Scalable Federated One-Step Multi-View Clustering with Tensorized Regularization

Wei Feng[1], Danting Liu[1], Qianqian Wang[2*], Wenqi Liang[3], Zheng Yan[4]
[1]School of Computer Science and Technology, Xi'an Jiaotong University
[2]School of Telecommunications Engineering, Xidian University
[3]University of the Chinese Academy of Sciences
[4]School of Cyber Engineering, Xidian University
weifeng.ft@xjtu.edu.cn, 2196123604@stu.xjtu.edu.cn,liangwenqi0123@gmail.com,
qqwang@xidian.edu.cn,zyan@xidian.edu.cn

Abstract

Multi-view clustering (MVC) methods have garnered considerable attention within centralized data frameworks. However, real-world multi-view data are often collected and stored by different organizations, complicating the practical deployment of MVC and motivating the emergence of federated multi-view clustering (FMVC). Existing FMVC approaches typically necessitate post-processing to derive clustering labels and confront challenges in effectively exploring the complementary and consistent information across multi-view data residing in different entities. To address these limitations, we propose a novel framework termed **S**calable **F**ederated **O**ne-Step **M**ulti-**V**iew **C**lustering with **T**ensorized **R**egularization (SFOMVC-TR). This framework facilitates one-step clustering at each client and employs tensor learning to capture consistent and complementary information through a centralized server. Additionally, it adopts anchor graphs to enhance clustering efficiency and scalability in high-dimensional data. By incorporating a $L_{p,q}$ sparse regularization on the projection matrix, SFOMVC-TR enables the direct projection of anchors into clustering assignments to mitigate redundancy. A federated optimization framework is developed to support collaborative and privacy-preserving training under the coordination of the server. Extensive experiments on multiple datasets validate the privacy and effectiveness of our method.

Introduction

Multi-view data offers a comprehensive description by providing insights from various perspectives. For instance, multiple cameras capturing the same scene from different angles can generate diverse viewpoints, thereby enhancing the overall monitoring and analysis of events. Considering the substantial costs of complex multi-view data label acquisition, multi-view clustering (MVC) has emerged as a popular approach for multi-view data processing. Current MVC methods can be categorized into traditional MVC and deep MVC approaches (Xu et al. 2021; Liu et al. 2023), both of which have demonstrated impressive performance. Notably, traditional MVC methods, including multi-view subspace clustering (Cai et al. 2023), multi-view graph clustering (Yu et al. 2024), multi-kernel learning (Tzortzis and Likas 2012), and non-negative matrix factorization (NMF) based techniques (Liu et al. 2013), still receives considerable attention due to their simplicity and interoperability.

In real-world applications, multi-view data may originate from various entities that prefer data confidentiality due to privacy concerns (Chen et al. 2023). This data isolation and privacy requirement present challenges for collaborative training of MVC models in distributed environments. Federated learning, encompassing horizontal federated learning (Hammoud et al. 2022) and vertical federated learning (Liu et al. 2024), becomes an effective solution as a distributed learning paradigm. However, neither approach is directly applicable to multi-view data, because they generally neglect the requirement in MVC to utilize the consistent and complementary information of multiple views. In particular, vertical federated learning methods (Zhu et al. 2023; Huang et al. 2022a) tend to focus more on dataset construction and communication efficiency, while giving insufficient attention to the extraction of complementary features.

The potential of federated learning in distributed environments has inspired the development of federated multi-view clustering (FMVC)(Huang et al. 2022b). In contrast to traditional federated learning, FMVC places greater emphasis on the extraction and fusion of consistent and complementary features. Huang *et al.*(2022b) introduced an efficient FMVC framework based on non-negative matrix factorization (NMF), which directly applies NMF to raw data and hence may encounter efficiency issues when processing high-dimensional data; Hu *et al.*(2023) proposed a federated multi-view fuzzy K-means method; Feng *et al.* developed a centerless NMF-based FMVC approach(Feng et al. 2024b). Nonetheless, these methods typically conduct feature extraction and clustering as separate processes and necessitate post-processing techniques like K-means to obtain clustering labels. This separation may produce redundancy in features and negatively impact clustering performance.

To address these limitations, we propose a novel framework termed **S**calable **F**ederated **O**ne-Step **M**ulti-**V**iew **C**lustering with **T**ensor **R**egularization (SFOMVC-TR). Our method leverages anchor graphs instead of raw data for federated multi-view clustering (FMVC), significantly reducing computational complexity and enhancing model scalability. SFOMVC-TR enables each client to perform one-step clus-

*Corresponding Author

tering on local data through $L_{p,q}$ sparse regularization, facilitating a direct projection from the anchor graph to clustering results. Besides, the server minimizes inconsistency among the clustering assignment matrices of each client by employing tensor learning. Additionally, we develop a privacy-preserving federated optimization framework to support collaborative training of the FMVC model. Our main contributions can be summarized as follows:

- We present SFOMVC-TR, a novel FMVC method, that utilizes anchor graphs for improved scalability. Besides, we introduce $L_{p,q}$ regularization to remove feature redundancy and realize one-step clustering.
- We develop a federated optimization framework with adaptive weights, in which the server employs a tensor-based approach to fuse the clustering results of different clients to guide the local learning process of clients.
- We conduct a security analysis and extensive experiments on multiple datasets to demonstrate the security and availability of our method.

Related Work

Multi-View Clustering

Multi-view clustering (MVC) demonstrates impressive performance due to its ability to describe objects from various perspectives. Current MVC approaches include traditional methods (Li et al. 2015) and deep learning methods (Wang et al. 2023; He et al. 2023). Among them, traditional methods exhibit potential applications due to their efficiency. Recently, anchor graph-based methods have garnered significant attention for their ability to substantially reduce computational complexity (Fang et al. 2023; Yang et al. 2022). For example, MVSC (Li et al. 2015) incorporated anchor graphs into multi-view spectral clustering; Kang *et al.* (2021) developed an anchor-based subspace clustering model that ensures connected components directly reflect clusters by imposing connectivity constraints. Additionally, Wang *et al.* (2021) combined anchor selection with graph construction, enabling mutual enhancement between anchor selection and subsequent clustering for improved performance. However, these methods are designed for centralized data settings and cannot be directly applied in federated scenarios due to data isolation and security issues.

Federated Multi-View Clustering

Federated learning is a distributed machine learning paradigm that facilitates the collaborative training of a unified model. In particular, vertical federated learning (VFL) is well-suited for the Federated Multi-View Clustering (FMVC) scenario, where multi-view data is maintained by different entities. So far, VFL has been extensively studied. For instance, Huang *et al.*(2022a) proposed a vertical K-means method incorporating coreset techniques to enhance efficiency; Zhu *et al.*(2023) introduced a fuzzy clustering method designed for vertically partitioned datasets. Although these vertical federated clustering methods can be readily adapted for MVC applications, they often fail to capture the complementary and consistent information necessary for achieving satisfactory performance on multi-view data. To address this challenge, several FMVC methods have been proposed. For example, Chen *et al.*(2023) developed a deep FMVC method that employs global self-supervision. Despite its effectiveness, the deep model introduces additional communication and computational overhead. In contrast, Huang *et al.*(2022b) constructed a FMVC method utilizing NMF and K-means, achieving improved efficiency. Inspired by this work, several FMVC methods have been developed (Feng et al. 2024a). However, these approaches either necessitate post-processing of the learned features or struggle to retain the complementary multi-view information effectively, resulting in suboptimal performance.

Proposed Method

Problem Statement

Problem Definition: Our work considers a typical FMVC scenario as introduced, where there is a server $\mathcal{S}$ and M clients $\{\mathcal{C}_m\}(1 \leq m \leq M)$, with multi-view data $\mathbf{X} = \{\mathbf{X}_1, \mathbf{X}_2, ..., \mathbf{X}_M\}$ distributed across these clients. Specifically, the m-th view data $\mathbf{X}_m \in \mathbb{R}^{N \times d_m}$ is held by $\mathcal{C}_m$, where N is the sample number and d_m is the feature dimension of the m-th view. The server $\mathcal{S}$ is responsible for assisting $\mathcal{C}_m$ to produce a global clustering result. Our method aims to learn a global cluster assignment from the local multi-view data without compromising privacy.

Security Model and Design Goals: FMVC aims to enable all clients to cooperatively train an MVC model without leaking any personal information. In this work, we assume all the clients and the server are semi-honest, which means that they are curious about the privacy of others but will behave honestly following the predefined protocol. However, they may exploit the data information of others with the information they obtained. Our underlying model adopts an anchor-based method and thus avoids exchanging raw data. However, we consider that the anchor graph is highly related to data features and might be used to infer data information, which should not be obtained by others. Therefore, we have the following goals:

- ***Security Goal:*** Neither of the server and clients could obtain the raw data or anchor graph of others.
- ***Availability Goal:*** The clustering performance of our FMVC model should be comparable to the centralized MVC models.

Objectives of SOFMVC-TR

SFOMVC-TR is composed of Initialization, Local Training, Global Training, and Aggregation. The Initialization process initializes the necessary parameters, during which all the clients $\mathcal{C}_m$ select a consistent group of anchors with the assistance of $\mathcal{S}$ and generate each local anchor graph; Local Training and Global Training respectively train the local models $\{f_m(\cdot)\}(1 \leq m \leq M)$ by $\mathcal{C}_m$ and a global model $f_g(\cdot)$ for consistency insurance. Aggregation is used to produce the final clustering result. In what follows, we introduce the local and global objectives in detail.

Objective for Local Training: For each client $\mathcal{C}_m$, it trains a local clustering model $\mathbf{F}_m \leftarrow f^{(m)}(\hat{\mathbf{F}}_m, \mathbf{G}_m)$ on

their local anchor graph $\mathbf{G}_m \in \mathbb{R}^{N\times\Omega}$, where $\mathbf{F}_m$ represents the learned local clustering matrix, $\hat{\mathbf{F}}_m$ represents guidance information from $\mathcal{S}$ for $\mathcal{C}_m$. We employ a local projection matrix $\mathbf{P}_m \in \mathbb{R}^{\Omega\times K}$ to directly project $\mathbf{G}_m$ into the clustering result, so as to eliminate the redundancy and directly learn the optimal clustering structure. In our work, we follow the methods in (Hu et al. 2017; Zhang, Wei, and Liu 2022) and introduce $L_{p,q}$ regularization, and with $\hat{\mathbf{F}}_m$ as the guidance, we have the following local objective:

$$\begin{aligned} &\min_{\mathbf{G}_m,\mathbf{P}_m} \|\mathbf{G}_m\mathbf{P}_m - \mathbf{F}_m\|_F^2 + \beta\|\mathbf{P}_m\|_{p,q}^q \\ &\text{s.t.} \quad \mathbf{F}_m^T\mathbf{F}_m = \mathbf{I}, \quad \mathbf{F}_m \geq 0 \quad \mathbf{F}_m = f_g^m(\{\mathbf{F}_m\}) \end{aligned} \tag{1}$$

where $\mathbf{F}_m$ is cluster assignment matrix; $\hat{\mathbf{F}}_m \leftarrow f_g^m(\{\mathbf{F}_m\})$ refers to the m-th component of the global model to generate $\hat{\mathbf{F}}_m$ as guidance to instruct $\mathcal{C}_m$, which will be introduced in the following part; $\|\cdot\|_{p,q}^q$ refers to $L_{p,q}$-regularization, which is defined as follows:

$$\|\mathbf{P}\|_{p,q}^q = (\sum_i \|\mathbf{P}_i\|_p^q)^{\frac{1}{q}} = (\sum_i (\sum_j |\mathbf{P}_{ij}|^p)^{\frac{q}{p}})^{\frac{1}{q}} \tag{2}$$

Tensor-Based Consistency Objective for Global Training: With the local clustering results $\{\mathbf{F}_m\}(1 \leq m \leq M)$ output by local models $\{f^{(m)}(\hat{\mathbf{F}}_m, \mathbf{G}_m)\}$, the server $\mathcal{S}$ trains a global fusion model $\hat{\boldsymbol{\mathcal{F}}} \leftarrow f_g(\{\mathbf{F}_m\})$ via tensor learning. Specifically, $\mathcal{S}$ first constructs a tensor $\boldsymbol{\mathcal{F}}$ by stacking all the $\mathbf{F}_m$ (as depicted in Figure 1). To enable the optimization of the multi-view clustering result, we introduce an extra auxiliary variable $\hat{\boldsymbol{\mathcal{F}}}$ to bridge the gap between the client and the server. By enforcing the optimized $\hat{\boldsymbol{\mathcal{F}}}$ to be similar to $\boldsymbol{\mathcal{F}}$ and minimizing its tensor Schatten-p norm, it ensures the consistency of clustering results. Following this line, the global objective to be optimized is as follows:

$$\min_{\hat{\boldsymbol{\mathcal{F}}}} \|\hat{\boldsymbol{\mathcal{F}}}\|_{S_p}^p \text{ s.t. } \hat{\mathbf{F}}_m = f^{(m)}(\hat{\mathbf{F}}_m, \mathbf{G}_m)(1 \leq m \leq M) \tag{3}$$

where $\|\cdot\|_{S_p}^p$ represents the tensor Schatten-p regularization. Then, $f_g^m(\cdot)$ returns $\hat{\mathbf{F}}_m$the guidance for $\mathcal{C}_m$ by taking the m-th frontal slice of $\hat{\boldsymbol{\mathcal{F}}}$.

Federated Optimization

The optimization of local objectives and the global objective requires the interaction between $\mathcal{S}$ and $\mathcal{C}_m$. When adopting the Alternative Iteration Method for optimization, the collaborative optimization of the local and global objectives is equivalent to optimizing the following objective:

$$\begin{aligned} &\min_{\mathbf{P}_m,\mathbf{F}_m} \sum_{m=1}^{M} \alpha_m^v \left(\|\mathbf{G}_m\mathbf{P}_m - \mathbf{F}_m\|_F^2 + \beta\|\mathbf{P}_m\|_{p,q}^q\right) + \gamma\|\hat{\boldsymbol{\mathcal{F}}}\|_{S_p}^p \\ &\text{s.t.} \quad \mathbf{F}_m^T\mathbf{F}_m = \mathbf{I}, \quad \mathbf{F}_m \geq 0 \quad \mathbf{F}_m = \hat{\mathbf{F}}_m \end{aligned} \tag{4}$$

where γ is a hyper-parameter to balance the influence of the global training. To further balance the contribution of each client and the influence of the server, we assign a weight α_m to $\mathcal{C}_m$ and a fusion weight γ to the server. α_m is the weight to balance the contribution of each client, which is set as $\alpha_m = \frac{\sqrt{s_m}}{\sum_{m=1}^{M}\sqrt{s_m}}$; v is used to adjust the influence of α_m, which is set as -1 in our work, indicating that a larger loss leads to smaller contribution; s_m is set as local loss, *i.e.*, $s_m = \|\mathbf{G}_m\mathbf{P}_m - \mathbf{F}_m\|_F^2 + \beta\|\mathbf{P}_m\|_{p,q}^q$. In our federated framework, the influence of $\mathcal{S}$ and $\mathcal{C}_m$ on each other is achieved with the constraint $\hat{\mathbf{F}}_m = f^{(m)}(\hat{\mathbf{F}}_m, \mathbf{G}_m)$ and $\mathbf{F}_m = f_g^m(\{\mathbf{F}_m\})$, which enforces the local cluster result to be similar to be the global fused result. We should note that the final aggregation will not be performed until the model converges, because the adopted tensor Schatten-p regularization helps to enforce the consistency of each client's $\mathbf{F}_m$ and meanwhile well exploits complementary information, thus providing more flexibility.

For another, since multi-view data is distributed across multiple clients, and for the sake of privacy, sensitive data could not be shared, which brings a great challenge for optimization. Since anchor graph $\mathbf{G}_m$ can also reflect certain data information, we consider $\mathbf{G}_m$ to be improper to be transmitted to the server or other clients directly. However, the clustering assignment matrix $\mathbf{F}_m$, based on local data, does not cause data leakage and can be transmitted between the server and the clients. Based on these analyses, we illustrate our federated optimization framework as follows.

System Initialization: In this process, the server and all the clients negotiate the consistent public parameters and cooperatively determine the anchors. Our work adopts a basic anchor selection strategy as illustrated in (Li et al. 2020), whose federated implementation can be illustrated as follows. First, each client computes local feature scores by $\mathbf{sc}_m = \sum_{j=1}^{d_m}\left(\mathbf{X}_{m(:,j)} - \mathbf{x}_{m,j}\mathbf{1}_N\right) \in \mathbb{R}^{N\times 1}$ and sends them to the server, where $\mathbf{x}_{m,j} = \min_i \mathbf{X}_{m(i,j)}$. The server aggregates all the local scores via $\mathbf{sc} = \sum_{m=1}^{M}\mathbf{sc}_m$ and selects anchor indices **Ind**. Since the score aggregation only contains addition operation, partial homomorphic encryption (PHE) can be utilized to prevent $\mathcal{S}$ from learning the original local score $\mathbf{sc}_m$ for better privacy. Then, it broadcasts the **Ind** to all clients, which computes their local anchor graphs $\mathbf{G}_m$ based on the received indices. Notice that it happens only once and PHE is relatively efficient, thus introducing limited communication or computational overhead.

Local Training: Each client $\mathcal{C}_m$ respectively optimizes the local objective in **Eq.** (1) with the assigned weight α_m via the Augmented Lagrange Multiplier (ALM) method. Introducing the auxiliary variables $\mathbf{H}_m$, and $\mathbf{Q}_m$, we rewrite the local objective as:

$$\begin{aligned} \min &\frac{1}{\alpha_m}\{\|\mathbf{G}_m\mathbf{P}_m - \mathbf{F}_m\|_F^2 + \beta\|\mathbf{Q}_m\|_{p,q}^q\} \\ &+ \frac{\rho_1}{2}\|\mathbf{F}_m - \mathbf{H}_m + \frac{\mathbf{M}_m}{\rho_1}\|_F^2 \\ &+ \frac{\rho_2}{2}\|\mathbf{P}_m - \mathbf{Q}_m + \frac{\mathbf{Y}_m}{\rho_2}\|_F^2 \\ &+ \frac{\rho_3}{2}\|\mathbf{F}_m - \hat{\mathbf{F}}_m + \frac{\mathbf{W}_m}{\rho_3}\|_F^2 \\ \text{s.t.} \quad &\mathbf{F}_m^T\mathbf{F}_m = \mathbf{I}, \quad \mathbf{H}_m \geq 0 \end{aligned} \tag{5}$$

Step 1. **Update $\mathbf{P}_m$:**

$$\min_{\mathbf{P}_m} \frac{1}{\alpha_m}\|\mathbf{G}_m\mathbf{P}_m - \mathbf{F}_m\|_F^2 + \frac{\rho_2}{2}\|\mathbf{P}_m - \mathbf{Q}_m + \frac{\mathbf{Y}_m}{\rho_2}\|_F^2 \quad \text{s.t.} \quad \mathbf{F}_m^T\mathbf{F}_m = \mathbf{I} \tag{6}$$

By setting its partial derivative to zero, the optimal value of $\mathbf{P}_m$ is given by:

$$\mathbf{P}_m = \left(\frac{2}{\alpha_m}\mathbf{G}_m^T\mathbf{G}_m + \rho_2\mathbf{I}\right)^{-1} \left(\frac{2}{\alpha_m}\mathbf{G}_m^T\mathbf{F}_m + \rho_2\left(\mathbf{Q}_m - \frac{\mathbf{Y}_m}{\rho_2}\right)\right) \tag{7}$$

Step 2. **Update $\mathbf{Q}_m$:** we focus on the term related to $\mathbf{Q}_m$:

$$\min_{\mathbf{Q}_m} \frac{\beta}{\alpha_m}\|\mathbf{Q}_m\|_{p,q}^q + \frac{\rho_2}{2}\|\mathbf{P}_m - \mathbf{Q}_m + \frac{\mathbf{Y}_m}{\rho_2}\|_F^2 \tag{8}$$

Proposition 1. (Hu et al. 2017)*, the optimization of* ***Eq.***(8) *is equivalent to solving the following M sub-problems:*

$$\min_{\mathbf{Q}_{m,i}} \frac{\beta}{\alpha_m}\|\mathbf{Q}_{m,i}\|_p^q + \frac{\rho_2}{2}\|\mathbf{P}_{m,i} - \mathbf{Q}_{m,i} + \frac{\mathbf{Y}_{m,i}}{\rho_2}\|_F^2 \tag{9}$$

According to (Hu et al. 2017), **Eq.** (9) does not have an analytical solution on all the conditions, and current works provide its analytical solution $q = 0, \frac{1}{2}, \frac{2}{3}, 1$, and $p = 2$. For other general cases, it can be optimized via Newton method (Hu et al. 2017).

Step 3. **Update $\mathbf{H}_m$:**

$$\min_{\mathbf{H}_m} \frac{\rho_1}{2}\|\mathbf{F}_m - \mathbf{H}_m + \frac{\mathbf{M}_m}{\rho_1}\|_F^2 \quad \text{s.t.} \quad \mathbf{H}_m \geq 0 \tag{10}$$

Setting **Eq.**(10) to zero under the constraint of $\mathbf{H}_m$, we obtain:

$$\mathbf{H}_m = \max\left(\mathbf{F}_m + \frac{\mathbf{M}_m}{\rho_1}, 0\right) \tag{11}$$

Step 4. **Update $\mathbf{F}_m$:** The objective function can be written as follows:

$$\begin{aligned}&\min_{\mathbf{F}_m} \frac{1}{\alpha_m}\|\mathbf{G}_m\mathbf{P}_m - \mathbf{F}_m\|_F^2 + \frac{\rho_1}{2}\|\mathbf{F}_m - \ \mathbf{H}_m + \frac{\mathbf{M}_m}{\rho_1}\|_F^2 \\ &\quad + \frac{\rho_3}{2}\|\mathbf{F}_m - \hat{\mathbf{F}}_m + \frac{\mathbf{W}_m}{\rho_3}\|_F^2 \\ &\quad \Leftrightarrow \\ &\max_{\mathbf{F}_m} \quad \mathrm{Tr}(\mathbf{F}_m^T(\frac{2}{\alpha_m}\mathbf{G}_m\mathbf{P}_m + \rho_1(\mathbf{H}_m - \frac{\mathbf{M}_m}{\rho_1}) + \\ &\quad \rho_3(\hat{\mathbf{F}}_m - \frac{\mathbf{W}_m}{\rho_3}))) \\ &\quad \text{s.t.} \quad \mathbf{F}_m^T\mathbf{F}_m = \mathbf{I}\end{aligned} \tag{12}$$

The optimal solution of $\mathbf{F}_m$ can be obtained by $\mathbf{F}_m^* = \mathbf{U}_m\mathbf{V}_m^\top$, where $\mathbf{U}_m$ and $\mathbf{V}_m$ are the left and right singular matrices of $\frac{2}{\alpha_m}\mathbf{G}_m\mathbf{P}_m + \rho_1(\mathbf{H}_m - \frac{\mathbf{M}_m}{\rho_1}) + \rho_3(\hat{\mathbf{F}}_m - \frac{\mathbf{W}_m}{\rho_3})$.

Algorithm 1: SFOMVC-TR

Input: The data $\mathbf{X} = \{\mathbf{X}_1, \mathbf{X}_2, ..., \mathbf{X}_M\}$ in M local clients; the number of cluster K;
Output: Global cluster result **F**

▷ **System Initialization**
Each client $\mathcal{C}_m$ initializes $\mathbf{F}_m$ with one-hot vector, $\mathbf{Y}_m = \mathbf{M}_m = \mathbf{W}_m = 0$, and $\alpha_m = \frac{1}{M}$; $\mathcal{S}$ sets $\boldsymbol{\mathcal{J}} = 0$
Clients select a consistent anchor set with the assistance of the server and produce local anchor graph $\mathbf{G}_m$
while not converged **do**
  for $m = 1$ **to** M **do**
    ▷ **Local Training on m-th client $\mathcal{C}_m$**
    Compute weight α_m with s_m and s
    Update $\mathbf{P}_m$, $\mathbf{Q}_m$, and $\mathbf{H}_m$
    Update $\mathbf{F}_m$
    Update $\sqrt{s_m}$
    Send $\mathbf{F}_m$ and $\sqrt{s_m}$ to $\mathcal{S}$
  end for
  ▷ **Global Training on the Server $\mathcal{S}$**
  Stack all $\mathbf{F}_m$ to construct tensor $\boldsymbol{\mathcal{F}}$
  Update $\hat{\boldsymbol{\mathcal{F}}}$ with $\boldsymbol{\mathcal{F}}$ according to **Eq**.(14)
  Compute $s = \sum_{m=1}^{M}\sqrt{s_m}$
  Send s and $\hat{\mathbf{F}}_m$ to $\mathcal{C}_m$
end while
▷ **Aggregation on the Server $\mathcal{S}$**
$\mathcal{S}$ aggregates $\mathbf{F}_m$ into **F** with $\left[\sum_{m=1}^{M}\frac{\mathbf{F}_m}{\alpha_m}\right]$
return F

Global Training: The only variable to be updated by $\mathcal{S}$ is $\hat{\boldsymbol{\mathcal{F}}}$. By performing ALM, the global objective function w.r.t $\hat{\boldsymbol{\mathcal{F}}}$ becomes:

$$\min_{\hat{\boldsymbol{\mathcal{F}}}} \gamma\|\hat{\boldsymbol{\mathcal{F}}}\|_{Sp}^p + \frac{\rho_4}{2}\|\boldsymbol{\mathcal{F}} - \hat{\boldsymbol{\mathcal{F}}} + \frac{\boldsymbol{\mathcal{J}}}{\rho_4}\|_F^2 \tag{13}$$

Accordingly, the solution is

$$\hat{\boldsymbol{\mathcal{F}}}^* = \mathrm{ifft}\left(\boldsymbol{\mathcal{U}} * \boldsymbol{\mathcal{D}}_{\frac{\gamma}{\rho_4},p}(\boldsymbol{\mathcal{P}}) * \boldsymbol{\mathcal{V}}^T\right) \tag{14}$$

where $\boldsymbol{\mathcal{P}} = \boldsymbol{\mathcal{F}} + \frac{\boldsymbol{\mathcal{J}}}{\rho_4}$, and $\boldsymbol{\mathcal{U}}$ and $\boldsymbol{\mathcal{V}}$ are acquired via t-SVD of $\mathcal{P}$, *i.e.*, $\boldsymbol{\mathcal{P}} = \boldsymbol{\mathcal{U}} * \boldsymbol{\mathcal{S}} * \boldsymbol{\mathcal{V}}^T$.

Workflow

Joint training of these variables requires several rounds of communication between the clients and the server to validate both the local and global training. To better present the procedures of our federated framework, we depict the system workflow in Figure 1 and describe it as follows. As shown in Figure 1, all the clients $\mathcal{C}_m$ and $\mathcal{S}$ initialize the parameters with the predefined values, and then, $\mathcal{S}$ assists the clients to generate a consistent anchor point index **Ind** and a local anchor graph $\mathbf{G}_m$ for $\mathcal{C}_m$.

After that, the client starts the local training and generates $\mathbf{F}_m$, which will be sent to $\mathcal{S}$ for global training. Since the computation of α_m requires the information from others, after local training, $\mathcal{C}_m$ also computes and transmits $\sqrt{s_m}$. With the collected $\mathbf{F}_m$ and $\sqrt{s_m}$, $\mathcal{S}$ performs global training

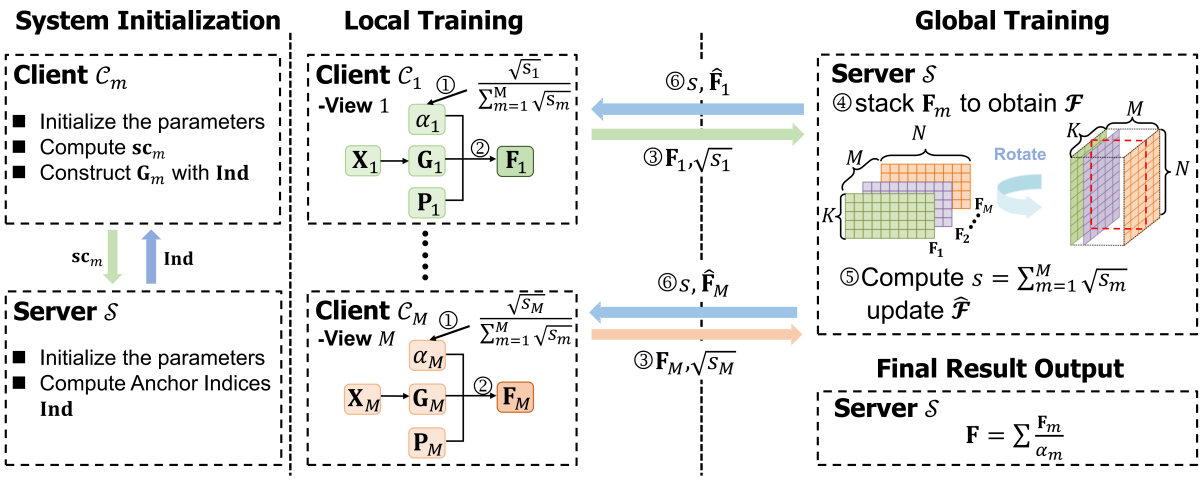

Figure 1: Workflow of SFOMVC-TR

to learn $\hat{\mathcal{F}}$ and computes $s = \sum_{m=1}^{M} \sqrt{s_m}$. Subsequently, it divides $\hat{\mathcal{F}}$ into M components $\hat{\mathbf{F}}_m$, and sends $\hat{\mathbf{F}}_m$ and s to $\mathcal{C}_m$. The local training and global training are iteratively conducted until the model converges.

Finally, $\mathcal{S}$ aggregates all the $\mathbf{F}_m$ into $\mathbf{F} = \sum_{m=1}^{M} \frac{\mathbf{F}_m}{\alpha_m}$. **Alg.**(1) summarizes the workflow.

Privacy Analysis and Performance Evaluation

Privacy Analysis

As mentioned, our primary privacy goal is to avoid data information leakage by protecting raw data $\mathbf{X}_m$ as well as anchor graph $\mathbf{G}_m$. During the federated optimization, we could observe that the parameters to be shared with $\mathcal{S}$ by $\mathcal{C}_m$ are $\mathbf{sc}_m$ in the Initialization, local clustering assignment matrix $\mathbf{F}_m$, and s_m. Among them $\mathbf{sc}_m = \sum_j \left(\mathbf{X}_{m(:,j)} - \mathbf{x}_{m,j}\mathbf{1}_N\right)$, which contains quite limited information of $\mathbf{X}_m$. For the privacy-sensitive applications, this process can be further preserved via PHE, as described in Initialization, which ensures $\mathcal{S}$ can only obtain $\sum_{m=1}^{M} \mathbf{sc}_m$, without revealing the plain $\mathbf{sc}_m$. Therefore, the disclosure of $\mathbf{sm}_m$ will not introduce privacy risk. For $\mathbf{F}_m$ and s_m, they are the clustering assignment and the local loss, which indicates less private information and will not harm the data privacy. To summarize, we consider that the federated optimization framework well protects the security of raw data.

Performance Evaluation

Experimental Settings: We validated our method on eight multi-view datasets and compared it with existing methods to evaluate its availability. Our experiments include a server and multiple clients, each holding a view of the data. Since the objective with $L_{p,q}$ regularization does not provide a unified-from solution for all the cases, our implementation fixes p of $L_{p,q}$ as 2, since in this case there exists an analytical solution for $0 < q \leq 1$

***Datasets*:** We evaluate our method on eight public multi-view datasets: (1)**ORL** (Samaria and Harter 1994); (2)**Yale**; (3)**BBCSport** (Greene and Cunningham 2006); (4)**3-sources**; (5)**WebKB** (Blum and Mitchell 1998); (6) **Uci-Digit**; (7)**NGs** (Hussain, Bisson, and Grimal 2010); (8) **Reuters** (Apté, Damerau, and Weiss 1994).

Compared Methods: We compared our methods with ten clustering methods: (1) **FastMICE** (Huang, Wang, and Lai 2023); (2) **MvLRSSC** (Brbić and Kopriva 2018); (3) **RMSL** (Li et al. 2019); (4) **GMC** (Wang, Yang, and Liu 2019); (5) **FPMVS-CAG** (Wang et al. 2021); (6) **VKMC** (Huang et al. 2022a); (7) **FCVFL** (Zhu et al. 2023); (8) **FedMVL** (Huang et al. 2022b) (9) **FMVC-IMK** (Feng et al. 2024a); (10) **TensorFMVC** (Feng et al. 2024b), where (1)-(5) are centralized multi-view methods, (6)-(7) are vertical federated learning methods, and (8)-(10) are federated MVC methods.

Metrics: We use three metrics, *i.e.*, Accuracy (ACC), Normalized Mutual Information (NMI), and Purity (PUR).

Experimental Results and Analyses

Clustering Performance: Table 1 and Table 2 present the clustering performance of SFOMVC-TR and the comparison methods, from which we have the following observations. Our method outperforms the other three FMVC methods, probably because SFOMVC-TR performs feature selection on the anchor graph to extract effective anchor points and employs tensor-based regularization to explore the complementary information between different views. Even when compared with centralized MVC methods, our method demonstrates comparable performance. Compared to vertical federated clustering, our method is superior and more stable, as it fully utilizes the complementary information from different view embeddings. Notably, on the ORL dataset, SFOMVC-TR achieved 100% in ACC, NMI, and PUR. On the large Reuters dataset, despite many algorithms failing due to memory constraints, our algorithm achieved 72.79% ACC, 64.51% NMI, and 77.22% PUR, indicating its superiority to other methods. This demonstrates the effectiveness and superiority of SFOMVC-TR.

Datasets	ORL			BBCSport			3-sources			Yale		
Metrics	**ACC**	**NMI**	**PUR**	**ACC**	**NMI**	**PUR**	**ACC**	**NMI**	**PUR**	**ACC**	**NMI**	**PUR**
FastMICE	76.25	92.17	81.50	41.91	14.14	45.96	44.38	36.70	55.62	65.46	66.98	66.06
MvLRSSC	86.00	94.48	89.75	76.63	72.36	76.63	80.36	76.78	81.89	58.79	39.20	66.09
RMSL	83.00	93.16	87.75	76.63	72.36	76.63	31.95	14.46	41.42	78.78	78.23	79.38
GMC	21.21	27.51	24.24	80.70	76.00	79.43	69.23	62.16	74.56	54.55	62.44	54.55
FPMVS-CAG	57.75	79.20	61.75	42.10	15.09	51.84	27.82	8.12	38.46	50.31	59.32	51.52
VKMC	31.75	24.63	30.39	38.79	11.42	41.18	54.44	27.56	56.21	53.33	60.78	53.94
FCVFL	71.50	82.86	75.75	47.98	24.06	48.90	65.09	40.08	66.27	36.36	37.30	37.58
FedMVL	51.75	66.84	55.00	78.12	61.84	78.12	70.41	53.53	70.41	49.70	54.12	50.91
FMVC-IMK	93.25	89.66	93.05	90.26	74.95	90.26	78.70	70.50	84.62	78.79	77.90	79.39
TensorFMVC	99.75	99.77	99.75	86.94	68.71	86.94	72.19	66.18	81.07	79.39	78.48	79.39
SFOMVC-TR	**100.00**	**100.00**	**100.00**	**93.38**	**84.98**	**93.38**	**84.02**	**76.20**	**87.57**	**98.18**	**97.73**	**96.18**

Table 1: Clustering performance comparison in terms of ACC(%), NMI(%), and PUR(%) on ORL, BBCSport, 3-sources, and Yale datasets.

Datasets	NGs			WebKB			Uci-digit			Reuters		
Metrics	**ACC**	**NMI**	**PUR**	**ACC**	**NMI**	**PUR**	**ACC**	**NMI**	**PUR**	**ACC**	**NMI**	**PUR**
FastMICE	38.40	26.63	48.00	95.62	66.40	95.63	84.05	85.95	86.25	47.05	33.49	60.97
MvLRSSC	90.26	88.82	91.72	92.58	58.19	92.58	80.36	76.78	81.89	56.58	37.15	56.98
RMSL	96.00	86.11	94.60	60.42	1.93	78.12	51.90	52.05	55.95	OM	OM	OM
GMC	97.80	92.93	97.80	84.02	25.78	84.02	83.90	87.41	86.35	OM	OM	OM
FPMVS-CAG	73.80	59.23	73.80	94.96	69.91	94.96	75.30	75.87	75.35	52.64	32.37	60.32
VKMC	24.00	5.88	24.40	78.02	6.15	78.12	77.80	75.12	80.05	OM	OM	OM
FCVFL	43.60	21.34	44.20	95.91	69.68	95.81	42.40	44.63	45.90	OM	OM	OM
FedMVL	76.60	56.65	76.60	78.12	61.84	78.12	42.50	33.26	44.05	OM	OM	OM
FMVC-IMK	93.20	81.95	93.20	93.72	78.17	97.43	51.80	40.26	52.15	OM	OM	OM
TensorFMVC	80.40	69.40	80.40	72.03	26.48	78.21	95.50	92.72	92.04	34.94	17.84	43.29
SFOMVC-TR	**99.00**	**97.10**	**99.00**	**99.14**	**91.20**	**99.14**	**99.90**	**99.73**	**99.90**	**72.79**	**64.51**	**77.22**

Table 2: Clustering performance comparison in terms of ACC(%), NMI(%), and PUR(%) on NGs, WebKB, Uci-digit, and Reuters datasets.

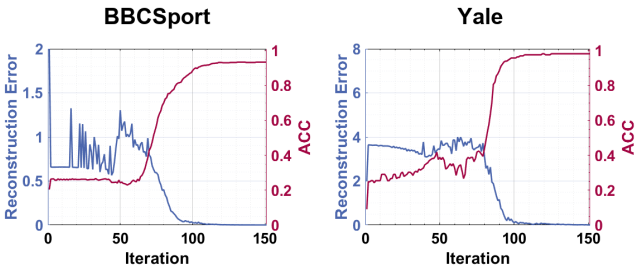

Figure 2: The convergence curves of SFOMVC-TR on BBCSport and Yale datasets

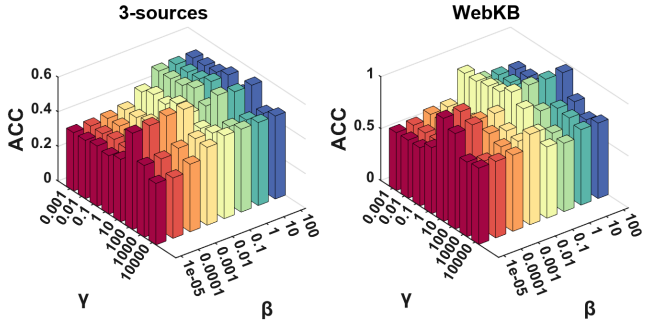

Figure 3: ACC w.r.t. β and γ on 3-sources, WebKB

Convergence Analysis: Figure 2 presents the convergence curves of ACC and the differences between the original variables and the auxiliary variables $\|\mathcal{F} - \hat{\mathcal{F}}\|_\infty$ on the BBCSport and Yale datasets. The results indicate that both ACC and the objective loss converge within a limited number of iterations. The fluctuations observed during the process are due to the global clustering assignments in each iteration, which are obtained by aggregating all local clustering assignments within each cluster.

Parameter Analysis: Our objective function involves two penalty parameters, *i.e.*, β and γ. Figure 3 illustrates the ACC with regard to (w.r.t) β and γ on two datasets. From this, we can observe that: (1) The appropriate β helps to

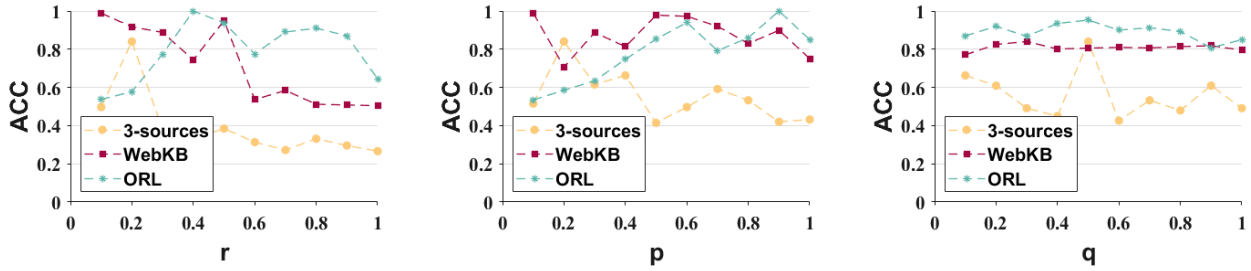

Figure 4: ACC w.r.t. r, p and q on 3-sources, WebKB, and ORL

Dataset	NGs			Yale			3-sources			BBCSport		
Metric	ACC	NMI	PUR	ACC	NMI	PUR	ACC	NMI	PUR	ACC	NMI	PUR
TS-SFOMVC-TR	30.40	6.72	31.00	43.03	45.23	44.24	55.02	26.13	56.80	44.85	11.63	46.69
w.o. S-C & T-R	29.80	4.50	29.80	44.84	49.90	48.48	44.37	23.81	50.29	37.31	5.40	40.07
w.o. S-C	94.60	87.09	94.60	88.48	87.71	88.48	72.78	61.58	73.37	90.07	77.41	90.07
w.o. T-R	36.80	7.52	36.80	56.36	55.33	56.97	60.95	36.27	63.91	45.58	11.12	47.24
SFOMVC-TR	**99.00**	**97.10**	**99.00**	**98.18**	**97.73**	**96.18**	**84.02**	**76.20**	**87.57**	**93.38**	**84.98**	**93.38**

Table 3: Clustering results of the ablation experiments

remove redundant information from the anchor graph, improving clustering performance; (2)A smaller γ reduces accuracy because it hinders learning consistent cluster assignments, but a larger γ also reduces accuracy as it overlooks the heterogeneity of local data.

We also test the performance w.r.t. the three hyperparameters: anchor rate r, parameter p of the Schatten p-norm, and q of the $L_{p,q}$-norm. The results are depicted in Figure 4, from which we have the following observations: (1) Setting the anchor rate as 1 does not always result in optimal performance, indicating that the model does not necessarily perform best when using all available data. This is probably because a large anchor rate may introduce redundant and irrelevant information, which can be removed with the anchor selection method. (2) When p ranges from 0 to 1, an appropriate Schatten p-norm helps improve clustering performance as it better captures the latent spatial distribution and complementary information of data across different clients. (3) When q ranges from 0 to 1, the smaller the value of q is, the more it encourages the row sparsity of the projection matrix. Therefore, an appropriate value of q can effectively remove the redundant connections and noise in the anchor graph and enhance the robustness of the method.

Ablation Experiments: We conduct ablation studies and summarize the results in Table 3. We evaluate the performance of SFOMVC-TR in four cases: (1) two-step SFOMVC-TR **(TS-SFOMVC-TR)**: Each client performs SVD on its predefined anchor graph to extract right singular vectors, then applies K-means clustering on these vectors. The clustering assignment matrices are sent to the server, which averages them to obtain global clustering results. (2) One-step SFOMVC-TR without sparse constraints and tensor regularizer **(w.o. S-C&T-R)**; (3) One-step SFOMVC-TR without sparse constraints **(w.o. S-C)**; (4) One-step SFOMVC-TR without tensor regularizer **(w.o. T-R)**; and (5) the complete SFOMVC-TR **(SFOMVC-TR)**.

From Table 3, SFOMVC-TR w.o. S-C outperforms w.o. S-C & T-R in terms of ACC, NMI, and PUR by 64.80%, 82.59%, and 64.80% on NGs dataset, respectively, indicating that tensor regularization effectively captures consistent information across different views. Additionally, on the 3-sources dataset, the ACC, NMI, and PUR of SFOMVC-TR w.o. T-R increases by 16.58%, 12.46%, and 13.62% compared to SFOMVC-TR w.o. S-C & T-R, which demonstrates that the sparse constraint on the projection helps to remove redundant information and enhances clustering performance. Compared to TS-SFOMVC-TR, SFOMVC-TR demonstrates that our method achieves clustering results in one step without requiring complex post-processing, significantly improving performance across various datasets. The ablation experiment illustrates the contribution and the necessity of each component of SFOMVC-TR.

Conclusion

In this paper, we developed a novel FMVC method based on anchor graph projection and tensor learning, termed SFOMVC-TR, which improves model scalability by leveraging anchor graphs constructed from the raw data for FMVC tasks. Its main idea is to perform one-step clustering on each client via a $L_{p,q}$ sparse regularization under the guidance of the server, and the server enforces the clustering results of each client to be consistent with tensor learning. The developed federated optimization framework avoids the exchange of sensitive information, including both the raw data and the anchor graphs, and eventually produces a global and unified clustering result. Security analysis and performance evaluation indicate the security and effectiveness of the proposed FMVC method.

Acknowledgments

This work is supported by the National Key Research and Development Project of China No. 2021ZD0110700, the National Natural Science Foundation of China under Grant 62176203 and Grant 62102306, the Key Research and Development Project in Shaanxi Province No.2022GXLH-01-03, the Fundamental Research Funds for the Central Universities, the Natural Science Basic Research Program of Shaanxi Province Grant 2023-JC-YB-534, and the Science and Technology Project of Xi'an (Grant 2022JH-JSYF-0009), Initiative Postdocs Supporting Program Grant BX20190262, Open Project of Anhui Provincial Key Laboratory of Multimodal Cognitive Computation, Anhui University No. MMC202416.

References

Apté, C.; Damerau, F.; and Weiss, S. M. 1994. Automated learning of decision rules for text categorization. *ACM Transactions on Information Systems (TOIS)*, 12(3): 233–251.

Blum, A.; and Mitchell, T. 1998. Combining labeled and unlabeled data with co-training. In *Proceedings of the eleventh annual conference on Computational learning theory*, 92–100.

Brbić, M.; and Kopriva, I. 2018. Multi-view low-rank sparse subspace clustering. *Pattern recognition*, 73: 247–258.

Cai, X.; Huang, D.; Zhang, G.-Y.; and Wang, C.-D. 2023. Seeking commonness and inconsistencies: A jointly smoothed approach to multi-view subspace clustering. *Information Fusion*, 91: 364–375.

Chen, X.; Xu, J.; Ren, Y.; Pu, X.; Zhu, C.; Zhu, X.; Hao, Z.; and He, L. 2023. Federated deep multi-view clustering with global self-supervision. In *Proceedings of the 31st ACM International Conference on Multimedia*, 3498–3506.

Fang, S.-G.; Huang, D.; Cai, X.-S.; Wang, C.-D.; He, C.; and Tang, Y. 2023. Efficient multi-view clustering via unified and discrete bipartite graph learning. *IEEE Transactions on Neural Networks and Learning Systems*.

Feng, W.; Wu, Z.; Wang, Q.; Dong, B.; Tao, Z.; and Gao, Q. 2024a. Efficient Federated Multi-View Clustering with Integrated Matrix Factorization and K-Means. In Larson, K., ed., *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI-24*, 3971–3979. International Joint Conferences on Artificial Intelligence Organization. Main Track.

Feng, W.; Wu, Z.; Wang, Q.; Dong, B.; Tao, Z.; and Gao, Q. 2024b. Federated Multi-View Clustering via Tensor Factorization. In Larson, K., ed., *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI-24*, 3962–3970. International Joint Conferences on Artificial Intelligence Organization. Main Track.

Greene, D.; and Cunningham, P. 2006. Practical solutions to the problem of diagonal dominance in kernel document clustering. In *Proceedings of the 23rd international conference on Machine learning*, 377–384.

Hammoud, A.; Otrok, H.; Mourad, A.; and Dziong, Z. 2022. On demand fog federations for horizontal federated learning in IoV. *IEEE Transactions on Network and Service Management*, 19(3): 3062–3075.

He, X.; Wang, B.; Li, R.; Gao, J.; Hu, Y.; Huo, G.; and Yin, B. 2023. Graph structure learning layer and its graph convolution clustering application. *Neural Networks*, 165: 1010–1020.

Hu, X.; Qin, J.; Shen, Y.; Pedrycz, W.; Liu, X.; and Liu, J. 2023. An efficient federated multi-view fuzzy C-means clustering method. *IEEE Transactions on Fuzzy Systems*.

Hu, Y.; Li, C.; Meng, K.; Qin, J.; and Yang, X. 2017. Group sparse optimization via lp, q regularization. *Journal of Machine Learning Research*, 18(30): 1–52.

Huang, D.; Wang, C.-D.; and Lai, J.-H. 2023. Fast multi-view clustering via ensembles: Towards scalability, superiority, and simplicity. *IEEE Transactions on Knowledge and Data Engineering*, 35(11): 11388–11402.

Huang, L.; Li, Z.; Sun, J.; and Zhao, H. 2022a. Coresets for Vertical Federated Learning: Regularized Linear Regression and K-Means Clustering. *Advances in Neural Information Processing Systems*, 35: 29566–29581.

Huang, S.; Shi, W.; Xu, Z.; Tsang, I. W.; and Lv, J. 2022b. Efficient federated multi-view learning. *Pattern Recognition*, 131: 108817.

Hussain, S. F.; Bisson, G.; and Grimal, C. 2010. An improved co-similarity measure for document clustering. In *2010 ninth international conference on machine learning and applications*, 190–197. IEEE.

Kang, Z.; Lin, Z.; Zhu, X.; and Xu, W. 2021. Structured graph learning for scalable subspace clustering: From single view to multiview. *IEEE Transactions on Cybernetics*, 52(9): 8976–8986.

Li, R.; Zhang, C.; Fu, H.; Peng, X.; Zhou, T.; and Hu, Q. 2019. Reciprocal multi-layer subspace learning for multi-view clustering. In *Proceedings of the IEEE/CVF international conference on computer vision*, 8172–8180.

Li, X.; Zhang, H.; Wang, R.; and Nie, F. 2020. Multiview clustering: A scalable and parameter-free bipartite graph fusion method. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(1): 330–344.

Li, Y.; Nie, F.; Huang, H.; and Huang, J. 2015. Large-scale multi-view spectral clustering via bipartite graph. In *Proceedings of the AAAI conference on artificial intelligence*, volume 29.

Liu, C.; Wen, J.; Wu, Z.; Luo, X.; Huang, C.; and Xu, Y. 2023. Information recovery-driven deep incomplete multi-view clustering network. *IEEE Transactions on Neural Networks and Learning Systems*.

Liu, J.; Wang, C.; Gao, J.; and Han, J. 2013. Multi-view clustering via joint nonnegative matrix factorization. In *Proceedings of the 2013 SIAM international conference on data mining*, 252–260. SIAM.

Liu, Y.; Kang, Y.; Zou, T.; Pu, Y.; He, Y.; Ye, X.; Ouyang, Y.; Zhang, Y.-Q.; and Yang, Q. 2024. Vertical federated learning: Concepts, advances, and challenges. *IEEE Transactions on Knowledge and Data Engineering*.

Samaria, F. S.; and Harter, A. C. 1994. Parameterisation of a stochastic model for human face identification. In *Proceedings of 1994 IEEE workshop on applications of computer vision*, 138–142. IEEE.

Tzortzis, G.; and Likas, A. 2012. Kernel-based weighted multi-view clustering. In *2012 IEEE 12th international conference on data mining*, 675–684. IEEE.

Wang, H.; Yang, Y.; and Liu, B. 2019. GMC: Graph-based multi-view clustering. *IEEE Transactions on Knowledge and Data Engineering*, 32(6): 1116–1129.

Wang, J.; Feng, S.; Lyu, G.; and Gu, Z. 2023. Triple-granularity contrastive learning for deep multi-view subspace clustering. In *Proceedings of the 31st ACM International Conference on Multimedia*, 2994–3002.

Wang, S.; Liu, X.; Zhu, X.; Zhang, P.; Zhang, Y.; Gao, F.; and Zhu, E. 2021. Fast parameter-free multi-view subspace clustering with consensus anchor guidance. *IEEE Transactions on Image Processing*, 31: 556–568.

Xu, J.; Ren, Y.; Li, G.; Pan, L.; Zhu, C.; and Xu, Z. 2021. Deep embedded multi-view clustering with collaborative training. *Information Sciences*, 573: 279–290.

Yang, H.; Gao, Q.; Xia, W.; Yang, M.; and Gao, X. 2022. Multiview spectral clustering with bipartite graph. *IEEE Transactions on Image Processing*, 31: 3591–3605.

Yu, S.; Wang, S.; Dong, Z.; Tu, W.; Liu, S.; Lv, Z.; Li, P.; Wang, M.; and Zhu, E. 2024. A non-parametric graph clustering framework for multi-view data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 16558–16567.

Zhang, Y.; Wei, C.; and Liu, X. 2022. Group Logistic Regression Models with Lp, q Regularization. *Mathematics*, 10(13): 2227.

Zhu, X.; Wang, D.; Pedrycz, W.; and Li, Z. 2023. Privacy-Preserving Realization of Fuzzy Clustering and Fuzzy Modeling Through Vertical Federated Learning. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*.