1, \mathbf{E}_{l[:,j]}^{\mathcal{G}}\geq 0$, where $\widehat{\mathbf{E}}_{l[:,j]}^{\mathcal{G}} = -\frac{\mathrm{f}^\top}{2(\lambda\beta^2+\gamma)}$, $\mathrm{f} = -2(\lambda\beta\boldsymbol{\Omega}_l^\top\mathbf{O}_l+\gamma\mathbf{M}_l^\top)_{[\mathrm{j},:]}$, $\boldsymbol{\Omega}_l = \mathbf{X}_l-\xi\mathbf{O}_l\mathbf{U}-\mathbf{E}_l^{\mathcal{F}}+\frac{1}{\lambda}\boldsymbol{\Lambda}_l$, and $\mathbf{M}_l = \widetilde{\mathbf{E}}_l^{\mathcal{G}}-\frac{1}{\gamma}\boldsymbol{\Upsilon}_l$. The analytical solution is in supplementary materials (Section 3).

Update $\mathbf{U}$ and $\mathbf{R}$: With others being fixed, we have:

$$\begin{aligned}\min_{\mathbf{U},\mathbf{R}}\ &\mathrm{Tr}\left(\mathbf{U}^\top\left(\sum_{l=1}^{v}\frac{\lambda}{2}\xi^2\mathbf{I}\right)\mathbf{U}-\left(\xi\lambda\sum_{l=1}^{v}\mathbf{G}_l^\top\mathbf{O}_l\right)\mathbf{U}\right)\\ &+\vartheta\mathrm{Tr}\left(\mathbf{R}^\top\widetilde{\mathbf{L}}\mathbf{R}\right),\ \text{s.t.}\ \mathbf{U}^\top\mathbf{1}_m=\mathbf{1}_n,\ \mathbf{U}\geq 0;\ \mathbf{R}^\top\mathbf{R}=\mathbf{I}_k,\end{aligned}\tag{11}$$

where $\mathbf{G}_l = \mathbf{X}_l-\beta\mathbf{O}_l\mathbf{E}_l^{\mathcal{G}}-\mathbf{E}_l^{\mathcal{F}}+\frac{1}{\lambda}\boldsymbol{\Lambda}_l$. Details are in supplementary materials (Section 4).

While fixing $\mathbf{U}$, $\mathbf{R} = [\mathbf{R}_n^\top\ \mathbf{R}_m^\top]^\top$ can be updated via:

$$\begin{aligned}\max_{\mathbf{R}_n,\mathbf{R}_m}\ &\mathrm{Tr}\left(\mathbf{R}_n^\top\boldsymbol{\Omega}_n^{-\frac{1}{2}}\mathbf{U}^\top\boldsymbol{\Omega}_m^{-\frac{1}{2}}\mathbf{R}_m\right),\\ \text{s.t.}\ &\mathbf{R}_n^\top\mathbf{R}_n+\mathbf{R}_m^\top\mathbf{R}_m=\mathbf{I}_k.\end{aligned}\tag{12}$$

The optimal solutions are $\mathbf{R}_n = \frac{\sqrt{2}}{2}\boldsymbol{\Sigma}$ and $\mathbf{R}_m = \frac{\sqrt{2}}{2}\boldsymbol{\Gamma}$, where $\boldsymbol{\Sigma}$ and $\boldsymbol{\Gamma}$ are the rank-k left and right singular matrices of $\boldsymbol{\Omega}_n^{-\frac{1}{2}}\mathbf{U}^\top\boldsymbol{\Omega}_m^{-\frac{1}{2}}$.

While fixing $\mathbf{R}$, each $\mathbf{U}_{[:,j]}$ is solved by:

$$\min_{\mathbf{U}_{[:,j]}}\ \frac{1}{2}\left\|\mathbf{U}_{[:,j]}-\widetilde{\mathbf{U}}_{[:,j]}\right\|_2^2,\ \text{s.t.}\ \mathbf{U}_{[:,j]}^\top\mathbf{1}=1,\ \mathbf{U}_{[:,j]}\geq 0,\tag{13}$$

where $\widetilde{\mathbf{U}}_{[:,j]} = -\frac{\mathbf{r}^\top}{v\lambda\xi^2}$, $\mathbf{r} = -(\mathbf{J}-\vartheta\mathbf{T})_{[j,:]}$, $\mathbf{J} = \sum_{l=1}^{v}\xi\lambda\mathbf{G}_l^\top\mathbf{O}_l$, $t_{ij} = \|\frac{\mathbf{r}_n^i}{\sqrt{\boldsymbol{\Omega}_{n[i,i]}}}-\frac{\mathbf{r}_m^j}{\sqrt{\boldsymbol{\Omega}_{m[j,j]}}}\|_2^2$.

Update ALM multipliers:

$$\begin{aligned}\boldsymbol{\Lambda}_l &= \boldsymbol{\Lambda}_l+\lambda\left(\mathbf{X}_l-\mathbf{O}_l(\xi\mathbf{U}+\beta\mathbf{E}_l^{\mathcal{G}})-\mathbf{E}_l^{\mathcal{F}}\right),\\ \boldsymbol{\Upsilon}_l &= \boldsymbol{\Upsilon}_l+\gamma\left(\mathbf{E}_l^{\mathcal{G}}-\widetilde{\mathbf{E}}_l^{\mathcal{G}}\right),\\ \lambda &= \rho\lambda,\ \gamma = \rho\gamma,\end{aligned}\tag{14}$$

where ρ is a scaling parameter.

Algorithm 1: Workflow for GProM With $\mathcal{L}(\cdot)$ Distribution.

Input: $\{\mathbf{X}_l\}_{l=1}^v$.
Initialize $\mathbf{O}_l$, $\mathbf{E}_l^{\mathcal{G}}$, $\widetilde{\mathbf{E}}_l^{\mathcal{G}}$, $\mathbf{\Lambda}_l$, $\mathbf{\Upsilon}_l$, $\mathbf{U}$, λ, γ, ρ.
while not converged **do**
  Update unconstrained $\mathbf{E}_l^{\mathcal{F}}$ and $\widetilde{\mathbf{E}}_l^{\mathcal{G}}$, respectively.
  Update constrained $\mathbf{O}_l$, $\mathbf{E}_l^{\mathcal{G}}$, and $\mathbf{U}$, respectively.
  Update ALM multipliers $\mathbf{\Lambda}_l$, $\mathbf{\Upsilon}_l$, λ, γ.
end while
Output: The predicted clustering labels.

A. Parameter Initialization and Convergence Criterion

1) Parameter Initialization: (1) $\mathbf{O}_l$ is initialized as the centroids obtained from applying k-means to the left singular vectors of the lth view data. (2) $\mathbf{U}$ is initialized by applying truncated SVD to the concatenated multi-view data to obtain a low-dimensional embedding, followed by k-means and one-hot encoding, and scaling transformation to the resulting matrix. (3) $\mathbf{E}_l^{\mathcal{G}}$ and auxiliary variable $\widetilde{\mathbf{E}}_l^{\mathcal{G}}$ are initialized to $\mathbf{U}$. (4) The ALM multipliers $\mathbf{\Lambda}_l$ and $\mathbf{\Upsilon}_l$ are initialized to zero. (5) The penalty parameters λ and γ are initialized to 1, the scaling parameterρ is initialized to 1.5. (6) The balanced parameter η is set to 1.

2) Convergence Criterion: The convergence of exact ALM methods for smooth objective function has been extensively studied [60], and the convergence of inexact ALM methods has also been explored [61]. Empirically, ADMM converges to a local optimum under general conditions [62]. However, establishing rigorous mathematical guarantees for ADMM convergence remains a challenging problem [56].

Based on the ADMM framework, the optimization of GProM is decomposed into several subproblems, each of which has a closed-form solution. According to [56], [59], [61], [63], the ALM parameters λ and γ control the convergence speed and generally have little impact on the final solution. Typically, larger λ and γ correspond to faster convergence, although they may incur precision loss in the objective. ρ is a scaling parameter that controls the update of λ and γ. With increasing iterations, the penalty terms in (8) gradually approach zero, and the ALM objective asymptotically converges to the original objective, which converges to a local optimum and is lower-bounded by 0. In experiments, the convergence criterion is set as follows:

$$\text{if } (t > 9) \text{ and } \left(\frac{|obj(t-1) - obj(t)|}{obj(t-1)} < 10^{-5} \text{ or } t > 30 \right.$$
$$\left. \text{or } obj(t) < 10^{-10} \right), \tag{15}$$

where t is the iteration step, and $obj(t)$ is the corresponding objective value.

B. Theoretical Property

Probability Perspective for Bipartite Graph: Theorem 2 provides a probability perspective for recovering instance-instance memberships $w_{ij} = p(\mathbf{x}_{l[i]} \mapsto \mathbf{x}_{l[j]})$ with instance-prototype memberships $z_{ri} = p(\mathbf{x}_{l[i]} \mapsto \mathbf{o}_{l[r]})$ and $z_{rj} = p(\mathbf{o}_{l[r]} \mapsto \mathbf{x}_{l[j]})$ [64]. Details are available in supplementary materials (Section 5).

Theorem 2: With the one-step transition probability matrix $\mathbf{P} = \mathbf{\Pi}^{-1}\mathbf{S}$, the graph embedding of the normalized Laplacian matrix $\widetilde{\mathbf{L}}$ is equivalent to the graph embedding of the full graph $\mathbf{W}$ that are derived from the double-step transition probability.

Complexity Analysis: For simplicity, we set $\delta = \sum_{l=1}^v d_l$, where d_l denotes the feature dimension of the lth view.

Time Complexity: The time complexity comes from ADMM optimization, as shown in Algorithm 1. Each option within our GProM shares the comparable time complexity $\mathcal{O}(n(m(\delta + m + v) + k) + \delta m^2 + mk)$.

Space Complexity: The space complexity primarily arises from storing large matrices, including $\{\mathbf{X}_l, \mathbf{O}_l, \mathbf{E}_l^{\mathcal{F}}, \mathbf{E}_l^{\mathcal{G}}, \widetilde{\mathbf{E}}_l^{\mathcal{G}}, \mathbf{\Lambda}_l, \mathbf{\Upsilon}_l\}_{l=1}^v$, $\mathbf{U}$, and $\mathbf{R}$, leading to a total memory requirement $\mathcal{O}(n(\delta + mv + k) + m(\delta + k))$.

Commonly, $n \gg m$, $n \gg k$. δ, v, and k are dataset-related information, which are constants in optimization. Therefore, both the time and space complexity of GProM are linear w.r.t. n, i.e., $\mathcal{O}(n)$, making it feasible to scale to large-scale datasets with $n \geq 100,000$.

TABLE II
13 REAL-WORLD DATASETS

Size	Dataset	Instance	View	Feature	Cluster
Small	Dermatology	358	2	12 / 22	6
	Mfeat	2,000	2	76 / 240	10
	VGGF2_50	16,936	4	944 / 576 / 512 / 640	50
	Caltech256	30,607	4	944 / 576 / 512 / 640	257
	VGGF2_100	36,287	4	944 / 576 / 512 / 640	100
Medium	CIFAR100-T	50,000	4	944 / 576 / 512 / 640	100
	CIFAR10-T	50,000	4	944 / 576 / 512 / 640	10
	CIFAR100	60,000	4	944 / 576 / 512 / 640	100
	CIFAR10	60,000	4	944 / 576 / 512 / 640	10
	YTF20	63,896	4	944 / 576 / 512 / 640	20
	VGGF2_200	72,283	4	944 / 576 / 512 / 640	200
Large	T-ImageNet	100,000	4	944 / 576 / 512 / 640	200
	EMNIST-D	280,000	4	944 / 576 / 512 / 640	10

V. EXPERIMENTS AND ANALYSIS

A. Benchmark Datasets and Compared Baselines

Table II records the 13 public real-world datasets: (1) 5 small-scale datasets: Dermatology, Mfeat, VGGF2_50, Caltech256, VGGF2_100; (2) 6 medium-scale datasets: CIFAR100-T, CIFAR10-T, CIFAR100, CIFAR10, YTF20, VGGF2_200; (3) 2 large-scale datasets: T-ImageNet, EMNIST-D. All these datasets are from the public websites.

We collected 18 baselines: (1) 4 MVC baselines: RMKM [65], AMGL [66], FMR [67], PMSC [68]; (2) 5 pioneering MVBGC baselines: BMVC [69], LMVSC [18], SMVSC [37], SFMC [17], FPMVS [38]; (3) 6 recent MVBGC methods: FMCNOF [70], SDAFG [28], UDBGL [71], FastMICE [35], FSMSC [43], PPTL [72]; (4) 3 deep learning-based methods: MVCAN [73], MFLVC [74], DSMVC [75].

B. Implementation Details

For GProM, m varies in $\{k, 2k, 3k, 4k\}$, ξ varies in $\{0.1, 0.2, \ldots, 0.9\}$, and $m \leq \min\{d_l\}_{l=1}^v$.

TABLE III
COMPARISON OF CLUSTERING METRICS (MEAN ± STD) W.R.T. ACC AND NMI ON SMALL- AND MEDIUM-SCALE DATASETS

Datasets	Dermatology	Mfeat	VGGF2_50	Caltech256	VGGF2_100	CIFAR100-T	CIFAR10-T	CIFAR100	CIFAR10	YTF20	VGGF2_200	Avg Rank
	ACC (%)											
RMKM‡	74.86	80.80	8.23	9.87	5.39	9.79	25.00	9.62	27.47	71.10	OOM	10.85
AMGL	22.57±0.59	71.42±5.42	2.95±0.35	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM	19.08
FMR	81.72±5.66	64.53±2.28	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM	19.54
PMSC	80.75±4.46	66.41±4.40	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM	20.08
BMVC‡	63.97	65.80	10.30	8.63	6.17	8.38	27.24	8.32	27.81	57.39	3.99	10.08
LMVSC	79.02±6.63	82.86±6.95	10.56±0.26	9.57±0.17	6.09±0.09	8.92±0.17	30.73±0.39	9.53±0.15	29.02±0.81	67.26±3.53	4.31±0.06	7.31
SMVSC	78.64±5.41	65.57±3.99	13.36±0.60	10.54±0.15	7.90±0.20	8.48±0.18	28.82±1.22	8.34±0.17	29.11±1.35	67.13±4.20	4.25±0.06	8.85
SFMC‡	49.44	66.20	3.64	4.50	1.98	N/A	10.02	1.18	10.02	N/A	N/A	19.77
FMCNOF‡	62.01	56.95	5.51	2.70	3.47	4.40	21.02	3.66	20.53	38.61	0.90	17.08
FPMVS	82.96±7.44	65.11±4.18	12.06±0.36	8.78±0.07	6.02±0.18	7.46±0.12	26.12±0.65	7.29±0.11	26.89±0.71	63.08±3.79	3.41±0.05	11.08
SDAFG‡	56.70	74.20	3.58	6.30	2.30	2.43	13.14	1.94	12.67	61.88	1.74	16.46
UDBGL‡	85.75	84.50	9.35	7.61	5.57	7.65	27.33	8.78	26.01	69.62	1.03	9.77
FastMICE‡	88.55	85.70	10.24	9.61	5.25	9.11	30.34	9.41	29.93	71.08	3.63	6.62
FSMSC	82.68±7.48	75.79±4.62	13.91±0.62	5.89±0.06	9.15±0.32	9.99±0.26	23.57±1.08	9.96±0.21	23.13±0.39	68.99±5.37	1.85±0.02	9.92
PPTL	62.43±6.14	67.87±2.76	7.18±0.20	8.50±0.33	5.08±0.14	5.81±0.12	21.86±0.14	5.87±0.08	20.79±0.16	71.47±4.24	3.34±0.05	13.31
MVCAN	43.18±1.79	91.20±0.09	7.32±0.16	9.64±0.15	5.64±0.06	6.37±0.10	25.81±0.46	6.75±0.14	24.92±0.32	68.95±2.39	3.63±0.05	9.92
DSMVC	62.79±6.81	70.62±7.09	5.04±0.16	4.49±0.22	3.21±0.07	3.99±0.22	20.41±0.82	4.16±0.27	21.36±0.62	72.07±1.82	1.94±0.07	14.38
MFLVC	72.12±8.10	91.11±3.84	8.38±0.29	7.88±0.39	5.43±0.28	4.39±1.05	27.46±1.49	5.99±0.20	27.40±0.48	62.17±3.24	3.44±0.39	11.85
GProM-GG‡	94.41	87.45	11.02	9.50	6.34	7.86	29.31	7.69	29.60	70.56	3.32	7.62
GProM-LG‡	**94.97**	92.90	**14.65**	**12.33**	**9.21**	**10.27**	**31.80**	**10.75**	*31.48*	**74.29**	**5.75**	1.15
GProM-GL‡	94.41	81.90	11.93	9.70	6.72	8.02	29.21	7.97	29.55	73.51	3.45	6.46
GProM-LL‡	**94.97**	**93.25**	*14.58*	*12.23*	*9.20*	**10.27**	*31.79*	**10.75**	**31.61**	*73.96*	*5.73*	1.85
	NMI (%)											
RMKM‡	71.10	82.28	9.66	31.01	11.14	17.80	16.18	17.38	16.50	78.34	OOM	10.77
AMGL	3.20±0.57	77.12±2.19	2.04±0.50	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM	18.46
FMR	79.97±3.67	66.21±0.73	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM	18.92
PMSC	85.11±1.91	63.29±1.63	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM	19.62
BMVC‡	60.79	59.39	13.48	31.83	14.26	15.12	17.82	15.05	17.90	70.65	15.04	8.31
LMVSC	70.17±3.94	82.46±2.79	12.64±0.28	31.96±0.11	11.92±0.11	15.49±0.15	17.80±0.26	15.40±0.18	17.84±0.53	76.78±1.34	13.94±0.09	7.92
SMVSC	66.62±2.66	57.99±2.11	16.21±0.49	28.27±0.24	14.80±0.23	14.83±0.22	15.92±0.91	14.40±0.20	16.00±0.99	78.36±2.39	13.94±0.13	10.62
SFMC‡	38.68	76.02	1.63	5.67	0.91	N/A	0.12	0.53	0.16	N/A	N/A	19.46
FMCNOF‡	54.24	55.47	4.74	0.00	5.81	8.49	11.07	7.04	10.33	45.45	0.00	17.85
FPMVS	71.90±5.05	57.77±2.73	14.74±0.55	22.97±0.21	12.33±0.29	13.87±0.21	15.06±1.07	13.62±0.16	15.45±0.99	74.30±1.95	11.18±0.18	12.08
SDAFG‡	51.61	74.70	2.01	13.43	3.03	5.06	6.35	3.15	5.10	73.18	5.26	15.85
UDBGL‡	**90.18**	82.25	11.24	22.67	11.28	14.28	16.77	14.89	*22.35*	80.49	0.74	8.46
FastMICE‡	84.96	85.48	12.20	*32.38*	10.49	16.47	18.17	16.50	17.93	79.34	12.86	5.92
FSMSC	82.19±3.05	70.50±2.83	16.73±0.52	19.59±0.11	16.46±0.33	16.33±0.25	11.21±0.46	16.01±0.17	10.77±0.24	78.58±1.76	7.48±0.07	10.31
PPTL	47.50±2.45	61.23±1.05	7.92±0.17	30.11±0.11	9.65±0.18	9.89±0.13	8.96±0.09	9.50±0.08	7.60±0.02	78.42±1.35	11.01±0.05	14.00
MVCAN	43.21±2.05	84.54±0.15	9.64±0.66	**33.12±0.10**	12.61±0.11	13.28±0.08	17.21±0.25	13.70±0.17	17.55±0.16	79.04±0.67	13.70±0.01	8.31
DSMVC	75.47±3.35	73.05±2.92	3.74±0.18	17.92±0.50	4.87±0.12	7.53±0.65	9.07±0.96	7.62±0.90	9.90±0.28	**81.30±0.27**	4.82±0.28	13.62
MFLVC	66.03±3.68	85.72±3.17	10.72±0.57	27.84±1.07	12.09±0.57	9.37±2.42	18.20±0.27	12.20±1.00	18.31±0.36	71.54±1.89	12.23±1.43	10.92
GProM-GG‡	*89.25*	79.16	14.21	21.81	12.24	15.37	17.12	14.34	17.16	79.32	9.91	9.15
GProM-LG‡	89.20	*85.89*	**18.00**	30.98	*17.26*	*18.34*	*19.39*	**18.53**	**23.05**	80.35	**17.22**	2.31
GProM-GL‡	*89.25*	75.20	15.03	23.88	13.19	14.78	17.93	14.86	18.02	79.48	10.82	7.62
GProM-LL‡	89.20	**86.63**	*17.93*	30.86	**17.28**	**18.41**	**19.41**	**18.53**	22.06	*80.67*	*17.18*	2.54

• OOM denotes out-of-memory errors, N/A denotes unavailable results caused by extremely long runtime.
• ‡ denotes stable algorithm without performance variance.

The codes for the compared baselines are collected from public websites. Their hyperparameters are tuned according to the original settings, and we report the best metrics. For baselines requiring k-means, we randomly initialize clustering centroids 30 times and report the average results (mean ± std) to alleviate randomness [76], [77]. The cluster number k is assumed to be pre-determined following existing research [78], [79], [80].

Four widely used clustering metrics, including ACC (accuracy), NMI (normalized mutual information), Purity, and F-score [81], [82], [83], [84], are used to evaluate clustering performance.

Experimental results for the 3 deep learning baselines are obtained using a server with one NVIDIA A100 GPU (80G) and the PyTorch platform. While for the rest 15 baselines, experiments were performed on a server with 6 core Intel(R) i9-9900 K CPUs @3.6 GHZ, 64 GB RAM, and Matlab 2020b.

C. Comparison of Clustering Metrics

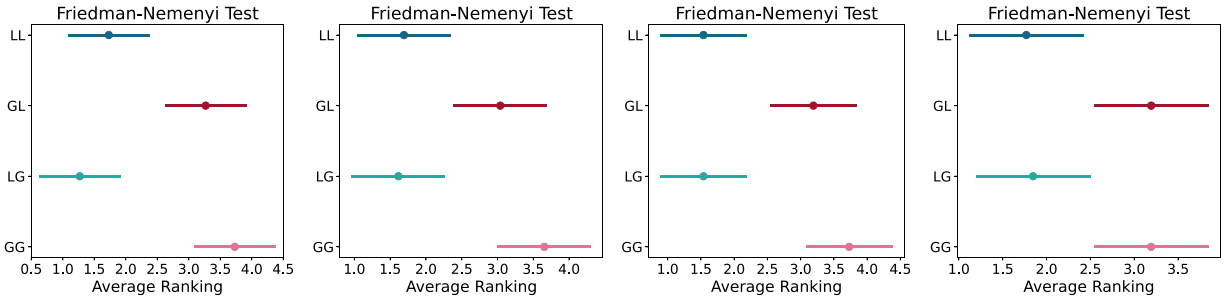
Table III presents the clustering metrics in terms of ACC and NMI on small- and medium-scale datasets, with additional metrics provided in supplementary materials (Section 7). Table IV summarizes the results on large-scale datasets. We also calculate the average ranking (Avg Rank) for a clearer comparison. From these results, we observe that:

1) The four MVC baselines encounter severe out-of-memory (OOM) issues, particularly FMR and PMSC, which can only handle small-scale datasets. This limitation arises from their quadratic space complexity, $\mathcal{O}(n^2)$.
2) The eleven MVBGC baselines are theoretically capable of handling large-scale datasets, given their linear complexity $\mathcal{O}(n)$. However, pioneering methods like SMVSC and SFMC still yield unavailable results ("N/A"), which is due to excessively long running time. Specifically, SMVSC fails to tackle EMNIST-D dataset because of the computationally expensive quadratic programming in solving the $\mathbf{U}$ sub-optimization. While for SFMC, the unavailable results are caused by slow convergence in solving the $\mathbf{R}$ sub-optimization.
3) Compared to the three recent deep learning baselines designed to address noise-related issues, GProM achieves competitive performance. Moreover, its explicit formulation and optimization make it inherently more interpretable than deep learning counterparts. Unlike these methods, which are sensitive to seed selection, GProM achieves stable performance without performance variance.

TABLE IV
COMPARISON OF CLUSTERING METRICS ON LARGE-SCALE DATASETS (MEAN ± STD)

	T-ImageNet	EMNIST-D	T-ImageNet	EMNIST-D	T-ImageNet	EMNIST-D	T-ImageNet	EMNIST-D
	ACC (%)		NMI (%)		Purity (%)		F-score (%)	
BMVC	4.09	68.99	*13.75*	70.08	4.69	71.38	1.55	61.38
LMVSC	4.28±0.05	61.75±4.05	13.23±0.06	61.87±2.47	*4.93±0.04*	65.08±2.93	1.36±0.01	54.07±3.51
SMVSC	3.49±0.04	N/A	11.95±0.12	N/A	3.73±0.04	N/A	1.72±0.01	N/A
FMCNOF‡	0.50	36.50	OOM	28.36	0.50	36.90	0.99	25.64
FPMVS	2.83±0.02	62.24±4.08	10.04±0.12	53.47±2.44	2.95±0.02	62.30±4.04	1.50±0.00	49.29±3.01
SDAFG‡	1.54	61.54	6.13	72.73	1.63	66.12	1.03	57.93
UDBGL‡	3.48	72.09	11.83	69.18	3.75	72.09	1.17	62.04
FastMICE‡	3.86	70.67	12.65	73.94	4.50	74.73	1.31	65.50
FSMSC	1.84±0.01	21.51±0.61	8.35±0.04	18.98±0.80	1.99±0.01	23.26±0.63	1.02±0.00	18.42±0.03
PPTL	3.43±0.05	51.50±0.90	10.52±0.06	43.46±0.44	3.82±0.05	52.41±0.68	1.11±0.02	38.42±0.44
MVCAN	4.42±0.05	76.31±2.93	**14.34±0.07**	73.63±1.68	**5.20±0.09**	78.05±2.25	0.43±0.05	2.80±7.75
DSMVC	2.59±0.08	66.97±7.64	9.17±0.35	66.54±7.30	2.83±0.08	67.24±7.56	0.23±0.03	3.78±3.83
MFLVC	N/A	72.12±8.10	N/A	66.03±3.68	N/A	77.10±4.34	N/A	14.81±13.52
GProM-GG‡	2.99	70.55	10.45	64.64	3.08	70.55	1.47	54.09
GProM-LG‡	**4.44**	*78.77*	13.36	**74.03**	4.75	**78.77**	*1.79*	**68.26**
GProM-GL‡	3.12	74.48	10.80	67.84	3.23	74.48	1.45	61.95
GProM-LL‡	*4.43*	78.73	13.30	*74.01*	4.75	*78.73*	**1.80**	*68.20*

• N/A denotes unavailable results caused by extremely long runtime.
• ‡ denotes stable algorithm without performance variance.



(a) ACC (b) NMI (c) Purity (d) F-score

Fig. 2. Friedman-Nemenyi test comparing different noise distribution assumptions at a significance level of $\alpha = 0.05$. The x-axis shows the average rankings of the four GProM variants, and the y-axis corresponds to their IDs.

4) GProM-LG and GProM-LL achieve competitive overall ranking, outperforming both the other two GProM options and the compared baselines. In particular, GProM-LG (ACC: 1.15) attains the top rank in all metrics, demonstrating its superior overall performance. Among the baselines, FastMICE (ACC: 6.62) and LMVSC (ACC: 7.31) emerge as the strongest competitors, while SFMC (ACC: 19.77) and PMSC (ACC: 20.08) show the weakest performance on this metric.

In summary, our GProM exhibits highly competitive performance compared to eighteen baselines, along with strong scalability and stability.

D. Statistical Significance Analysis Under Different Distribution Assumptions

We utilize the Friedman-Nemenyi test [85] to evaluate the significance of different distribution assumptions. The Friedman test assesses whether there are statistical significance among the candidates, while the Nemenyi post-hoc test identifies which specific pairs differ significantly. Note that this test does not impose specific requirements on the data distribution. Further technical details are provided in supplementary materials (Section 8.1).

In our case, the null hypothesis assumes that there is no significant difference among the four options. The calculated Friedman test statistics (τ_F) are 64.24, 18.87, 39.87, and 7.48 for ACC, NMI, Purity, and F-score, respectively. Since all of these values exceed the threshold of 2.866 at the 0.05 significance level ($\alpha = 0.05$), we reject the null hypothesis. This result indicates that the performance among the four options statistically significant.

Fig. 2 presents the results of the Friedman–Nemenyi test, based on a critical distance (CD) of 2.569 at the 0.05 significance level ($\alpha = 0.05$). We note that:

1) GProM-LG achieves the top-1 average rank, followed by GProM-LL, GProM-GL, and GProM-GG.
2) Significant differences are observed between the GProM-LG/GG and GProM-LL/GL pairs, with GProM-LG and GProM-LL consistently outperforming their counterparts.

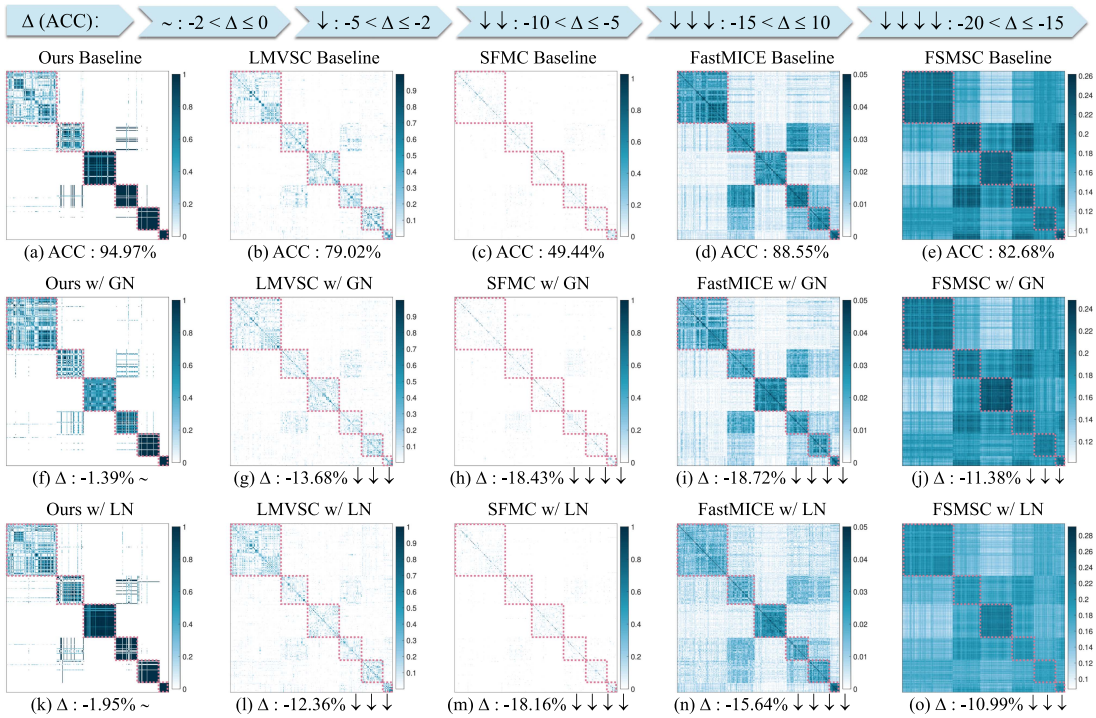


Fig. 3. Visualizing the significance of pruning feature-level $\mathbf{E}_l^{\mathcal{F}}$ on Dermatology dataset. The second view is corrupted with random noise sampled from either $\mathcal{N}(\cdot)$ distribution (denoted as w/ GN) or $\mathcal{L}(\cdot)$ distribution (denoted as w/ LN). The top row displays the performance of our GProM alongside four baselines, while the subsequent rows depict the performance drop Δ(ACC). The notation $\sim$: $-2 < \Delta \leq 0$ indicates a slight decline, whereas more $\downarrow$ represent greater performance degradation.

This suggests that the $\mathcal{L}(\cdot)$-based penalty is more effective than the $\mathcal{N}(\cdot)$-based one in regularizing $\mathbf{E}_l^{\mathcal{F}}$. A possible explanation is that raw features may contain large-magnitude noise or redundancies, and $\mathcal{L}(\cdot)$-based penalty is more effective in mitigating their adverse impact compared to the $\mathcal{N}(\cdot)$-based penalty.

3) Conversely, GProM-LG/LL pair overlaps with each other, as well as GProM-GG/GL pair, suggesting there is no significant differences between $\mathcal{L}(\cdot)$ or $\mathcal{N}(\cdot)$ distributions for $\mathbf{E}_l^{\mathcal{G}}$. This may be attributed to the candidate graphs $\{\mathbf{H}_l\}_{l=1}^v$ are constructed from refined features, resulting in $\mathbf{E}_l^{\mathcal{G}}$ are with small magnitudes. As a consequence, these distributions exhibit comparable significance.

Remark 4: The statistical significance analysis provides suggestions for selecting appropriate distribution assumptions. In summary, the $\mathcal{L}(\cdot)$ distribution is preferred for penalizing $\mathbf{E}_l^{\mathcal{F}}$, whereas both $\mathcal{G}(\cdot)$ and $\mathcal{L}(\cdot)$ distributions yield comparable performance for penalizing $\mathbf{E}_l^{\mathcal{G}}$. More exploration on designing data-driven distributions is meaningful. Unless otherwise specified, the following results are based on the strongest GProM-LG option.

Additional results under a significance level of $\alpha = 0.1$ are provided in supplementary materials (Section 8.2), and a visual comparison of the four GProM options is presented in supplementary materials (Section 9).

E. Significance of Pruning Feature-Level Noise

This section evaluates the significance of pruning feature-level noise $\mathbf{E}_l^{\mathcal{F}}$. We introduce random noise sampled from either $\mathcal{N}(\cdot)$ or $\mathcal{L}(\cdot)$ distributions to contaminate the second view of the Dermatology dataset. Specifically, each instance $\mathbf{x}_{[j]}$ is perturbed by additive noise. The Gaussian noise is drawn from a distribution with zero mean $\boldsymbol{\mu} = \mathbf{0}$ and isotropic covariance $\sigma^2\mathbf{I}$, where $\sigma^2 = 0.3$, that is, $\mathcal{N}(\mathbf{0}, 0.3\mathbf{I})$. The rotational invariant Laplacian noise is generated with a location parameter $\boldsymbol{\mu} = \mathbf{0}$ and a scale parameter $b = 2.5$, i.e., $\mathcal{L}(\mathbf{0}, 2.5\mathbf{I})$. GProM is compared against two pioneering models (LMVSC and SFMC) and two recent methods (FastMICE and FSMSC).

Fig. 3 reports baseline performance and the degraded amplitude Δ(ACC). The results show that the four competitors suffer substantial performance drops, with declines ranging from -18.72% to -10.99%, indicating they are struggle to withstand the effects of feature noise. Conversely, our GProM model exhibits promising robustness, with a maximum drop amplitude of only -1.95%, which is substantially lower than those of the competitors. These findings provide compelling evidence to support the necessity and effectiveness of pruning feature-level noise.

F. Significance of Pruning Structure-Level Noise

This section evaluates the significance of pruning structure-level noise $\mathbf{E}_l^{\mathcal{G}}$.

Fig. 4 visualizes the view-specific affinity matrix derived from three parts: the unfiltered bipartite graph $\mathbf{U} + \mathbf{E}_l^{\mathcal{G}}$, the view-specific structural noise $\mathbf{E}_l^{\mathcal{G}}$, and the consensus refined bipartite graph $\mathbf{U}$. As shown, different views exhibit varying levels of structural noise, which destroy the graph structure. By pruning these noise, the resulting bipartite graph exhibit

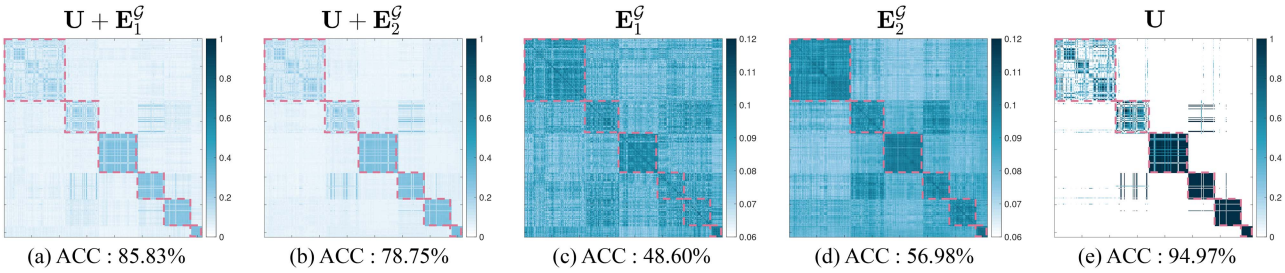


Fig. 4. Visualization on Dermatology dataset demonstrating the necessity of pruning structure-level noise $\mathbf{E}_l^{\mathcal{G}}$. (a)-(b) display the affinity matrix derived from the unfiltered bipartite graph, (c)-(d) plot view-related structural noise, and (e) presents the affinity matrix from the refined bipartite graph after noise pruning.

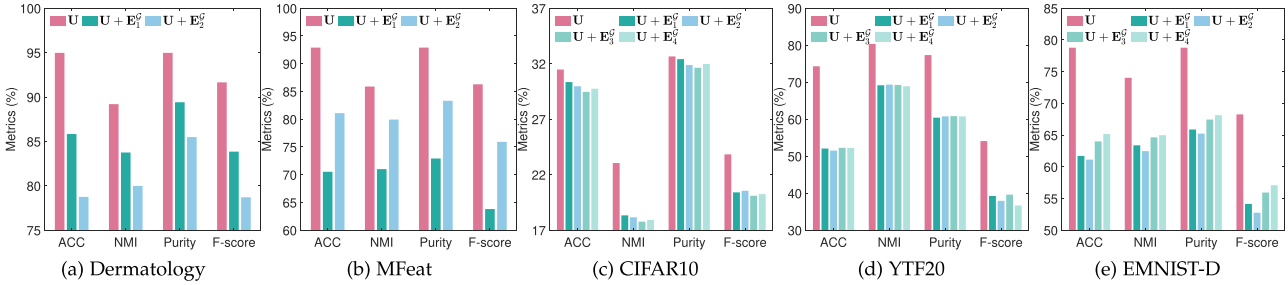


Fig. 5. Quantitative evaluation of the importance of pruning structure-level $\mathbf{E}_l^{\mathcal{G}}$. The refined bipartite graph $\mathbf{U}$ achieves higher performance over the unfiltered one.

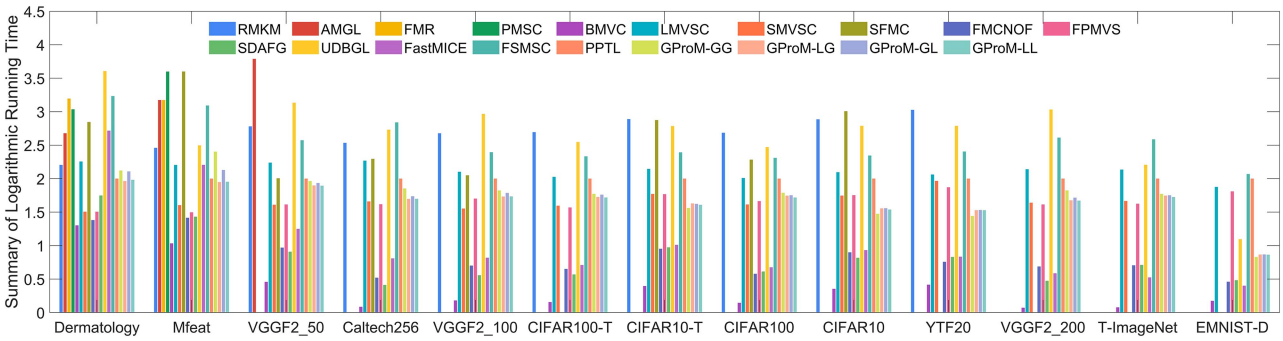


Fig. 6. Summary of the relative logarithm runtime, using PPTL as the baseline for scaling.

clearer and more distinct diagonal block structures. Furthermore, Fig. 5 quantifies the performance improvements achieved by the refined bipartite graph, compared to its degraded counterparts in which structural noise is not pruned. These results further highlight the importance and effectiveness of pruning structure-level noise.

G. *Efficiency*

Fig. 6 reports the relative logarithm runtime, which enables a comparison that is independent of runtime scale differences caused by varying algorithm types and dataset sizes. Specifically, we apply a logarithmic transformation to the absolute runtime and set the log-transformed time of PPTL to 1 as the baseline. The runtime of the other methods are scaled accordingly. We have the following observations:

1) The four MVC baselines, namely RMKM, AMGL, FMR, and PMSC, require significantly more runtime over MVBGC models and are prone to "OOM" errors on medium-scale datasets ($n \geq 72,283$). This is primarily due to their $\mathcal{O}(n^2)$ space complexity.
2) As a pioneering method in addressing scalability issue, BMVC costs the least runtime among MVBGC baselines due to its binary code modeling. However, it relies on random sampling for prototype selection, making its performance sensitive to sampling. Similarly, FMCNOF and FastMICE also use random sampling. Despite their efficient optimization, they are hindered by inflexible prototypes and unstable performance.
3) Four pioneering MVBGC baselines, including LMVSC, SMVSC, SFMC, and FPMVS, and three recent MVBGC baselines, UDBGL, FSMSC, and PPTL, share a comparable linear time complexity $\mathcal{O}(n)$. However, these models typically require more runtime than our GProM, particularly on large-scale datasets, highlighting our efficiency.

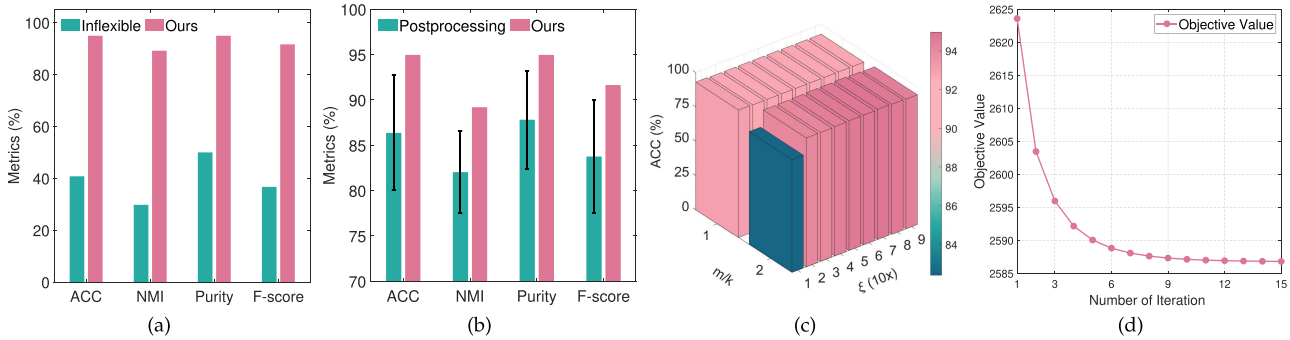


Fig. 7. Results on Dermatology dataset. (a) Ablation study for prototype selection (Inflexible versus Ours). (b) Ablation study for label inference (Postprocessing versus Ours). (c) Parameter sensitivity w.r.t. m and ξ. (d) Convergence.

4) For GProM, the four options exhibit comparable runtime. While our model requires more execution time than BMVC, FMCNOF, SDAFG, and FastMICE due to our complex ADMM optimization, we believe the additional computation is worthwhile for competitive performance.

For reference, the absolute runtime results are provided in supplementary materials (Section 13).

H. Ablation Study

Fig. 7(a) compares prototype selection strategies: Inflexible versus Ours. "Inflexible" refers to the use of k-means sampling to generate prototypes, which is a popular method. "Ours" denotes a learnable prototype selection strategy that adaptively optimizes prototypes.

The results indicate that our flexible prototype learning strategy significantly outperforms the "Inflexible" method, verifying the importance of updating prototypes. The inferior performance of "Inflexible" prototypes can be attributed to their lack of updates to the pre-generated prototypes from k-means. Conversely, the learnable prototype method adaptively updates the prototype distribution, facilitating the exploration of inherent topological structures. Additional results are provided in supplementary materials (Section 10.1).

Fig. 7(b) compares label inference strategies: Postprocessing versus Ours. "Postprocessing" refers to a two-stage manner that first generates spectral embedding and then performs clustering partition to output labels, while "Ours" represents a one-stage discrete label inference strategy.

We observe that the "Postprocessing" demonstrates significant instability, as evidenced by notable performance variance, whereas "Ours" achieves zero variance and superior performance metrics. These results validate our stability and effectiveness. Further details are provided in supplementary materials (Section 10.2).

I. Parameter Sensitivity

Fig. 7(c) reports experimental results on Dermatology across varying $m \in \{k, 2k, 3k, 4k\}$ and $\xi \in \{0.1, 0.2, \ldots, 0.9\}$. For clarity, parameter ξ is scaled by a factor of 10 (10x). More results are provided in supplementary materials (Section 11.1). To satisfy the constraint $m \leq \min\{d_l\}_{l=1}^{v}$, the parameter m is adaptively set to $m \in \{k, 2k\}$ for Dermatology, VGGF2_200, T-ImageNet, while it is fixed at $m = k$ for Caltech256. We observe dataset-related parameter sensitivity, with slight performance fluctuations concerning m and ξ on Dermatology. While, for the other datasets, optimal performance is generally achieved with smaller m and ξ, typically at $m = 1k$ and $\xi = 0.1$.

In addition, a sensitivity analysis with respect to the ADMM scaling parameter ρ is provided in supplementary materials (Section 11.2).

J. Convergence

As discussed in Section IV-A2, the ALM terms in (8) gradually approach zero as the parameters λ and γ are iteratively increased. Consequently, the ALM objective converges asymptotically to the original objective, with the gap between them approaching 0. From Fig. 7(d), the objective value decreases monotonically and stabilizes at a converged value. Further results are provided in supplementary materials (Section 11).

VI. Conclusion

This paper pioneers introducing probabilistic graphical models for modeling multi-view bipartite graph clustering, providing a novel perspective on the probabilistic relationships among variables. We demonstrate that multi-view data can be explicitly decomposed into view-related and view-shared components, each with level-specific noise, within the framework of a MLE problem. Using Gaussian and rotational invariant Laplacian distributions as examples, we instantiate the likelihood function and prove that minimizing feature- and structure-level noise actually approximates the lower bound of the MLE for data observations. We further extend the MLE setting by incorporating tailored constraints specifically designed for clustering tasks, providing insightful suggestions for MVBGC modeling. Future work can explore more distribution assumptions to develop data-driven probabilistic graphical modeling for MVBGC tasks.

References

[1] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[2] T. W. Tsai, C. Li, and J. Zhu, "MiCE: Mixture of contrastive experts for unsupervised image clustering," in *Proc. 9th Int. Conf. Learn. Representations*, Austria, 2021, pp. 1–13.

[3] S. Gao, Z. Li, M. Yang, M. Cheng, J. Han, and P. H. S. Torr, "Large-scale unsupervised semantic segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 6, pp. 7457–7476, Jun. 2023.

[4] S. Liu, L. Zhang, X. Yang, H. Su, and J. Zhu, "Unsupervised part segmentation through disentangling appearance and shape," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 8355–8364.

[5] L. Bai, J. Liang, and Y. Zhao, "Self-constrained spectral clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 4, pp. 5126–5138, Apr. 2023.

[6] X. Li, H. Zhang, and R. Zhang, "Adaptive graph auto-encoder for general data clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 12, pp. 9725–9732, Dec. 2022.

[7] Y. Liu et al., "Dink-net: Neural clustering on large graphs," in *Proc. 40th Int. Conf. Mach. Learn.*, Honolulu, Hawaii, USA, 2023, pp. 21794–21812.

[8] N. Zhang and S. Sun, "Multiview unsupervised shapelet learning for multivariate time series clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 4, pp. 4981–4996, Apr. 2023.

[9] Q. Wang, Z. Tao, W. Xia, Q. Gao, X. Cao, and L. Jiao, "Adversarial multiview clustering networks with adaptive fusion," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 10, pp. 7635–7647, Oct. 2023.

[10] C. Liu, G. Xu, J. Wen, Y. Liu, C. Huang, and Y. Xu, "Partial multi-view multi-label classification via semantic invariance learning and prototype modeling," in *Proc. 41th Int. Conf. Mach. Learn.*, Vienna, Austria, 2024, pp. 32253–32267.

[11] W. Zhang, L. Jiao, F. Liu, S. Yang, and J. Liu, "Adaptive contourlet fusion clustering for SAR image change detection," *IEEE Trans. Image Process.*, vol. 31, pp. 2295–2308, 2022.

[12] L. Zhu, A. Galstyan, J. Cheng, and K. Lerman, "Tripartite graph clustering for dynamic sentiment analysis on social media," in *Proc. ACM SIGMOD Int. Conf. Knowl. Manag. Data.*, Snowbird, UT, USA, 2014, pp. 1531–1542.

[13] Y. Liu et al., "End-to-end learnable clustering for intent learning in recommendation," in *Proc. 38th Adv. Neural Inf. Process. Syst.*, Vancouver, BC, Canada, 2024, pp. 5913–5949.

[14] C. Zhang et al., "Generalized latent multi-view subspace clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 1, pp. 86–99, Jan. 2020.

[15] Z. Huang, P. Hu, J. T. Zhou, J. Lv, and X. Peng, "Partially view-aligned clustering," in *Proc. 33th Adv. Neural Inf. Process. Syst.*, 2020, pp. 2892–2902.

[16] L. Li et al., "Local sample-weighted multiple kernel clustering with consensus discriminative graph," *IEEE Trans. Neural Netw.*, vol. 35, no. 2, pp. 1721–1734, Feb. 2024.

[17] X. Li, H. Zhang, R. Wang, and F. Nie, "Multiview clustering: A scalable and parameter-free bipartite graph fusion method," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 1, pp. 330–344, Jan. 2022.

[18] Z. Kang, W. Zhou, Z. Zhao, J. Shao, M. Han, and Z. Xu, "Large-scale multi-view subspace clustering in linear time," in *Proc. 34th Conf. Artif. Intell.*, New York, NY, USA, 2020, pp. 4412–4419.

[19] H. Zhang, J. Shi, R. Zhang, and X. Li, "Non-graph data clustering via $\mathcal{O}(n)$ bipartite graph convolution," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 7, pp. 8729–8742, Jul. 2023.

[20] H. Zhang, F. Nie, and X. Li, "Large-scale clustering with structured optimal bipartite graph," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 8, pp. 9950–9963, Aug. 2023.

[21] X. Xue, W. Zhao, Q. Gao, M. Yang, and C. Deng, "Image clustering with transition probabilities learning," *IEEE Trans. Image Process.*, vol. 34, pp. 1441–1453, 2025.

[22] S. Wang et al., "Align then fusion: Generalized large-scale multi-view clustering with anchor matching correspondences," in *Proc. 35th Adv. Neural Inf. Process. Syst.*, New Orleans, LA, USA, 2022, pp. 5882–5895.

[23] F. Nie, J. Xue, W. Yu, and X. Li, "Fast clustering with anchor guidance," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 4, pp. 1898–1912, Apr. 2024.

[24] L. Li et al., "BGAE: Auto-encoding multi-view bipartite graph clustering," *IEEE Trans. Knowl. Data Eng.*, vol. 36, no. 8, pp. 3682–3696, Aug. 2024.

[25] S. Liu et al., "Learn from view correlation: An anchor enhancement strategy for multi-view clustering," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Seattle, WA, USA, 2024, pp. 26151–26161.

[26] E. Pan and Z. Kang, "Multi-view contrastive graph clustering," in *Proc. 35th Adv. Neural Inf. Process. Syst.*, 2021, pp. 2148–2159.

[27] W. Xia, Q. Gao, Q. Wang, X. Gao, C. Ding, and D. Tao, "Tensorized bipartite graph learning for multi-view clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 4, pp. 5187–5202, Apr. 2023.

[28] X. Lu and S. Feng, "Structure diversity-induced anchor graph fusion for multi-view clustering," *ACM Trans. Knowl. Discov. Data*, vol. 17, no. 2, pp. 17:1–17:18, 2023.

[29] D. Koller and N. Friedman, *Probabilistic Graphical Models: Principles and Techniques*. Cambridge, MA, USA: MIT Press, 2009.

[30] C. H. Q. Ding, D. Zhou, X. He, and H. Zha, "R_1-PCA: Rotational invariant ℓ_1-norm principal component analysis for robust subspace factorization," in *Proc. 23th Int. Conf. Mach. Learn.*, Pittsburgh, PA, USA, 2006, pp. 281–288.

[31] C. C. Fowlkes, S. J. Belongie, F. R. K. Chung, and J. Malik, "Spectral grouping using the nyström method," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 2, pp. 214–225, Feb. 2004.

[32] Y. Li, F. Nie, H. Huang, and J. Huang, "Large-scale multi-view spectral clustering via bipartite graph," in *Proc. 39th AAAI Conf. Artif. Intell.*, Austin, Texas, USA, 2015, pp. 2750–2756.

[33] K. Gatmiry, M. Aliakbarpour, and S. Jegelka, "Testing determinantal point processes," in *Proc. 34th Adv. Neural Inf. Process. Syst.*, 2020, pp. 12779–12791.

[34] D. Huang, C. Wang, J. Wu, J. Lai, and C. Kwoh, "Ultra-scalable spectral clustering and ensemble clustering," *IEEE Trans. Knowl. Data Eng.*, vol. 32, no. 6, pp. 1212–1226, Jun. 2020.

[35] D. Huang, C. Wang, and J. Lai, "Fast multi-view clustering via ensembles: Towards scalability, superiority, and simplicity," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 11, pp. 11388–11402, Nov. 2023.

[36] W. Zhu, F. Nie, and X. Li, "Fast spectral clustering with efficient large graph construction," in *Proc. 42th IEEE Int. Conf. Acoust. Speech Signal Process.*, New Orleans, LA, USA, 2017, pp. 2492–2496.

[37] M. Sun et al., "Scalable multi-view subspace clustering with unified anchors," in *Proc. 29th ACM Int. Conf. Multimedia*, China, 2021, pp. 3528–3536.

[38] S. Wang et al., "Fast parameter-free multi-view subspace clustering with consensus anchor guidance," *IEEE Trans. Image Process.*, vol. 31, pp. 556–568, 2022.

[39] L. Li et al., "JetBGC: Joint robust embedding and structural fusion bipartite graph clustering," *IEEE Trans. Knowl. Data Eng.*, vol. 37, no. 9, pp. 5346–5359, Sep. 2025.

[40] E. Elhamifar and R. Vidal, "Sparse subspace clustering: Algorithm, theory, and applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 11, pp. 2765–2781, Nov. 2013.

[41] L. K. Saul and S. T. Roweis, "Think globally, fit locally: Unsupervised learning of low dimensional manifolds," *J. Mach. Learn. Res.*, vol. 4, pp. 119–155, 2003.

[42] L. Li, J. Zhang, S. Wang, X. Liu, K. Li, and K. Li, "Multi-view bipartite graph clustering with coupled noisy feature filter," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 12, pp. 12842–12854, Dec. 2023.

[43] Z. Chen, X. Wu, T. Xu, and J. Kittler, "Fast self-guided multi-view subspace clustering," *IEEE Trans. Image Process.*, vol. 32, pp. 6514–6525, 2023.

[44] Q. Gao, F. Li, Q. Wang, X. Gao, and D. Tao, "Manifold based multi-view k-means," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 47, no. 4, pp. 3175–3182, Apr. 2025.

[45] N. Zhang, X. Zhang, and S. Sun, "Efficient multiview representation learning with correntropy and anchor graph," *IEEE Trans. Knowl. Data Eng.*, vol. 36, no. 9, pp. 4632–4645, Sep. 2024.

[46] S. Yu et al., "Sparse low-rank multi-view subspace clustering with consensus anchors and unified bipartite graph," *IEEE Trans. Neural Netw.*, vol. 36, no. 1, pp. 1438–1452, Jan. 2025.

[47] P. Zhang et al., "Let the data choose: Flexible and diverse anchor graph fusion for scalable multi-view clustering," in *Proc. 37th AAAI Conf. Artif. Intell.*, Washington DC, USA, 2023, pp. 11262–11269.

[48] W. Xia, T. Wang, Q. Gao, M. Yang, and X. Gao, "Graph embedding contrastive multi-modal representation learning for clustering," *IEEE Trans. Image Process.*, vol. 32, pp. 1170–1183, 2023.

[49] J. Ji and S. Feng, "Anchor structure regularization induced multi-view subspace clustering via enhanced tensor rank minimization," in *Proc. 19th IEEE Int. Conf. Comput. Vis.*, Paris, France, 2023, pp. 19286–19295.

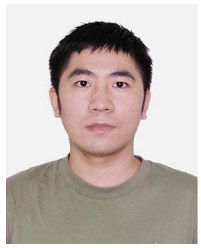
[50] H. Yang, Q. Gao, W. Xia, M. Yang, and X. Gao, "Multiview spectral clustering with bipartite graph," *IEEE Trans. Image Process.*, vol. 31, pp. 3591–3605, 2022.

[51] C. Liu et al., "Masked two-channel decoupling framework for incomplete multi-view weak multi-label learning," in *Proc. 37th Adv. Neural Inf. Process. Syst.*, New Orleans, LA, USA, 2023, pp. 32387–32400.

[52] S. Wang et al., "Highly-efficient incomplete large-scale multi-view clustering with consensus bipartite graph," in *Proc. 35th IEEE Conf. Comput. Vis. Pattern Recognit.*, New Orleans, LA, USA, 2022, pp. 9776–9785.

[53] C. Liu, J. Wen, Y. Xu, B. Zhang, L. Nie, and M. Zhang, "Reliable representation learning for incomplete multi-view missing multi-label classification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 47, no. 6, pp. 4940–4956, Jun. 2025.
[54] H. Zhang, J. Shi, R. Zhang, and X. Li, "Non-graph data clustering via O(N) bipartite graph convolution," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 7, pp. 8729–8742, Jul. 2023.
[55] J. A. Bilmes et al., "A gentle tutorial of the em algorithm and its application to parameter estimation for Gaussian mixture and hidden Markov models," *Int. Comput. Sci. Inst.*, vol. 4, no. 510, pp. 1–248, 1998.
[56] S. P. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, 2011.
[57] K. Fan, "On a theorem of weyl concerning eigenvalues of linear transformations I," *Proc. Nat. Acad. Sci. USA*, vol. 35, no. 11, pp. 652–655, 1949.
[58] P. Tseng, "Convergence of a block coordinate descent method for nondifferentiable minimization," *J. Optim. Theory. Appl.*, vol. 109, no. 3, pp. 475–494, 2001.
[59] J. Huang, F. Nie, H. Huang, and C. Ding, "Robust manifold nonnegative matrix factorization," *ACM Trans. Knowl. Discov. Data.*, vol. 8, no. 3, pp. 1–21, 2014.
[60] D. P. Bertsekas, *Constrained Optimization and Lagrange Multiplier Methods*. Cambridge, MA, USA: Academic Press, 2014.
[61] Z. Lin, M. Chen, and Y. Ma, "The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices," *J. Struct. Biol.*, vol. 181, no. 2, pp. 116–27, 2010.
[62] J. Yurkiewicz, "Constrained optimization and lagrange multiplier methods," *Networks*, vol. 15, no. 1, pp. 138–140, 1985.
[63] D. Kong, C. H. Q. Ding, and H. Huang, "Robust nonnegative matrix factorization using $\ell_{2,1}$-norm," in *Proc. 20th ACM Int. Conf. Inf. Knowl. Manag.*, Glasgow, United Kingdom, 2011, pp. 673–682.
[64] C.-L. Wang, F. Nie, R. Wang, and X. Li, "Revisiting fast spectral clustering with anchor graph," in *Proc. 45th IEEE Int. Conf. Acoust. Speech Signal Process.*, Barcelona, Spain, 2020, pp. 3902–3906.
[65] X. Cai, F. Nie, and H. Huang, "Multi-view k-means clustering on Big Data," in *Proc. 23th Int. Joint Conf. Artif. Intell.*, Beijing, China, 2013, pp. 2598–2604.
[66] F. Nie, J. Li, and X. Li, "Parameter-free auto-weighted multiple graph learning: A framework for multiview clustering and semi-supervised classification," in *Proc. 25th Int. Joint Conf. Artif. Intell.*, New York, NY, USA, 2016, pp. 1881–1887.
[67] R. Li, C. Zhang, Q. Hu, P. Zhu, and Z. Wang, "Flexible multi-view representation learning for subspace clustering," in *Proc. 28th Int. Joint Conf. Artif. Intell.*, Macao, China, 2019, pp. 2916–2922.
[68] Z. Kang et al., "Partition level multiview subspace clustering," *Neural Netw.*, vol. 122, pp. 279–288, 2020.
[69] Z. Zhang, L. Liu, F. Shen, H. T. Shen, and L. Shao, "Binary multi-view clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 7, pp. 1774–1782, Jul. 2019.
[70] B. Yang, X. Zhang, F. Nie, F. Wang, W. Yu, and R. Wang, "Fast multi-view clustering via nonnegative and orthogonal factorization," *IEEE Trans. Image Process.*, vol. 30, pp. 2575–2586, 2020.
[71] S. Fang, D. Huang, X. Cai, C. Wang, C. He, and Y. Tang, "Efficient multi-view clustering via unified and discrete bipartite graph learning," *IEEE Trans. Neural Netw.*, vol. 35, no. 8, pp. 11436–11447, Aug. 2024.
[72] Q. Shen, Y. Chen, C. Zhang, Y. Tian, and Y. Liang, "Pick-and-place transform learning for fast multi-view clustering," *IEEE Trans. Image Process.*, vol. 33, pp. 1272–1284, 2024.
[73] J. Xu et al., "Investigating and mitigating the side effects of noisy views for self-supervised clustering algorithms in practical multi-view scenarios," in *Proc. 37th IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Seattle, WA, USA, 2024, pp. 22957–22966.
[74] J. Xu, H. Tang, Y. Ren, L. Peng, X. Zhu, and L. He, "Multi-level feature learning for contrastive multi-view clustering," in *Proc. 35th IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, New Orleans, LA, USA, 2022, pp. 16030–16039.
[75] H. Tang and Y. Liu, "Deep safe multi-view clustering: Reducing the risk of clustering performance degradation caused by view increase," in *Proc. 35th IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, New Orleans, LA, USA, 2022, pp. 202–211.
[76] Y. Liu et al., "Improved dual correlation reduction network with affinity recovery," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 36, no. 4, pp. 6159–6173, Apr. 2025.
[77] C. Liu, Z. Wu, J. Wen, Y. Xu, and C. Huang, "Localized sparse incomplete multi-view clustering," *IEEE Trans. Multimedia*, vol. 25, pp. 5539–5551, 2023.
[78] Y. Liu, S. Zhu, T. Yang, J. Ma, and W. Zhong, "Identify then recommend: Towards unsupervised group recommendation," in *Proc. 38th Adv. Neural Inf. Process. Syst.*, Vancouver, BC, Canada, 2024, pp. 96101–96126.
[79] J. Zhang et al., "TFMKC: Tuning-free multiple kernel clustering coupled with diverse partition fusion," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 36, no. 5, pp. 9592–9605, May 2025.
[80] Y. Liu et al., "Simple contrastive graph clustering," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 35, no. 10, pp. 13789–13800, Oct. 2024.
[81] C. Liu, J. Wen, Z. Wu, X. Luo, C. Huang, and Y. Xu, "Information recovery-driven deep incomplete multiview clustering network," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 35, no. 11, pp. 15442–15452, Nov. 2024.
[82] J. Zhang et al., "Multiple kernel clustering with dual noise minimization," in *Proc. 30th ACM Int. Conf. Multimedia*, New York, NY, USA, 2022, pp. 3440–3450.
[83] Y. Liu et al., "Hard sample aware network for contrastive deep graph clustering," in *Proc. 37th AAAI Conf. Artif. Intell.*, Washington, DC, USA, 2023, pp. 8914–8922.
[84] C. Liu, J. Wen, X. Luo, and Y. Xu, "Incomplete multi-view multi-label learning via label-guided masked view- and category-aware transformers," in *Proc. 37th AAAI Conf. Artif. Intell.*, Washington, DC, USA, 2023, pp. 8816–8824.
[85] M. Friedman, "A comparison of alternative tests of significance for the problem of m rankings," *Ann. Math. Stat.*, vol. 11, no. 1, pp. 86–92, 1940.

Liang Li (Graduate Student Member, IEEE) received the BS degree from the Huazhong University of Science and Technology (HUST), Wuhan, China, in 2018, and the MS and PhD degrees in computer science from the National University of Defense Technology (NUDT), Changsha, China, in 2020 and 2025, respectively. He is a visiting PhD student with A∗STAR Centre for Frontier AI Research, Singapore from 2023 to 2025. He is currently a researcher with Alibaba Group. His research interests include graph learning and AI4Science.



Yuangang Pan received the PhD degree in computer science from the University of Technology Sydney (UTS), Ultimo, NSW, Australia, in 2020. He is working as a research scientist with the A∗STAR Centre for Frontier AI Research, Singapore. He has authored or coauthored articles in various top journals, such as *Journal of Machine Learning Research*, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *IEEE Transactions on Neural Networks and Learning Systems*, *IEEE Transactions on Knowledge and Data Engineering*, and *ACM Transactions on Information Systems*. His research interests include deep clustering, deep generative learning, and robust ranking aggregation.

Yinghua Yao received the PhD degree in computer science from the University of Technology Sydney (UTS), Ultimo, NSW, Australia, in 2023. She is working as a research scientist with the A∗STAR Centre for Frontier AI Research, Singapore.

Junpu Zhang received the BS degree from the Ocean University of China, Qingdao, China, in 2020. He is currently working toward the PhD degree with the National University of Defense Technology, Changsha, China. His current research interests include kernel learning, ensemble learning, and multi-view clustering.

Moyun Liu received the BS degree from the Hubei University of Technology, Wuhan, China, in 2019, and the MS and PhD degrees from the Huazhong University of Science and Technology (HUST), Wuhan, China, in 2021 and 2025, respectively. He was a visiting PhD student wuth the A∗STAR Centre for Frontier AI Research, Singapore. He is currently a postdoctoral researcher with HUST. His research interests include embodied AI and multi-modal perception.

Xueling Zhu received the BS and MS degrees from Northwestern Polytechnical University and National University of Defense Technology, China, in 2002 and 2004, respectively, and the PhD degree from Central South University, Changsha, China, in 2011. She is currently a professor with Central South University. Her research interests on interdisciplinary research of computer and medicine, especially psychiatric disorder brain imaging.

Xinwang Liu (Senior Member, IEEE) received the PhD degree from the National University of Defense Technology (NUDT), Changsha, China, in 2013. He is currently a full professor with the School of Computer, NUDT. His current research interests include kernel learning and unsupervised feature learning. He has published more than 100 peer-reviewed papers, such as *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *IEEE Transactions on Knowledge and Data Engineering*, *IEEE Transactions on Image Processing*, *IEEE Transactions on Neural Networks and Learning Systems*, *IEEE Transactions on Multimedia*, *IEEE Transactions on Information Forensics and Security*, ICML, NeurIPS, ICCV, CVPR, AAAI, and IJCAI. He serves as an associated editor of the *IEEE Transactions on Neural Networks and Learning Systems* and *IEEE Transactions on Cybernetics*.

Kenli Li (Senior Member, IEEE) received the PhD degree from the Huazhong University of Science and Technology (HUST), China, in 2003. He is currently a full professor of computer science and technology with Hunan University and director of the National Supercomputing Center in Changsha. His research interests include parallel and distributed computing, high-performance computing, AI, cloud computing, and Big Data. He has published more than 600 research papers, such as *IEEE Transactions on Computers*, *IEEE Transactions on Parallel and Distributed Systems*, *IEEE Transactions on Cloud Computing*, *IEEE Transactions on Knowledge and Data Engineering*, DAC, AAAI, ICPP, etc. He serves as the editorial board of *IEEE Transactions on Computers*.

Ivor W. Tsang (Fellow, IEEE) is the director of A∗STAR Centre for Frontier AI Research, Singapore. He is a professor of artificial intelligence with the University of Technology Sydney (UTS), Australia, and the research director of the Australian Artificial Intelligence Institute (AAII). His research interests include transfer learning, deep generative models, learning with weakly supervision, Big Data analytics for data with extremely high dimensions in features, samples and labels. He was the recipient of the ARC Future Fellowship for his outstanding research on Big Data analytics and large-scale machine learning, in 2013. In 2019, his JMLR article toward ultrahigh dimensional feature selection for Big Data was the recipient of the International Consortium of Chinese Mathematicians Best Paper Award. In 2020, he was recognized as the AI 2000 AAAI/IJCAI Most Influential Scholar in Australia for his outstanding contributions to the field between 2009 and 2019. He serves as the editorial board for *Journal of Machine Learning Research*, *Machine Learning*, *Journal of Artificial Intelligence Research*, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *IEEE Transactions on Artificial Intelligence*, *IEEE Transactions on Big Data*, and *IEEE Transactions on Emerging Topics in Computational Intelligence*. He serves/served as an AC or Senior AC for NeurIPS, ICML, AAAI, and IJCAI, and the steering committee of ACML.

Keqin Li (Fellow, IEEE) received the BS degree in computer science from Tsinghua University, in 1985, and the PhD degree in computer science from the University of Houston, in 1990. He is a SUNY distinguished professor with the State University of New York and a National distinguished professor with Hunan University, China. He has authored or co-authored more than 1080 journal articles, book chapters, and refereed conference papers. He holds more than 75 patents announced or authorized by the Chinese National Intellectual Property Administration. Since 2020, he has been among the world's top few most influential scientists in parallel and distributed computing regarding single-year impact (ranked #2) and career-long impact (ranked #4) based on a composite indicator of the Scopus citation database. He is listed in Scilit Top Cited Scholars (2023–2024). He was a 2017 recipient of the Albert Nelson Marquis Lifetime Achievement Award for being listed in Marquis Who's Who in Science and Engineering, Who's Who in America, Who's Who in the World, and Who's Who in American Education for more than twenty consecutive years. He received the Distinguished Alumnus Award from the Computer Science Department, University of Houston, in 2018. He received the IEEE TCCLD Research Impact Award from the IEEE CS Technical Committee on Cloud Computing, in 2022 and the IEEE TCSVC Research Innovation Award from the IEEE CS Technical Community on Services Computing, in 2023. He won the IEEE Region 1 Technological Innovation Award (Academic), in 2023. He was a recipient of the 2022–2023 International Science and Technology Cooperation Award and the 2023 Xiaoxiang Friendship Award of Hunan Province, China. He is a member of the SUNY Distinguished Academy. He is an AAAS fellow, an AAIA fellow, an ACIS fellow, and an AIIA fellow. He is a member of the European Academy of Sciences and Arts. He is a member of Academia Europaea (Academician of the Academy of Europe).