Max-Mahalanobis Anchors Guidance for Multi-View Clustering

Anonymous submission

Abstract

Anchor selection or learning has become a critical component in large-scale multi-view clustering. Existing anchor-based methods, which either select-then-fix or initialize-then-optimize with orthogonality, yield promising performance. However, these methods still suffer from instability and insufficient depiction of data distribution. Moreover, the desired properties of anchors in multi-view clustering remain unspecified. To address these issues, this paper first formalizes the desired characteristics of anchors, namely *Diversity, Balance* and *Compactness*. We then devise and mathematically validate anchors that satisfy these properties by maximizing the Mahalanobis distance between anchors. Furthermore, we introduce a novel approach called **M**ax-Mahalanobis **A**nchors **G**uidance for multi-v**I**ew **C**lustering (MAGIC), which iteratively guides the cross-view representations to progressively align with our well-defined anchors. This process yields highly discriminative and compact representations, significantly enhancing the performance of multi-view clustering. Experimental results show that our meticulously designed anchor strategy significantly outperforms existing anchor-based methods in enhancing anchor efficacy, leading to substantial improvement in multi-view clustering performance.

1 Introduction

Data appears in various forms in the information era, each offering a distinct perspective. However, a single view of data often fails to capture the complexity and heterogeneity of data. In bioinformatics, for example, the intricate behaviors of organisms are determined by gene expression data, protein-protein interaction, and phenotypic characteristics. Isolating any single view may lead to incomplete conclusions (Rappoport and Shamir 2018). Multi-view clustering (MVC) addresses this by processing multiple views simultaneously to reveal their interrelationships. Current multi-view clustering methodologies focus on three core objectives: better representation (Liu et al. 2013; Gao et al. 2015; Sun et al. 2021; Ma et al. 2024), better alignment (Wang et al. 2019; Zhang et al. 2021; Wang et al. 2022) and better fusion (Kang et al. 2020a; Li et al. 2020; Zhang et al. 2023), which are also essential for handling missing views (Li et al. 2023b; Jin et al. 2023; Wen et al. 2023; Yu et al. 2024).

Rapid technological advancement has led to an influx of multi-view data, offering opportunities for more valuable insights but also presenting challenges in processing the increase in data volume. Efficient clustering for large-scale multi-view data has become a key research focus. Current mainstream methods use anchor points to avoid associating all samples. This paper categorizes these anchor-based MVC methods into two groups: *fixed-anchor-based MVC* (Li et al. 2015, 2020; Li and He 2020; Kang et al. 2020b; Xia et al. 2022) and *optimized-anchor-based MVC* (Ou et al. 2024; Wang et al. 2021; Chen et al. 2022b; Li et al. 2023b, 2024). *Fixed-anchor-based MVC* typically generates anchors through pre-processing techniques such as k-means or sampling strategies before the algorithm. Sampling methods, which select anchors randomly or heuristically, are simple and efficient but often lead to poor and unstable performance due to randomness and lack of structure relevance (Xia et al. 2022; Li et al. 2020). Additionally, sampling may introduce more challenges, such as potential discrepancies in anchor correspondence across views. Alternatively, using k-means cluster centers as the anchors generally improves performance by leveraging clustering-relevant information in anchors (Kang et al. 2020b; Li and He 2020; Yang et al. 2022). However, k-means is sensitive to initialization, requiring multiple runs to mitigate randomness, and incorrect anchor initialization may cause disappointed clustering results. *Optimized-anchor-based MVC* improves upon the fixed-anchor strategy by incorporating anchors directly into the multi-view clustering optimization process. These methods typically employ the self-expression concept but use representative points to establish relationships with all samples. To ensure anchor representativeness, they generally impose orthogonal constraints to enforce diversity among anchors (Wang et al. 2021; Liu et al. 2022).

Both *fixed-anchor-based MVC* and *optimized-anchor-based MVC* have their strengths and limitations. Fixed-anchor methods, derived from the original data space, aim to select anchors that better fit the data but suffer from instability in initialization and the complexity of designing effective heuristic strategies. On the other hand, optimized-anchor methods learn anchors during the algorithm, making the acquisition of anchors less manual and more diverse through orthogonal constraints. However, this results in unnecessary sacrifice in the anchors' fitting performance to the data due to overly restrictive orthogonality constraints, particularly when the number of anchors is large.

Given the above research findings, this paper proposes an

explicit definition of desirable anchors in multi-view clustering. In multi-view clustering, we expect to obtain a set of common anchors that possess three key properties: *Diversity*, *Balance*, and *Compactness*. Specifically, *Diversity* implies that the obtained anchors should be as dissimilar as possible, maximizing inter-cluster distances and facilitating more distinguishable data representations. *Balance* refers to the equilibrium between anchors across multi-view data, ensuring robust and stable structure. Data usually lie on a low-dimensional manifold within a high-dimensional space, exhibiting a more compact rather than uniform distribution. As a sketch of the data, the distribution of anchors should also maintain *Compactness* while satisfying the above two criteria, avoiding unnecessary increases in their dimensionality. Revisiting previous anchor-based methods with our defined anchor properties, we conclude that fixed-anchor methods mainly focus on balance, selecting anchors to effectively cover the data distribution. On the other hand, optimized-anchor methods emphasize diversity, striving to learn significantly different anchors to separate the data. However, orthogonal constraints on anchors may be overly restrictive, potentially limiting the model's representational capacity. Besides, this can impede data point discrimination in high-dimensional spaces due to the curse of dimensionality.

Since the ultimate goal of algorithmic models is to continuously fit data to anchors (either fixed or optimized), anchors with superior characteristics can guide the representation to be more discriminative, thereby achieving enhanced clustering performance. In light of the aforementioned analysis, we aim to provide an explicit definition of anchor properties and propose a design strategy for obtaining optimal anchors that satisfy these definitions. Integrating these anchors into our multi-view clustering framework can lead to superior data representations, thereby improving clustering performance. Specifically, we propose an optimal anchor design strategy called Max-Mahalanobis Anchors (MMA), which is carefully designed by maximizing the minimum angle between any two anchors, thereby achieving the promising iter-cluster dispersion effect. This paper then leverages the superior properties inherent in the MMA to achieve more efficacious re-representations of multi-view data, resulting in enhanced clustering performance. In summary, the main contributions of this paper are as follows:

- This paper provides formal definitions and mathematical formulation for the desirable properties (*Diversity, Balance* and *Compactness*) of anchors, revealing the deficiency of current anchor-based MVC methods.
- This paper proposes a rational-design anchor strategy, termed Max-Mahalanobis Anchors (MMA), satisfying the expected properties of anchors with theoretical proof. We integrate the novel MMA into our multi-view clustering framework, guiding the consensus representation to gradually align with our well-designed structure.
- Extensive experiments demonstrate our method's effectiveness. Comparisons with anchor-based MVC highlight our superior anchor performance and data fitting effects, validating the outstanding properties of MMA.

2 Related Work

In this section, we review the rationale and literature of anchor-based multi-view clustering methods. Then, the most relevant algorithms are introduced in detail. The key notations used throughout this paper are listed in Tab. 1.

Notation	Description
N, K, V	Number of samples, clusters and views
d_i	Feature dimension of i-th view
$\mathbf{X}^{(i)} \in \mathbb{R}^{N*d_i}$	Data matrix in the i-th view
$\mathbf{X}^{(i)}_{[j,:]}$	j-th row/sample in the i-th view
$\mathbf{e}_1$	The first unit basis vector
$\mathbf{0}_K$	The zero vector in $\mathbb{R}^K$
$\mathbf{I}_K$	The all one vector in $\mathbb{R}^K$.

Table 1: Description of notations in this paper.

Anchor-based multi-view clustering methods have gained focus for their efficiency in large-scale scenarios, which involve using a small subset of representative points. Both fixed and optimized anchor-based methods fundamentally aim to associate original instances with a few representative points, thereby avoiding exploring global relationships.

Current fixed anchor-based methods obtain anchors using techniques such as random sampling, k-means, or by designing heuristic sampling strategies, which are then utilized for subsequent clustering tasks. Although random sampling methods are simple and efficient, they often result in clustering outcomes that are both unsatisfactory and unstable. Furthermore, researchers employ a meticulously crafted algorithm to directly sample data points as anchors (Xia et al. 2022). To create more representative anchors, Li et al. believe that anchors should effectively cover the entire data distribution. Therefore, they alternately sample anchors based on the feature similarity of data across all clusters. For more cluster-information induced, many studies utilize k-means clustering to generate these anchors and keep them fixed throughout the subsequent process (Li et al. 2015; Yang et al. 2020; Li and He 2020; Yang et al. 2022). Similarly, Kang et al. firstly propose to replace the self-expression in multi-view subspace clustering with anchor-samples expression, making it possible for the subspace-based methods to handle large-scale clustering scenarios.

In contrast to those fixed-anchor-based approaches mentioned above, optimized-anchor-based methods propose an integrated framework that simultaneously optimizes the orthogonal anchors and constructs corresponding anchor graphs. Representative methods (Sun et al. 2021; Wang et al. 2021) consider that the anchors obtained by optimization are more representative than fixed anchors and propose to optimize the orthogonal anchors and construct an anchor graph in a unified framework. Based on it, Liu et al. proposes a one-pass approach to directly obtain the clustering labels by imposing graph connectivity constraints on the anchor graph (Liu et al. 2022). Next, various variants of anchor-based orthogonal optimization methods are considered, incorporating aspects such as fusion (Zhang et al. 2023; Wang et al.

2022; Zhang et al. 2022; Li et al. 2023b), noise (Li et al. 2023a; Liu et al. 2024b), and graph constraints (Liu et al. 2024b,a; Yu et al. 2023; Li et al. 2024) and so on.

Despite the success of the current anchor-based MVC method over the earlier sampling methods, several essential issues persist. The fixed-anchor-based approach offers better alignment and balance in the original space with computational simplicity. However, within it, random sampling and k-means-based methods have inherent randomness, potentially compromising the stability of subsequent algorithms. Moreover, designing anchor selection algorithms is often heuristic and challenging. Optimized anchor-based methods, on the other hand, effectively enhance anchor diversity and reduce the need for manual design by imposing orthogonal constraints. Nevertheless, the orthogonal constraints on the anchors cause tension in the optimization process between fitting the data and maintaining orthogonality during optimization, hindering the achievement of compact representation. In summary, while existing anchor-based methods each have their advantages and disadvantages, the specific characteristics that anchors should possess for multi-view clustering have not yet been explicitly defined in the literature. To address this gap, we will present in the subsequent section our formal definition of anchor properties, along with a novel method that adheres to these properties.

3 Methodology

This section begins with the definition of anchor properties and the construction of anchors. Then, we demonstrate its integration within our multi-view clustering framework.

3.1 Definition of desired anchors

In anchor-based multi-view clustering, we aim to obtain anchors with desirable properties to provide better guidance for multi-view data representation learning. Conceptually, we aspire for the anchors to exhibit properties of *Diversity, Balance,* and *Compactness*. However, it is challenging to constrain the anchors with these properties simultaneously. To achieve this, we propose to leverage the angular relationships between anchors and formulate the following three key definitions of anchor property.

Firstly, Definition 1 formalizes the concept of *Diversity* that the desired anchor should be as distinct as possible to facilitate discriminative capacity. We define this diversity metric by considering the mean of the angles between all unique pairs of anchors to be as large as possible, ensuring that anchors are apart from each other.

Definition 1 (**Diversity**). *Given a set of anchors $\{\mu_i\}_{i=1}^K$, the angle θ_{ij} between any pair of distinct anchors μ_i and μ_j $(i \neq j)$ is expected to be an obtuse angle. Defining the average angle $\bar{\theta} = \frac{2}{K(K-1)} \sum_{i \neq j} \theta_{ij}$, it then subject to $90^\circ \leq \bar{\theta} \leq 180^\circ$. Equally, $-1 \leq \frac{2}{K(K-1)} \sum_{i \neq j} \cos(\theta_{ij}) \leq 0$.*

Secondly, Definition 2 formalizes the concept of *Balance* that anchors should be distributed relatively uniformly, thereby creating a more robust and stable structure. We quantify balance by measuring the variance of angles between different anchor vectors.

Definition 2 (**Balance**). *Given a set of anchors $\{\mu_i\}_{i=1}^K$, the angle θ_{ij} of different pairs of distinct anchors μ_i and μ_j $(i \neq j)$ should be similar with minimal variations, satisfying:*

$$Var(\{\theta_{ij} | \forall i, j \in [K], i \neq j\}) \leq \varepsilon, \tag{1}$$

where $\theta_{ij} = \arccos\left((\mu_i^\top \mu_j)/(\|\mu_i\|_2 \|\mu_j\|_2)\right)$, ε is a small positive threshold. $Var(\cdot)$ is the variance of a set of angles.

Last but not least, while satisfying *Diversity* and *Balance*, the anchor distribution should be as *compact* as possible, which means occupying as small space as possible and avoiding unnecessary increases in dimensionality.

Definition 3 (**Compactness**). *Given a set of anchors $\{\mu_i\}_{i=1}^K$, let $dim(\{\mu_i\}_{i=1}^K)$ denote the dimensionality of the space spanned by the anchors. Under the conditions specified by Definition 1 and Definition 2, $dim(\{\mu_i\}_{i=1}^K)$ is expected to be small, the smaller the better.*

3.2 Generate Max-Mahalanobis Anchors

Unlike previous approaches that relied on unstable initialization or rigid orthogonality constraints on anchors, we propose to simultaneously consider *Diversity, Balance, Compactness* criteria for anchors. Anchors satisfying these criteria leverage geometric properties for a diverse and uniform distribution of anchors within a compact space. Specifically, we propose a strategy called Max-Mahalanobis Anchors (MMA) to achieve these properties.

We approach the problem from the perspective of angular relationships between anchor vectors, aiming to maximize the minimum angle between any pair of distinct anchors. Mathematically, denoting the angle between anchors μ_i and μ_j as θ_{ij}, we formulate the problem as: $\boldsymbol{\mu}^* = \arg\max_{\boldsymbol{\mu}} \min_{i \neq j} \theta_{ij}$. Intuitively, this criterion aims to maximize the angle between any two centers, which means that the anchors are as distant from each other as possible in the anchor feature space. Such dispersion enhances the model's discrimination capacity by reducing inter-cluster overlap, as demonstrated in (Pang, Du, and Zhu 2018; Pang et al. 2020). However, directly manipulating the angles between anchors is challenging. Therefore, we define the Mahalanobis distance between any two anchors μ_i and μ_j as $\Delta_{ij} = \left[(\mu_i - \mu_j)\Sigma^{-1}(\mu_i - \mu_j)\right]^{\frac{1}{2}}$, and the target problem can be equivalently transformed into the following form:

$$\boldsymbol{\mu}^* = \arg\max_{\Delta} \min_{i \neq j} \frac{1}{2} \Delta_{ij}^2. \tag{2}$$

Denoting the minimal distance as $\text{MiD} = \min_{i \neq j} \frac{1}{2} \Delta_{ij}^2$ and $\|\mu_i\|_2^2 = C \ \forall i \in [K]$, where C is a positive constant, the following theorem provides a tight upper bound of MiD.

Theorem 1. *Given a set of anchors $\{\mu_i\}_{i=1}^K$ where $\sum_{i=1}^K \mu_i = \mathbf{0}_K$ and $\|\mu_i\|_2^2 = C$, we can derive an upper bound for* MiD*:*

$$\text{MiD} \leq \frac{KC}{K-1}.$$

The equality holds if and only if

$$\mu_i^\top \mu_j = \begin{cases} C & i = j, \\ C/(1-K) & i \neq j, \end{cases} \tag{3}$$

where $0 \leq i, j \leq K$. See Appendix for detailed proof.

A set of anchors that satisfies Eq. (3) is the optimal anchors $\boldsymbol{\mu}^*$, denoted as Max-Mahalanobis Anchors (MMA), indicating that they reach the maximum of the minimal Mahalanobis distance between any two distinct anchors. To achieve this condition, we design the following strategy to obtain anchors that meet the desired criteria:

a) Initialization: Initialize $\mu_1^* = e_1$ and $\mu_i^* = 0_K \ \forall i \geq 2$, where $e_1 = [1, 0, \cdots, 0]^T \in \mathbb{R}^K$ is first unit basis vector and 0_K is a K-dimension zero vector.

b) Recursive generation: Starting from μ_2^*, recursively generate anchors according to Eq. (4):

$$\mu_i^*(j) = \begin{cases} -\frac{\frac{1}{K-1} + \langle \mu_i^*, \mu_j^* \rangle}{\mu_j^*(j)} & j \neq i, \\ \sqrt{1 - \|\mu_i^*\|_2^2} & j = i, \end{cases} \tag{4}$$

where $2 \leq i \leq K$ and $1 \leq j \leq i$.

c) Uniform scaling: Apply a uniform scaling to the anchors by setting $\mu_k^* = \sqrt{C} \cdot \mu_k^*, \forall k \in [K]$.

In light of the design of MMA, we demonstrate through Theorems 2 and Theorems 3 that the optimal $\boldsymbol{\mu}^*$ generated by MMA exhibit the properties of *Diversity* and *Balance*.

Theorem 2. *Our Max-Mahalanobis Anchors $\boldsymbol{\mu}^*$ strictly satisfies the Diversity property in Definition 1, i.e., the average angle between any two distinct anchors in $\boldsymbol{\mu}^*$ lies between $90°$ and $180°$.*

Proof. Since our MMA $\boldsymbol{\mu}^*$ satisfies Eq. (3), the angle θ_{ij} between any two distinct anchors is constant, i.e.,

$$\theta_{ij} = \frac{\mu_i^\top \mu_j}{\|\mu_i\|_2 \|\mu_j\|_2} = \arccos(\frac{1}{1-K}), \quad \forall 1 \leq i \neq j \leq K,$$

where K is the number of clusters.

Therefore, the average angle satisfies

$$\bar{\theta} = \frac{2}{K(K-1)} \sum_{i \neq j} \theta_{ij} = \arccos(\frac{1}{1-K}).$$

Given that $K \geq 2$ (as there must be at least two clusters), we have $\frac{1}{1-K} \in [-1, 0)$, consequently $\bar{\theta} \in (90°, 180°]$. □

Theorem 3. *Our Max-Mahalanobis Anchors $\boldsymbol{\mu}^*$ strictly possess the Balance property in Definition 2. Particularly, the angles between any two distinct anchors in $\boldsymbol{\mu}^*$ are the same and the variance of them is thus zero.*

Proof. Similar to the proof of Theorem 2, the angle θ_{ij} between any two distinct anchors is constant, i.e., $\theta_{ij} = \arccos(\frac{1}{1-K}), \forall 1 \leq i \neq j \leq K$. Let $\Theta = \{\theta_{ij} | i \neq j\}$, the variance of the angles between any two distinct anchors satisfies $\text{Var}(\Theta) = 0$, indicating that the MMA possess the "Balance" property with $\varepsilon = 0$. □

To elucidate the *Compactness* of MMA, we provide an intuitive understanding of the shape of MMA in low-dimensional cases. When $K = 2$, the MMA correspond to two vertices of a line segment. For $K = 3$, they form three vertices of an equilateral triangle. In the case of $K = 4$, the MMA correspond to the four vertices of a regular tetrahedron. This geometrical phenomenon demonstrates that the anchors generated by our MMA strategy are confined to $K - 1$ dimensions space, which is more compact compared to the d_i dimensions of fixed anchors in the original data space and the K dimensions of orthogonal anchors. The lower-dimensional manifold on which our MMA are distributed offers advantages in measuring distances for clustering tasks in high-dimensional data space.

3.3 MMA guidance for multi-view clustering

Given the multi-view dataset $\mathcal{X} = \{\mathbf{X}^{(i)}\}_{i=1}^V$ composed of V views and N instances, where $\mathbf{X}^{(i)} \in \mathbb{R}^{N \times d_i}$ and d_i is the dimension of the samples from i-view. Having obtained a set of rationally designed anchors $\boldsymbol{\mu}^*$ by MMA, we aim to establish mappings from multiple views to these shared optimal anchors. Typically, we have the following objective:

$$\min_{\mathbf{B}, \{\mathbf{P}^{(i)}\}_{i=1}^V, \boldsymbol{\gamma}} \sum_{i=1}^V \gamma_i^2 \|\mathbf{X}^{(i)} - \mathbf{B}\boldsymbol{\mu}^* \mathbf{P}^{(i)}\|_F^2 + \lambda \|\mathbf{B}\|_F^2 \tag{5}$$

$$\text{s.t. } \mathbf{P}^{(i)} \mathbf{P}^{(i)^\top} = \mathbf{I}_K, \mathbf{B} \geq 0, \mathbf{B}\mathbf{1}_K = \mathbf{1}_N, \boldsymbol{\gamma}^\top \mathbf{1}_V = 1.$$

In Eq. (5), $\boldsymbol{\mu}^* \in \mathbb{R}^{K \times K}$ is fixed to be our designed MMA. The cross view $\mathbf{B} \in \mathbb{R}^{N \times K}$ is the new consensus representation of multi-view data. $\mathbf{P}^{(i)} \in \mathbb{R}^{K \times d_i}$ is the i-th view projection between MMA space and original data space.

In the following, we employ the coordinate descent method to solve the optimization problem Eq. (5), optimizing one variable at a time while keeping the others fixed. Specifically, the optimization comprises three steps:

1) **Optimize $\{\mathbf{P}^{(i)}\}_{i=1}^V$ while fixing $\mathbf{B}$ and $\boldsymbol{\gamma}$.** For each view, we have the following problem w.r.t. $\mathbf{P}^{(i)}$: $\min_{\mathbf{P}^{(i)}} \|\mathbf{X}^{(i)} - \mathbf{B}\boldsymbol{\mu}\mathbf{P}^{(i)}\|_F^2$, s.t. $\mathbf{P}^{(i)}\mathbf{P}^{(i)^\top} = \mathbf{I}_K$. The optimization goal can be simplified to $\max_{\mathbf{P}^{(i)}} \text{Tr}(\mathbf{P}^{(i)^\top} \mathbf{M}_i)$ by expanding the objective and ignoring irrelevant terms, where $\mathbf{M}_i = \boldsymbol{\mu}^\top \mathbf{B}^\top \mathbf{X}^{(i)}$. Assuming the singular value decomposition (SVD) of $\mathbf{M}_i$ is $\mathbf{M}_i = \mathbf{U}_m \Sigma_m \mathbf{V}_m^\top$, the optimal $\mathbf{P}^{(i)}$ is given by $\mathbf{P}^{(i)} = \mathbf{U}_m \mathbf{V}_m^\top$ (Wang et al. 2019).

2) **Optimize $\mathbf{B}$ while fixing $\boldsymbol{\gamma}$ and $\{\mathbf{P}^{(i)}\}_{i=1}^V$.** The optimization problem can be rewritten as the following Quadratic Programming (QP) problem. For each row in $\mathbf{B}$,

$$\min_{\mathbf{b}_j} \frac{1}{2} \mathbf{b}_j \mathbf{Q} \mathbf{b}_j^\top + \mathbf{c}^\top \mathbf{b}_j^\top, \text{ s.t. } \mathbf{b}_j \mathbf{1} = 1, \mathbf{b}_j \geq 0, \tag{6}$$

where $\mathbf{b}_j = \mathbf{B}_{[j,:]} \in \mathbb{R}^{1 \times K}$ refers to the j-th row of $\mathbf{B}$. $\mathbf{Q} = \sum_{i=1}^V \gamma_i^2 \boldsymbol{\mu}\boldsymbol{\mu}^\top + \lambda \mathbf{I}_K$ is a symmetric matrix, and $\mathbf{c}^\top = -\sum_{i=1}^V \gamma_i^2 \mathbf{X}_{[j,:]}^{(i)} \mathbf{P}^{(i)^\top} \boldsymbol{\mu}^\top$. Therefore, the optimization problem for $\mathbf{B}$ is transformed into solving QP problems for each row $\mathbf{b}_j$, which can be efficiently solved and parallelized to accelerate the calculation.

3) **Optimize $\boldsymbol{\gamma}$ while fixing $\mathbf{B}$ and $\{\mathbf{P}^{(i)}\}_{i=1}^V$.** Setting $\beta_i = \|\mathbf{X}^{(i)} - \mathbf{B}\boldsymbol{\mu}\mathbf{P}^{(i)}\|_F^2$, we can obtain the following problem: $\min_{\boldsymbol{\gamma}} \sum_{i=1}^V \gamma_i^2 \beta_i$, s.t. $\boldsymbol{\gamma}^\top \mathbf{1} = 1, \boldsymbol{\gamma} \geq 0$, where $\boldsymbol{\gamma} = [\gamma_1; \cdots; \gamma_V] \in \mathbb{R}^V$. The optimal $\boldsymbol{\gamma}$ can be obtained by $\gamma_i = \frac{\frac{1}{\beta_i}}{\sum_{i=1}^V \frac{1}{\beta_i}}$ according to the Cauchy-Schwarz inequality.

Algorithm 1: MMA guidance for multi-view clustering

Input: Multi-view data $\{\mathbf{X}^{(i)}\}_{i=1}^{v}$, constant C, #clusters K.
Initialize: Initialize $\mathbf{B}$ by concatenating the identity matrix and the zero matrix. Initialize γ_i with the average weight $\frac{1}{V}$.

1: Generate Max-Mahalanobis Anchors $\boldsymbol{\mu}^*$.
2: **while** not converged **do**
3: Update $\mathbf{P}^{(i)} = \mathbf{U}_m \mathbf{V}_m^\top$.
4: Update $\mathbf{B}$ by solving problem (Eq. (6)).
5: Update $\gamma_i = \frac{\frac{1}{\beta_i}}{\sum_{i=1}^{V} \frac{1}{\beta_i}}$.
6: **end while**

Output: Perform k-means on the left singular vector $\mathbf{U}_b$ of $\mathbf{B}$ to obtain the final clustering results.

The overall algorithm process is delineated in Algorithm 1. As iteration progresses, the value of the objective function monotonically decreases and is bounded by zero, thus guaranteeing convergence. To demonstrate the guidance of our MMA on the consensus representation learning, we derive the first-order derivative[1] of the objective function in Eq. (5) w.r.t. $\mathbf{B}$, which is given by $\nabla \mathcal{J}(\mathbf{B}) = 2\mathbf{B}\mathbf{E} - 2\mathbf{G}{\boldsymbol{\mu}^*}^\top$, where $\mathbf{G} = \sum_{i=1}^{V} \gamma_i^2 \mathbf{X}^{(i)} {\mathbf{P}^{(i)}}^\top$ and $\mathbf{E} = \sum_{i=1}^{V} \gamma_i^2 \boldsymbol{\mu}^* {\boldsymbol{\mu}^*}^\top + \lambda \mathbf{I}_K$. After each gradient update, the new consensus representation can be formulated as:

$$\mathbf{B}_{t+1} = \mathbf{B}_t - \nabla \mathcal{J}(\mathbf{B}) = \mathbf{B}_t(\mathbf{I}_K - 2\mathbf{E}) + 2\mathbf{G}{\boldsymbol{\mu}^*}^\top. \quad (7)$$

Eq. (7) shows that the new representation is actually an interpolation between the previous one and our MMA, i.e. $\boldsymbol{\mu}^*$. Given a suitable hyperparameter λ, the new data representation $\mathbf{B}$ will progressively converge towards our MMA throughout the iteration. Specifically, the whole objective continually adjusts the data representation to better align with the underlying structure expressed by our well-designed MMA. This alignment process facilitates cluster separation and enhances the learned representation's overall discriminative ability.

4 Experiment

This section compares MAGIC with state-of-the-art methods. We first introduce the datasets, compared methods and the experimental setup, followed by the detailed analysis.

4.1 Experimental Setup

We conduct experiments on the following ten widely-used datasets: BBC, Wikipedia, Reuters(Amini and Goutte 2013), 100Leaves, Cora, Wiki_fea (Costa Pereira et al. 2014), ALOI-100, VGGFace, YouTubeFace, CIFAR100, Detailed information and links to the datasets are in the Appendix.

We compare our proposed approach with nine state-of-the-art methods. These methods represent a diverse range of techniques in the field of MVC and anchor-based MVC. The compared methods are as follows: **BMVC** (Zhang et al. 2018), **LMVSC** (Kang et al. 2020b), **SFMC** (Li et al. 2020), **FPMVS** (Wang et al. 2021), **OMSC** (Chen et al. 2022b), **AIMC** (Chen et al. 2022a), **AWMVC** (Wan et al. 2023), **SMSC** (Ma et al. 2024), **RCAGL** (Liu et al. 2024b).

For fair comparison, we use the official baseline codes. For methods requiring k-means, we run 50 times to obtain the best results. Optimal parameters are determined via grid search within the recommended parameter ranges. The parameters involved in this study include the constant C and the balance parameter λ. For simplicity, we set $C = 1$, leaving λ as the only hyperparameter that requires tuning. Based on previous research, we use $\lambda \in \{0.01, 1, 10, 100, 1000\}$. We utilized widely used clustering metrics such as Accuracy (ACC), Normalized Mutual Information (NMI), Purity and Fscore, where higher values indicate better performance. All experiments are executed using MATLAB 2023b on an AMD EPYC 7513 32-Core Processor.

4.2 Clustering Performance

We compare our method with the state-of-the-art algorithms on ten datasets. Tab. 2 presents the detailed results for all metrics. We have the following observations:

(1) Our method surpasses competitors on most metrics, achieving the highest ACC and NMI across all datasets. It improves by 18.25%, 7.92%, 6.00%, 11.08%, and 7.22% over the second-best method on the BBC, Reuters, 100Leaves, Cora, and Wiki_fea datasets, respectively, with similar gains across other metrics. This demonstrates the superiority of our approach's clustering performance.

(2) Our method shows robust performance across diverse datasets, from text to image domains, demonstrating adaptability to different multi-view data types. It also excels on datasets with many clusters, achieving the best ACC, NMI, and F-score on ALOI-100, VGGFace, and CIFAR100, highlighting its effectiveness in complex clustering scenarios.

(3) Our proposed MAGIC approach consistently outperforms fixed-anchor-based MVC methods (LMVSC and SFMC) and optimized-anchor-based MVC methods (FPMVS, OMSC, AIMC, and RCAGL) across all datasets. This highlights the effectiveness of our well-designed anchor properties and the model validity. Additionally, methods with optimized anchors generally outperform fixed-anchor methods due to the consideration of anchor diversity and balance through orthogonal constraints.

To illustrate our method's advantages, we visualize the new data representations $\mathbf{B}$ and optimal anchors $\boldsymbol{\mu}$ in Fig. 1, comparing them to FPMVS and OMSC on BBC and Wiki_fea datasets. FPMVS and OMSC show anchor collapse on the BBC dataset shown in the left panel of Fig. 1, leading to incorrect data aggregation, whereas our method displays a diverse anchor distribution, guiding the data to better representations. On the Wiki_fea dataset in the right panel of Fig. 1, our approach achieves a more balanced anchor distribution, leading to compact intra-cluster and distinct inter-cluster separations. This demonstrates the effectiveness of our anchors in improving clustering structures. The visual evidence aligns with quantitative results, highlighting our method's ability to enhance diversity, balance, and compactness, reducing representation collapse and improving representation learning across multiple views.

[1] We omit the constraints for the purposes of interpretation.

Dataset	BMVC	LMVSC	SFMC	FPMVS	OMSC	AIMC	AWMVC	SMSC	RCAGL	Proposed
						ACC				
BBC	55.91	40.29	33.58	32.26	35.91	27.15	65.55	37.23	54.75	**83.80**
Wikipedia	19.05	21.79	46.75	32.61	37.95	56.85	23.95	32.32	30.88	**60.90**
Reuters	35.67	39.83	17.25	41.42	45.42	46.58	44.25	47.50	47.58	**55.50**
100Leaves	71.81	55.75	70.88	34.88	36.56	31.75	72.13	64.81	60.56	**78.13**
Cora	32.57	33.20	30.28	55.76	55.54	31.50	39.18	42.25	50.44	**66.84**
Wiki_fea	43.16	18.70	20.03	31.47	36.74	54.29	21.70	41.42	30.81	**61.51**
ALOI-100	62.86	55.32	67.20	32.95	35.02	31.60	69.22	53.20	39.41	**73.77**
VGGFace	10.29	7.48	3.67	9.70	9.71	10.30	14.52	9.41	12.97	**15.61**
YouTubeFace	47.58	78.79	35.61	71.49	78.26	76.02	83.61	OOM	78.68	**86.55**
CIFAR100	7.71	7.27	1.60	7.16	7.49	7.39	10.77	OOM	9.58	**11.82**
						NMI				
BBC	29.10	10.09	1.87	2.97	6.86	2.69	41.33	18.95	30.34	**62.31**
Wikipedia	6.08	6.47	48.71	17.34	25.60	53.92	11.31	19.75	17.36	**54.06**
Reuters	16.24	21.40	1.46	21.10	20.41	24.54	20.14	22.63	26.04	**30.57**
100Leaves	86.22	79.02	86.33	70.22	74.06	71.32	85.48	82.68	84.43	**90.36**
Cora	10.10	6.93	0.54	30.02	29.78	10.05	24.08	20.68	31.23	**45.43**
Wiki_fea	35.84	5.14	14.31	17.15	21.12	51.87	7.83	31.37	15.02	**54.66**
ALOI-100	76.36	72.58	75.73	64.39	68.56	64.93	81.90	69.16	70.67	**83.05**
VGGFace	14.48	9.41	1.59	12.75	13.12	14.25	17.72	11.06	16.55	**19.27**
YouTubeFace	55.84	82.49	48.46	77.40	82.83	83.35	83.66	OOM	80.89	**84.81**
CIFAR100	13.50	13.57	1.79	13.62	14.19	14.26	17.83	OOM	17.57	**17.83**
						Purity				
BBC	55.91	**87.03**	33.87	37.37	40.58	34.60	65.55	39.12	78.10	83.80
Wikipedia	22.37	46.18	51.08	35.79	43.00	60.90	26.70	34.63	35.50	**61.62**
Reuters	39.83	48.50	17.42	45.33	45.83	46.58	46.58	47.50	52.00	**57.25**
100Leaves	74.81	67.94	72.75	36.63	36.94	32.81	75.06	68.06	80.44	**81.00**
Cora	36.48	**95.61**	30.39	55.76	55.54	38.48	46.20	46.79	63.89	66.84
Wiki_fea	47.17	36.36	23.17	33.67	38.97	60.68	25.47	43.86	34.12	**62.81**
ALOI-100	64.95	64.51	68.20	33.61	36.15	32.82	70.84	55.25	**75.22**	75.08
VGGFace	11.83	10.42	3.84	9.97	10.14	10.83	15.83	10.41	**18.81**	16.65
YouTubeFace	55.26	83.30	42.06	76.22	83.03	84.90	83.74	OOM	84.80	**86.57**
CIFAR100	8.51	10.08	1.88	7.34	7.63	7.56	12.09	OOM	**21.15**	12.89
						Fscore				
BBC	42.90	37.61	37.97	27.59	28.75	24.77	51.38	40.88	48.55	**73.64**
Wikipedia	12.42	17.62	32.26	21.12	26.65	49.42	15.04	20.52	19.09	**50.72**
Reuters	26.70	28.87	28.41	30.64	32.86	33.22	30.86	32.51	33.79	**40.35**
100Leaves	62.25	41.54	35.48	22.42	20.82	17.51	62.28	53.57	47.37	**71.23**
Cora	23.41	30.88	30.39	37.35	37.16	21.91	31.70	29.85	38.43	**48.39**
Wiki_fea	35.92	15.82	19.09	21.46	23.77	48.07	14.32	30.97	19.30	**54.84**
ALOI-100	50.04	41.31	12.05	17.99	19.50	13.32	57.73	35.43	24.06	**59.73**
VGGFace	5.34	3.73	4.15	5.75	5.70	5.79	7.50	4.21	6.73	**8.03**
YouTubeFace	42.02	77.38	29.40	69.60	74.63	77.77	**83.28**	OOM	66.49	79.78
CIFAR100	3.47	3.01	1.98	3.40	3.46	3.40	4.49	OOM	4.14	**5.01**

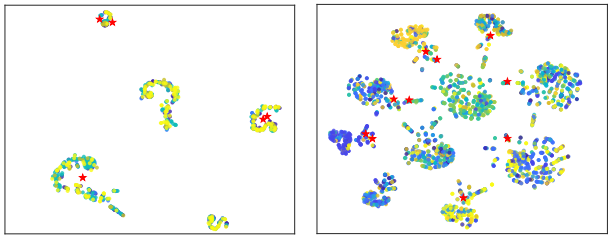
Table 2: The clustering performance comparison across ten datasets. The best results are in bold, while the second-best results are indicated with an underline. "OOM" denotes that the algorithm encountered an out-of-memory error on our device.

4.3 Running time

We compared our method's running time against other baselines across all datasets. Fig. 2 shows the runtime comparison, with the y-axis scaled $\log_{10}$ to accommodate the wide range of execution times. Our method shows competitive efficiency, performing well on smaller datasets like BBC and Wikipedia, outpacing several baselines. Our approach maintains reasonable runtime while some methods fail to complete for larger, complex datasets (e.g., VGGFace, YouTubeFace, CIFAR100). Although BMVC and AIMC occasionally show faster runtime, our method consistently delivers superior clustering performance with acceptable runtime.

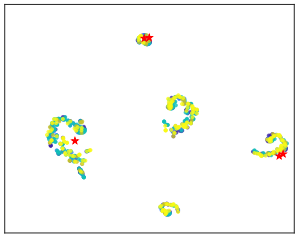
4.4 Convergence and Parameter Analysis

The convergence curve in Fig. 3(a) for the Wiki_fea dataset shows that the objective function value decreases monotonically with the alternating variable updates, confirming the algorithm's convergence. The ACC curve increases as iter-

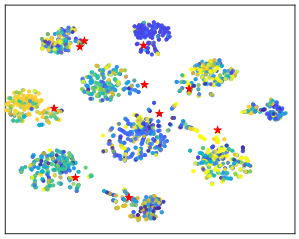


(a) BBC (FPMVS)

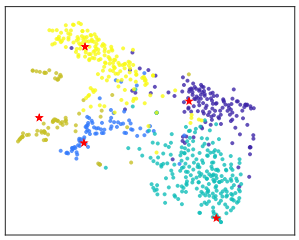
(b) Wiki_fea (FPMVS)



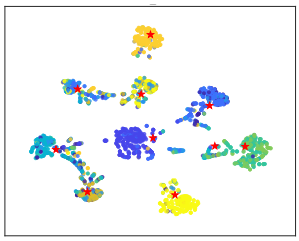
(c) BBC (OMSC)



(d) Wiki_fea (OMSC)



(e) BBC (MAGIC)



(f) Wiki_fea (MAGIC)

Figure 1: Comparison of feature and anchor visualizations using t-SNE on BBC and Wiki_fea datasets.

ations progress, indicating improved accuracy with convergence. Clustering performance rapidly improves alongside the objective function's decline.

The proposed method has one hyperparameter λ, balancing the regularization term. Fig. 3(b) shows ACC results across datasets under different λ. Our method performs stable, achieving satisfactory ACC for λ ranging from 1 to 100, demonstrating robustness to λ selection. Generally, we recommend $\lambda = 10$ across most datasets.

4.5 Ablation Study

We conduct an ablation study to assess our proposed MMA" component's effectiveness across anchor-based MVC methods. We replaced orthogonal anchors with our MMA module in baselines SMVSC, OMSC(K), and AIMC, with MMA-enhanced variants denoted by appending "-MMA". $\triangle$ refers to the variation relative to the original method. Tab. 3 shows ACC improvement across datasets with MMA component. SMVSC-MMA significantly outperforms original SMVSC, with improvements ranging from +10.08% to +47.89%. Similarly, AIMC-MMA consistently surpasses AIMC. OMSC(K)-MMA also shows improvements in most cases, particularly on the YouTubeFace dataset (+4.98%), with a slight decrease on the ALOI-100 dataset. These results validate our MMA's effectiveness in boosting MVC accuracy, highlighting its potential for further applications.

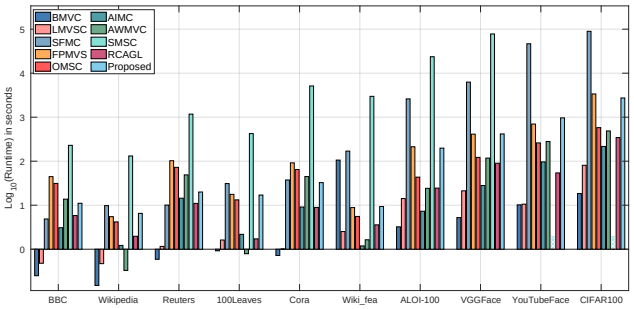
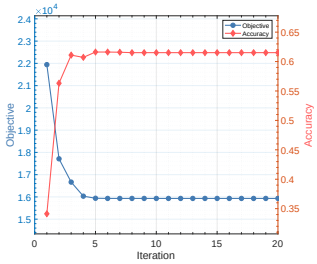
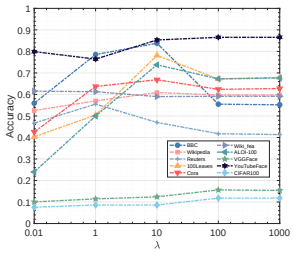


Figure 2: Runtime comparison across all datasets.



(a) Convergence on Wiki_fea.



(b) The impact of different λ

Figure 3: The convergence curves and parameter analysis.

Method	BBC	100Leaves	ALOI-100	YouTubeFace
SMVSC	35.91	38.19	34.82	76.47
SMVSC-MMA	83.80	78.13	73.77	86.55
$\triangle$	(+47.89)	(+39.94)	(+38.95)	(+10.08)
OMSC(K)	32.41	34.31	32.91	71.49
OMSC(K)-MMA	35.91	34.94	32.69	76.47
$\triangle$	(+3.50)	(+0.62)	(-0.22)	(+4.98)
AIMC	27.15	31.75	31.60	76.02
AIMC-MMA	28.91	32.56	33.91	76.44
$\triangle$	(+1.75)	(+0.81)	(+2.31)	(+0.42)

Table 3: Ablation study of MMA's impact on accuracy.

5 Conclusion

This paper introduces MAGIC, a novel approach addressing the limitations of existing anchor-based MVC methods. Firstly, we theoretically formalize the properties of *Diversity*, *Balance* and *Compactness* inherent in anchors. Then, we propose a rational strategy MMA, considering inter-anchor Mahalanobis distances to meet the properties with theoretical guarantees. Furthermore, MAGIC iteratively aligns features with well-designed MMA, enhancing representation discriminability and cluster clarity, thereby improving MVC performance. Experiments confirm our MMA's superiority in MVC. Additionally, our approach is applicable to any task requiring anchors. Future work includes a deeper exploration of anchor properties and their integration with multi-view scenarios.

References

Amini, M.-R.; and Goutte, C. 2013. Reuters RCV1 RCV2 Multilingual, Multiview Text Categorization Test collection. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5FS5R.

Chen, M.-S.; Liu, T.; Wang, C.-D.; Huang, D.; and Lai, J.-H. 2022a. Adaptively-weighted integral space for fast multiview clustering. In *Proceedings of the 30th ACM international conference on multimedia*, 3774–3782.

Chen, M.-S.; Wang, C.-D.; Huang, D.; Lai, J.-H.; and Yu, P. S. 2022b. Efficient orthogonal multi-view subspace clustering. In *Proceedings of the 28th ACM SIGKDD conference on knowledge discovery and data mining*, 127–135.

Costa Pereira, J.; Coviello, E.; Doyle, G.; Rasiwasia, N.; Lanckriet, G.; Levy, R.; and Vasconcelos, N. 2014. On the role of Correlation and Abstraction in Cross-Modal Multimedia Retrieval. *Transactions of Pattern Analysis and Machine Intelligence*, 36(3): 521–535.

Gao, H.; Nie, F.; Li, X.; and Huang, H. 2015. Multi-view subspace clustering. In *Proceedings of the IEEE international conference on computer vision*, 4238–4246.

Jin, J.; Wang, S.; Dong, Z.; Liu, X.; and Zhu, E. 2023. Deep incomplete multi-view clustering with cross-view partial sample and prototype alignment. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 11600–11609.

Kang, Z.; Zhao, X.; Peng, C.; Zhu, H.; Zhou, J. T.; Peng, X.; Chen, W.; and Xu, Z. 2020a. Partition level multiview subspace clustering. *Neural Networks*, 122: 279–288.

Kang, Z.; Zhou, W.; Zhao, Z.; Shao, J.; Han, M.; and Xu, Z. 2020b. Large-scale multi-view subspace clustering in linear time. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 4412–4419.

Li, L.; and He, H. 2020. Bipartite graph based multi-view clustering. *IEEE transactions on knowledge and data engineering*, 34(7): 3111–3125.

Li, L.; Pan, Y.; Liu, J.; Liu, Y.; Liu, X.; Li, K.; Tsang, I. W.; and Li, K. 2024. Bgae: Auto-encoding multi-view bipartite graph clustering. *IEEE Transactions on Knowledge and Data Engineering*.

Li, L.; Zhang, J.; Wang, S.; Liu, X.; Li, K.; and Li, K. 2023a. Multi-view bipartite graph clustering with coupled noisy feature filter. *IEEE Transactions on Knowledge and Data Engineering*, 35(12): 12842–12854.

Li, X.; Sun, Y.; Sun, Q.; Dai, J.; and Ren, Z. 2023b. Distribution Consistency based Fast Anchor Imputation for Incomplete Multi-view Clustering. In *Proceedings of the 31st ACM International Conference on Multimedia*, 368–376.

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 *Twenty-Ninth AAAI Conference on Artificial Intelligence*.

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. Society for Industrial and Applied Mathematics.

Liu, S.; Liang, K.; Dong, Z.; Wang, S.; Yang, X.; Zhou, S.; Zhu, E.; and Liu, X. 2024a. Learn from View Correlation: An Anchor Enhancement Strategy for Multi-view Clustering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 26151–26161.

Liu, S.; Liao, Q.; Wang, S.; Liu, X.; and Zhu, E. 2024b. Robust and Consistent Anchor Graph Learning for Multi-View Clustering. *IEEE Transactions on Knowledge and Data Engineering*.

Liu, S.; Wang, S.; Zhang, P.; Xu, K.; Liu, X.; Zhang, C.; and Gao, F. 2022. Efficient one-pass multi-view subspace clustering with consensus anchors. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, 7576–7584.

Ma, H.; Wang, S.; Zhang, J.; Yu, S.; Liu, S.; Liu, X.; and He, K. 2024. Symmetric Multi-view Subspace Clustering with Automatic Neighbor Discovery. *IEEE Transactions on Circuits and Systems for Video Technology*, 1–1.

Ou, Q.; Wang, S.; Zhang, P.; Zhou, S.; and Zhu, E. 2024. Anchor-based multi-view subspace clustering with hierarchical feature descent. *Information Fusion*, 106: 102225.

Pang, T.; Du, C.; and Zhu, J. 2018. Max-Mahalanobis Linear Discriminant Analysis Networks. In Dy, J.; and Krause, A., eds., *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, 4016–4025. PMLR.

Pang, T.; Xu, K.; Dong, Y.; Du, C.; Chen, N.; and Zhu, J. 2020. Rethinking Softmax Cross-Entropy Loss for Adversarial Robustness. In *International Conference on Learning Representations*.

Rappoport, N.; and Shamir, R. 2018. Multi-omic and multiview clustering algorithms: review and cancer benchmark. *Nucleic acids research*, 46(20): 10546–10562.

Sun, M.; Zhang, P.; Wang, S.; Zhou, S.; Tu, W.; Liu, X.; Zhu, E.; and Wang, C. 2021. Scalable multi-view subspace clustering with unified anchors. In *Proceedings of the 29th ACM International Conference on Multimedia*, 3528–3536.

Wan, X.; Liu, X.; Liu, J.; Wang, S.; Wen, Y.; Liang, W.; Zhu, E.; Liu, Z.; and Zhou, L. 2023. Auto-weighted multi-view clustering for large-scale data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 10078–10086.

Wang, S.; Liu, X.; Liu, S.; Jin, J.; Tu, W.; Zhu, X.; and Zhu, E. 2022. Align then fusion: Generalized large-scale multi-view clustering with anchor matching correspondences. *Advances in Neural Information Processing Systems*, 35: 5882–5895.

Wang, S.; Liu, X.; Zhu, E.; Tang, C.; Liu, J.; Hu, J.; Xia, J.; and Yin, J. 2019. Multi-view Clustering via Late Fusion Alignment Maximization. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intel-*

ligence, IJCAI-19, 3778–3784. International Joint Conferences on Artificial Intelligence Organization.

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.

Wen, J.; Liu, C.; Xu, G.; Wu, Z.; Huang, C.; Fei, L.; and Xu, Y. 2023. Highly confident local structure based consensus graph learning for incomplete multi-view clustering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 15712–15721.

Xia, W.; Gao, Q.; Wang, Q.; Gao, X.; Ding, C.; and Tao, D. 2022. Tensorized bipartite graph learning for multi-view clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(4): 5187–5202.

Yang, B.; Zhang, X.; Li, Z.; Nie, F.; and Wang, F. 2022. Efficient multi-view K-means clustering with multiple anchor graphs. *IEEE Transactions on Knowledge and Data Engineering*, 35(7): 6887–6900.

Yang, B.; Zhang, X.; Nie, F.; Wang, F.; Yu, W.; and Wang, R. 2020. Fast multi-view clustering via nonnegative and orthogonal factorization. *IEEE Transactions on Image Processing*, 30: 2575–2586.

Yu, S.; Liu, S.; Wang, S.; Tang, C.; Luo, Z.; Liu, X.; and Zhu, E. 2023. Sparse low-rank multi-view subspace clustering with consensus anchors and unified bipartite graph. *IEEE Transactions on Neural Networks and Learning Systems*.

Yu, S.; Wang, S.; Zhang, P.; Wang, M.; Wang, Z.; Liu, Z.; Fang, L.; Zhu, E.; and Liu, X. 2024. Dvsai: Diverse view-shared anchors based incomplete multi-view clustering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 16568–16577.

Zhang, C.; Wang, S.; Liu, J.; Zhou, S.; Zhang, P.; Liu, X.; Zhu, E.; and Zhang, C. 2021. Multi-view clustering via deep matrix factorization and partition alignment. In *Proceedings of the 29th ACM International Conference on Multimedia*, 4156–4164.

Zhang, P.; Wang, S.; Li, L.; Zhang, C.; Liu, X.; Zhu, E.; Liu, Z.; Zhou, L.; and Luo, L. 2023. Let the data choose: Flexible and diverse anchor graph fusion for scalable multi-view clustering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 11262–11269.

Zhang, T.; Liu, X.; Zhu, E.; Zhou, S.; and Dong, Z. 2022. Efficient anchor learning-based multi-view clustering–a late fusion method. In *Proceedings of the 30th ACM International Conference on Multimedia*, 3685–3693.

Zhang, Z.; Liu, L.; Shen, F.; Shen, H. T.; and Shao, L. 2018. Binary multi-view clustering. *IEEE transactions on pattern analysis and machine intelligence*, 41(7): 1774–1782.

Reproducibility Checklist

1. This paper:
   (a) Includes a conceptual outline and/or pseudocode description of AI methods introduced (yes/partial/no/NA) [Yes]
   (b) Clearly delineates statements that are opinions, hypothesis, and speculation from objective facts and results (yes/no) [Yes]
   (c) Provides well marked pedagogical references for less-familiare readers to gain background necessary to replicate the paper (yes/no) [Yes]
2. Does this paper make theoretical contributions? (yes/no) [Yes]
   (a) All assumptions and restrictions are stated clearly and formally. (yes/partial/no) [Yes]
   (b) All novel claims are stated formally (e.g., in theorem statements). (yes/partial/no) [Yes]
   (c) Proofs of all novel claims are included. (yes/partial/no) [Partial]
   (d) Proof sketches or intuitions are given for complex and/or novel results. (yes/partial/no) [Yes]
   (e) Appropriate citations to theoretical tools used are given. (yes/partial/no) [Yes]
   (f) All theoretical claims are demonstrated empirically to hold. (yes/partial/no/NA) [Yes]
   (g) All experimental code used to eliminate or disprove claims is included. (yes/no/NA) [Yes]
3. Does this paper rely on one or more datasets? (yes/no) [Yes]
   (a) A motivation is given for why the experiments are conducted on the selected datasets (yes/partial/no/NA) [Yes]
   (b) All novel datasets introduced in this paper are included in a data appendix. (yes/partial/no/NA) [Yes]
   (c) All novel datasets introduced in this paper will be made publicly available upon publication of the paper with a license that allows free usage for research purposes. (yes/partial/no/NA) [Yes]
   (d) All datasets drawn from the existing literature (potentially including authors' own previously published work) are accompanied by appropriate citations. (yes/no/NA) [Yes]
   (e) All datasets drawn from the existing literature (potentially including authors' own previously published work) are publicly available. (yes/partial/no/NA) [Yes]
   (f) All datasets that are not publicly available are described in detail, with explanation why publicly available alternatives are not scientifically satisficing. (yes/partial/no/NA) [Yes]
4. Does this paper include computational experiments? (yes/no) [Yes]
   (a) Any code required for pre-processing data is included in the appendix. (yes/partial/no). [Yes]
   (b) All source code required for conducting and analyzing the experiments is included in a code appendix. (yes/partial/no) [Yes]
   (c) All source code required for conducting and analyzing the experiments will be made publicly available upon publication of the paper with a license that allows free usage for research purposes. (yes/partial/no) [Yes]

(d) All source code implementing new methods have comments detailing the implementation, with references to the paper where each step comes from (yes/partial/no) [Yes]

(e) If an algorithm depends on randomness, then the method used for setting seeds is described in a way sufficient to allow replication of results. (yes/partial/no/NA) [Yes]

(f) This paper specifies the computing infrastructure used for running experiments (hardware and software), including GPU/CPU models; amount of memory; operating system; names and versions of relevant software libraries and frameworks. (yes/partial/no) [Yes]

(g) This paper formally describes evaluation metrics used and explains the motivation for choosing these metrics. (yes/partial/no) [Yes]

(h) This paper states the number of algorithm runs used to compute each reported result. (yes/no) [Yes]

(i) Analysis of experiments goes beyond single-dimensional summaries of performance (e.g., average; median) to include measures of variation, confidence, or other distributional information. (yes/no) [Yes]

(j) The significance of any improvement or decrease in performance is judged using appropriate statistical tests (e.g., Wilcoxon signed-rank). (yes/partial/no) [No]

(k) This paper lists all final (hyper-)parameters used for each model/algorithm in the paper's experiments. (yes/partial/no/NA) [Yes]

(l) This paper states the number and range of values tried per (hyper-) parameter during development of the paper, along with the criterion used for selecting the final parameter setting. (yes/partial/no/NA) [Yes]