symmetry

MDPI

Article

Correntropy-Guided Tensor Graph Learning for Robust Semi-Supervised Multi-View Clustering

Lin Hu [1], Song Jiang [1,*], Xiu Liu [2,*], Pucha Song [1], Yue Yu [1] and Nan Zhou [1]

[1] College of Electronic Information and Electrical Engineering, Chengdu University, Chengdu 610106, China; hulin@cdu.edu.cn (L.H.); songpucha@cdu.edu.cn (P.S.); yuyue@cdu.edu.cn (Y.Y.); zhounan123@cdu.edu.cn (N.Z.)
[2] School of Mathematics, Southwest Jiaotong University, Chengdu 611756, China
* Correspondence: ztgfh_5666@163.com (S.J.); xiuliu@swjtu.edu.cn (X.L.)

Abstract

Multi-view clustering has emerged as a significant research direction in the information age, as multiple feature representations become increasingly available. However, traditional multi-view clustering methods are often unsupervised and fail to exploit available label information. In practice, fully labeled data are scarce, while partially labeled data are more common and can significantly improve clustering performance. Moreover, real-world data are frequently corrupted by noise or outliers. To address these challenges, this paper proposes a Correntropy-guided Matrix Factorization and Tensor Graph Learning (CMTGL) framework for robust semi-supervised multi-view clustering. Specifically, CMTGL incorporates partial label information into a shared low-dimensional representation through constrained low-rank matrix factorization and employs the maximum correntropy criterion (MCC) to reduce the influence of noisy samples and outliers. In addition, view-specific graphs are stacked into a third-order tensor and regularized by the tensor Schatten p-norm to exploit high-order correlations across multiple views. A consensus graph is jointly learned to capture the common structural information shared among different views. The resulting optimization problem is efficiently solved by a block coordinate descent algorithm with an extrapolation accelerated block coordinate update (BCU) scheme. Extensive experiments on six public benchmark datasets demonstrate that the proposed method achieves superior clustering performance compared to state-of-the-art approaches.

Keywords: low-rank matrix factorization; semi-supervised learning; tensor low rank constraints; maximum correntropy criterion

1. Introduction

With the rapid development of the information age, multi-view clustering has emerged as a mainstream data analytics paradigm for multimedia vision tasks. Massive amounts of data collected from social media platforms, multi-sensor systems and biomedical imaging naturally carry multiple heterogeneous feature modalities, where each view delivers distinct descriptive information of identical samples. Such multi-source complementary features have proven vital in various complex visual and cognitive tasks. For instance, recent studies have successfully integrated electroencephalogram (EEG) and eye movement features for enhanced multi-modal emotion recognition [1], and designed joint learning frameworks to extract discriminative features for facial expression recognition under severely low-light conditions [2]. Such multi-source complementary features can significantly boost

Received: 7 August 2026
Revised: 3 September 2026
Accepted: 9 September 2026
Published: 11 September 2026

clustering discrimination, yet conventional single-view clustering algorithms cannot fully exploit cross-view correlations and supplementary cues. As a result, multi-view clustering has become a thriving research hotspot. Existing multi-view clustering approaches can be roughly categorized into five mainstream branches: co-training clustering methods [3,4], graph-based clustering methods [5,6], subspace clustering methods [7–9], multi-kernel learning clustering methods [10,11], and deep learning clustering methods [12,13]. This paper focuses primarily on graph-based multi-view clustering algorithms, which explicitly model sample local geometric structures via similarity graphs.

Representative graph-based works are reviewed as follows. Lv et al. [14] proposed Multi-View Subspace Clustering (MVSC), which simultaneously learns view-specific subspace embeddings and performs clustering within a unified shared subspace. Tang et al. [15] built individual similarity graphs for each view and propagated node information across views through cross-view graph diffusion to capture inter-view dependencies, followed by spectral clustering on fused graphs. Zheng et al. [16] designed a Weighted Multi-Kernel Multi-View Subspace Clustering (WMKMSC) framework with robust multi-kernel weighting to learn an optimal global consensus kernel. Recent studies have further investigated nonconvex tensor representations for multi-view subspace clustering. By incorporating deep priors into nonconvex low-rank tensor learning, both high-order global correlations and local geometric structures can be jointly preserved, leading to more expressive multi-view representations [17].

Nevertheless, most traditional multi-view clustering algorithms fall into the unsupervised paradigm and overlook partially labeled samples widely accessible in real-world scenarios. Moreover, these methods adopt squared reconstruction loss, which is extremely vulnerable to noise and outliers. When partial views are heavily contaminated, clustering performance degrades drastically. Addressing data corruption and noisy labels is a persistent challenge across machine learning domains; for example, integrating correntropy with traditional cross-entropy has been demonstrated to yield highly robust loss functions against noisy labels [18]. Motivated by the robust properties of correntropy, our framework seeks to suppress the negative impact of such outliers. To intuitively demonstrate the negative impact of heterogeneous view corruption and verify the necessity of our robust semi-supervised design, we first conduct controlled synthetic toy experiments as illustrated in Figure 1.

We construct a synthetic dataset with 10 clusters and four independent views injected with outlier rates of 5%, 12%, 22% and 40% respectively. Experimental observations reveal that clustering quality declines sharply as view corruption intensifies, and naive multi-view fusion without adaptive weighting will be dominated by noisy views. In contrast, the proposed framework automatically assigns small weights to heavily polluted views and suppresses large reconstruction residuals introduced by outliers, yielding a clean block-diagonal consensus graph. This synthetic evidence motivates our unified model that jointly exploits partial label cues and robust noise-resistant constraints.

To address the above drawbacks, recent studies introduce semi-supervised learning into multi-view clustering to make full use of limited labeled data. A series of semi-supervised multi-view clustering techniques have been developed, including graph-based, kernel-based and tensor decomposition-based pipelines. Among these developments, low-rank matrix factorization techniques have been elegantly combined with bipartite graph regularization to achieve fast and robust semi-supervised image clustering, significantly reducing computational complexity for large-scale datasets while fully exploiting partial labels [19]. Xin et al. [20] built a semi-supervised re-identification framework to aggregate heterogeneous features from multiple Convolutional Neural Network (CNN) models and generate pseudo-labels for unlabeled instances with only a small fraction of annotated

samples. Nie et al. [21] proposed a joint optimization framework for clustering and semi-supervised classification, which learns an optimal partitionable graph by capturing local data structures simultaneously. Cai et al. [22] constructed a shared label constraint matrix across all views and adopted constrained sparse non-negative matrix factorization to integrate label information into view-wise representations. Jiang et al. [23] presented Tensorial Multi-View Clustering (TMvC), which constructs high-order graphs in tensor space to fuse global multi-view holistic information. Yu et al. [24] further developed a tensor-based semi-supervised multi-view subspace clustering method that jointly optimizes subspace learning and constraint propagation via tensor low-rank representation, avoiding two-stage error accumulation.

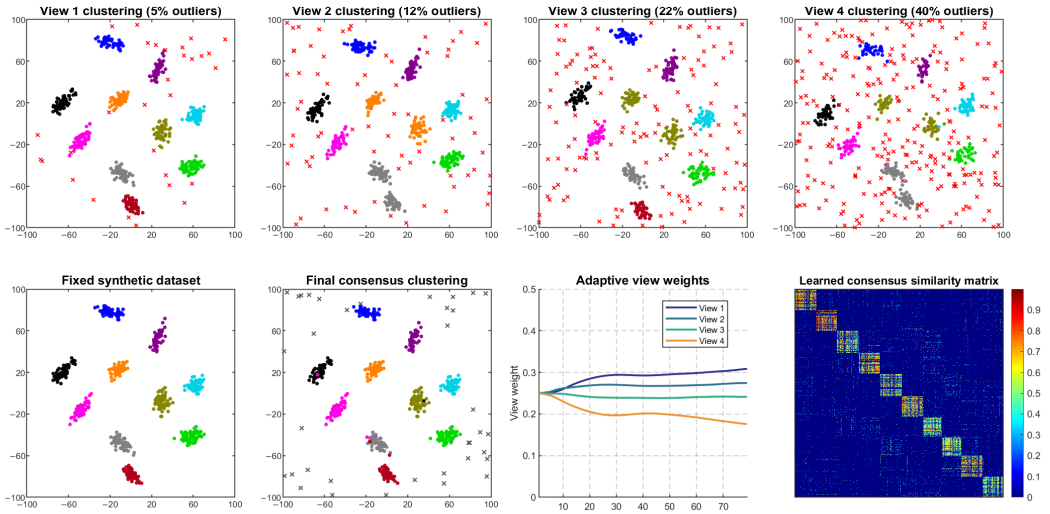


Figure 1. Visualization of the synthetic toy experiment.Single-view clustering results with outlier ratios of 5%, 12%, 22% and 40%, respectively. Ground-truth cluster distribution and the consensus clustering result of CMTGL, where gray crosses mark injected outliers. Adaptive view weight curves during iteration and the learned block-diagonal consensus similarity matrix. This figure intuitively shows that single-view clustering performance degrades sharply with rising outlier rates, which illustrates the motivation for robust multi-view fusion.

Against this background, we propose a Correntropy-guided Matrix Factorization and Tensor Graph Learning (CMTGL) framework for robust semi-supervised multi-view clustering. The overall framework is illustrated in Figure 2. First, a label-constrained low-rank matrix factorization model is constructed to exploit the available partial label information and learn a shared low-dimensional representation across multiple views. Second, the maximum correntropy criterion is incorporated into the reconstruction terms to suppress the influence of samples with large reconstruction errors, thereby improving robustness against noise and outliers. Third, the view-specific graph matrices are stacked into a third-order tensor and regularized by the tensor Schatten p-norm to capture high-order correlations among different views. The strategy of introducing sophisticated entropy and structural constraints to enhance model representation is widely applicable across domains. As seen in other structural reconstruction tasks, incorporating constraints such as wavelet energy entropy has proven highly effective in regularizing recursive neural networks for the super-resolution analysis of medical images [25]. Finally, a consensus graph is jointly learned to preserve the common structural information shared across views. To solve the resulting optimization problem, we employ a Fenchel-conjugate reformulation and develop an accelerated block coordinate update scheme. Experimental results demonstrate that the proposed method exhibits excellent clustering performance on various datasets. The contributions of this article include the following:

- We propose a novel Correntropy-guided Matrix Factorization and Tensor Graph Learning (CMTGL) framework for semi-supervised multi-view clustering, which jointly exploits partial label information, robust representation learning, and high-order cross-view structural correlations.
- We construct a third-order tensor by stacking view-specific graph matrices and impose tensor Schatten p-norm regularization to obtain a flexible low-rank approximation and capture high-order dependencies among multiple views.
- We develop an adaptive consensus graph learning strategy guided by the label-constrained representation, which jointly preserves view-specific local structures and common cross-view information while reducing the influence of corrupted views.
- We develop an accelerated block coordinate update scheme based on a Fenchel-conjugate reformulation to efficiently solve the resulting nonconvex optimization problem.

The remainder of this paper is structured as follows. Section 2 reviews the representative existing multi-view clustering literature. Section 3 presents the formulation of the proposed CMTGL model, while Section 4 introduces the corresponding optimization procedure. Section 5 reports the experimental results, including benchmark comparisons, ablation studies, robustness analysis, parameter sensitivity analysis, and empirical convergence analysis. Finally, Section 6 concludes the paper.

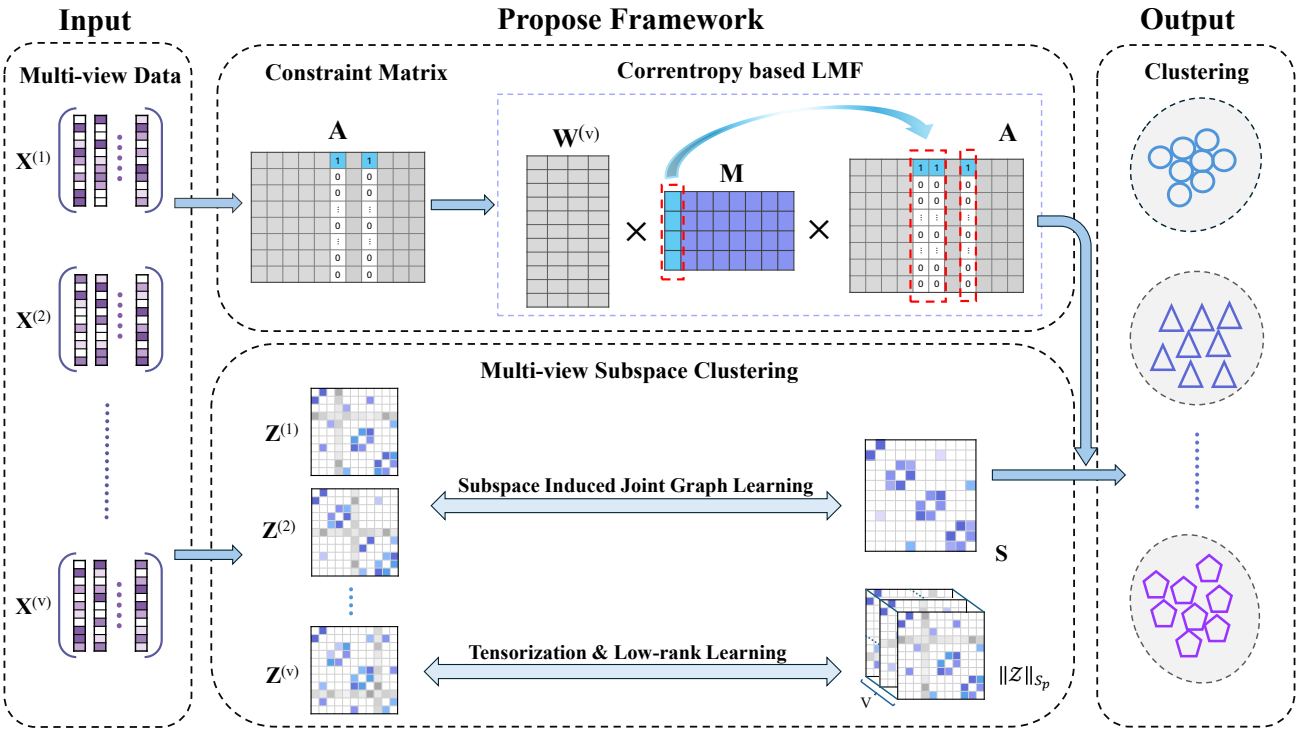


Figure 2. Overall framework of the proposed CMTGL. Given multi-view input data, the upper branch implements correntropy-based low-rank matrix factorization (LMF) with the label constraint matrix $\mathbf{A}$, learning the shared latent representation matrix $\mathbf{M}$ with view-specific basis matrices $\mathbf{W}^{(v)}$. The lower branch constructs view-specific similarity graphs $\{\mathbf{Z}^{(v)}\}$, stacks them into a third-order tensor regularized by tensor Schatten p-norm, and learns the global consensus graph $\mathbf{S}$ via subspace-induced joint graph learning. The two branches are jointly optimized to produce the final clustering results.

2. Notations and Preliminaries

This section introduces the main notations and several preliminaries used in the proposed model, including the maximum correntropy criterion, low-rank matrix factorization, and the tensor Schatten p-norm regularization for multi-view graph learning.

2.1. Notations

The main notations used in this paper are summarized in Table 1.

Table 1. Notations and their descriptions.

Notation	Description
$\mathcal{X}, \mathbf{X}, \mathbf{x}$	Tensor, matrix, and vector, respectively
$\mathbf{X}^v \in \mathbb{R}^{d^v \times n}$	Feature matrix of the v-th view
$\mathbf{W}^v \in \mathbb{R}^{d^v \times r}$	Basis matrix of the v-th view
$\mathbf{Z}^v \in \mathbb{R}^{n \times n}$	View-specific similarity graph of the v-th view
$\mathbf{S} \in \mathbb{R}^{n \times n}$	Consensus similarity graph shared by all views
$\|\cdot\|_F$	Frobenius norm
$\mathcal{Z}$	Third-order tensor by stacking $\{\mathbf{Z}^v\}_{v=1}^V$
$\mathcal{Z}_{ijk}$	The (i,j,k)-th entry of $\mathcal{Z}$
$\mathcal{Z}(:,:,k)$	$\mathcal{Z}$ the k-th frontal slice
$\bar{\mathcal{Z}}$	FFT of $\mathcal{Z}$ along the third dimension
$\|\mathcal{Z}\|_{S_p}$	Tensor Schatten p-norm of $\mathcal{Z}$
$\alpha, \beta, \lambda, \gamma$	Trade-off parameters
ρ	Regularization parameter for the S_p-norm term

The Fast Fourier Transform (FFT) of a tensor $\mathcal{X}$ along the third dimension is denoted by $\bar{\mathcal{X}} = fft(\mathcal{X}, [], 3)$, and the Inverse Fast Fourier Transform (IFFT) is denoted by $\mathcal{X} = ifft(\bar{\mathcal{X}}, [], 3)$. Moreover, for a tensor $\mathcal{X} \in R^{n_1 \times n_2 \times n_3}$, the block vector is defined as $(bvec(\mathcal{X})) = [\mathbf{X}^{(1)}; \mathbf{X}^{(2)}; \ldots; \mathbf{X}^{(n_3)}]$, and the inverse operation as $fold(bvec(\mathcal{X})) = \mathcal{X}$. The block-diagonal matrix and the block circulant matrix are denoted as follows [26]:

$$bdiag(\mathcal{X}) := \begin{bmatrix} \mathbf{X}^{(1)} & & & \\ & \mathbf{X}^{(2)} & & \\ & & \ddots & \\ & & & \mathbf{X}^{(n_3)} \end{bmatrix}$$

$$bcirc(\mathcal{X}) := \begin{bmatrix} \mathbf{X}^{(1)} & \mathbf{X}^{(n_3)} & \cdots & \mathbf{X}^{(2)} \\ \mathbf{X}^{(2)} & \mathbf{X}^{(1)} & \cdots & \mathbf{X}^{(3)} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{X}^{(n_3)} & \mathbf{X}^{(n_3-1)} & \cdots & \mathbf{X}^{(1)} \end{bmatrix}$$

Definition 1 (*Tensor Transpose* [26]). *For a tensor $\mathcal{Z} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, its transpose $\mathcal{Z}^\top \in \mathbb{R}^{n_2 \times n_1 \times n_3}$ is obtained by transposing each frontal slice and reversing the order of the transposed frontal slices from the second to the last slice.*

Definition 2 (*Identity Tensor* [26]). *The identity tensor $\mathcal{I} \in \mathbb{R}^{n \times n \times n_3}$ is a tensor whose first frontal slice is the $n \times n$ identity matrix and whose other frontal slices are zero matrices.*

Definition 3 (*f-Diagonal* [26]). *A tensor is called f-diagonal tensor when all of its frontal slices are diagonal matrices.*

Definition 4 (*t-Product* [26]). *If* $\mathcal{M} \in R^{n_1 \times n_2 \times n_3}$ *, and* $\mathcal{N} \in R^{n_2 \times n_4 \times n_3}$*, the t-product* $\mathcal{M} * \mathcal{N}$ *has a tensor of size* $n_1 \times n_4 \times n_3$

$$\mathcal{M} * \mathcal{N} = fold(bcirc(\mathcal{M})bvec(\mathcal{N})), \tag{1}$$

where bcirc$(\cdot)$ *and* bvec$(\cdot)$ *denote the block circulant operator and block vectorization operator, respectively.*

Definition 5 (*Orthogonal Tensor* [26]). *A tensor* $\mathcal{Q} \in R^{n_1 \times n_2 \times n_3}$ *is orthogonal if*

$$\mathcal{Q} * \mathcal{Q}^\top = \mathcal{Q}^\top * \mathcal{Q} = \mathcal{I}, \tag{2}$$

where $*$ *is the t-product.*

Definition 6 (*Tensor Singular Value Decomposition, t-SVD* [26]). *Given a tensor* $\mathcal{Z} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$*, its t-SVD is defined as*

$$\mathcal{Z} = \mathcal{U} * \mathcal{G} * \mathcal{V}^\top, \tag{3}$$

where $\mathcal{U} \in \mathbb{R}^{n_1 \times n_1 \times n_3}$ *and* $\mathcal{V} \in \mathbb{R}^{n_2 \times n_2 \times n_3}$ *are orthogonal tensors, and* $\mathcal{G} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ *is an f-diagonal tensor.*

2.2. Maximum Correntropy Criterion

In real world computer vision scenarios [27–29], face recognition against noise and outliers is extremely challenging, mainly due to the unpredictability of errors (biases) caused by these noises and outliers. This means that the location of these errors in the image may vary from test sample to test sample, and it is difficult for the computer to predict in advance. Errors can be arbitrarily large and therefore cannot be ignored or treated as small noise. In recent years, the concept of correntropy has been proposed by Information Theoretic Learning (ITL) [30] to deal with non-Gaussian noise [31] and pulse noise [32]. Correntropy is closely related to Renyi's quadratic entropy [33], which selects the optimal model parameters through MCC between two probability distributions. Let x and y be two random variables, then the correlation coefficient $V_\sigma(x, y)$ can be defined as

$$V_\sigma(x, y) = E[k_\sigma(x - y)], \tag{4}$$

where $k_\sigma(\cdot)$ is a shift-invariant Mercer kernel [34], σ is the parameter of the kernel function used to control the bandwidth, and $E[\cdot]$ denotes the expectation. It utilizes the kernel trick of nonlinearly mapping the input space to a high-dimensional feature space. Unlike traditional kernel methods, it works independently on pairwise samples. Therefore, the sample estimator of entropy is given by

$$\hat{V}_\sigma(x, y) = \frac{1}{n} \sum_{i=1}^{n} k_\sigma(x_i - y_i), \tag{5}$$

where $\{x_i, y_i\}_{i=1}^n$ is the sample set. For simplicity, the Gaussian kernel $k_\sigma(x) = g(x, \sigma) \triangleq \exp(-x^2/2\sigma^2)$ is widely used. The main advantage of MCC is that it can effectively deal with non-Gaussian and nonlinear problems and has good robustness.

2.3. Low-Rank Matrix Factorization

Matrix factorization, by decomposing an original matrix into a low-rank matrix, provides a unified framework for dimensionality reduction, clustering, and matrix completion.

Low-rank matrix factorization (LMF) has been widely used in many applications, such as sound signal processing [21] and environmental measurement [35]. The initial

study of LMF was referred to as "Self-modeling curve resolution" [36] or "Non-negative Matrix Factorization" [35]. LMF is also applied to face images to look for partial face [37] decomposition. Given a Low-rank data matrix $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n] \in \mathbb{R}^{d \times n}$, LMF finds the sum of two matrices $\mathbf{W} = [w_{ij}] \in \mathbb{R}^{d \times k}$ and $\mathbf{H} = [h_{ij}] \in \mathbb{R}^{k \times n}$ by minimizing the following objective functions:

$$\begin{aligned} &\min_{\mathbf{W},\mathbf{H}} \ \|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2 \\ &\text{s.t.} \quad \mathbf{W}^\top \mathbf{W} = \mathbf{I}. \end{aligned} \tag{6}$$

LMF can be applied to the statistical analysis of multivariate data in the following ways. Given a set of multivariate N-dimensional data vectors, place these vectors in the columns of the $n \times m$ matrix $\mathbf{X}$, where m is the number of examples in the data set. This matrix is then approximately decomposed into an $n \times r$ matrix $\mathbf{W}$ and an $r \times m$ matrix $\mathbf{H}$, with $r < \min\{m, n\}$.

In semi-supervised multi-view clustering, partial label information can be incorporated through the label constraint matrix $\mathbf{A}$. Specifically, $\mathbf{M}$ is introduced as the latent representation matrix associated with $\mathbf{A}$, and $\mathbf{MA}$ represents the label-constrained representation of all samples. For the v-th view, the reconstruction is formulated as

$$\mathbf{X}^v \approx, \mathbf{W}^v \mathbf{M}\mathbf{A}, \tag{7}$$

where $\mathbf{W}^v$ is the view-specific basis matrix.This formulation enables different views to share a common label-constrained representation while preserving view-specific feature mappings.

2.4. Tensor Schatten p-Norm for Multi-View Graph Learning

In multi-view graph learning, each view constructs an individual similarity graph to characterize the local geometric structure of samples. However, independently optimizing each view-specific graph fails to capture the high-order correlations across views, which degrades clustering performance especially when partial views are corrupted by noise or redundant features [36]. To model cross-view consistency, tensor-based frameworks stack all view-specific graphs into a third-order tensor along the view dimension and impose low-rank constraints. The tensor nuclear norm derived from t-SVD is one of the most widely used convex surrogates for tensor rank. However, it penalizes different singular-value components uniformly, which may lead to a loose approximation of the intrinsic rank structure. Recent studies have therefore explored more flexible nonconvex low-rank regularizers. For example, capped reweighting strategies adaptively distinguish different rank components and provide tighter low-rank approximations with theoretical convergence guarantees [38], while transformed Schatten-type penalties have demonstrated improved capability in recovering intrinsic low-rank subspace structures [39]. More recently, capped tensor norm minimization has further extended this idea to multi-view tensor learning [40]. Motivated by these developments, we employ the tensor Schatten p-norm, which provides a flexible spectral constraint by adjusting the penalization of singular values through the parameter p. Given a third-order tensor $\mathcal{Z} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, let $\overline{\mathcal{Z}}$ denote its Fast Fourier Transform along the third dimension. The tensor Schatten p-norm is formally defined as

$$\|\mathcal{Z}\|_{S_p} = \left(\sum_{k=1}^{n_3} \sum_{i=1}^{\min(n_1,n_2)} \sigma_i^p \left(\overline{\mathcal{Z}}^{(k)} \right) \right)^{1/p}, \qquad \text{s.t.} \quad 0 < p \tag{8}$$

where $\sigma_i(\overline{\mathcal{Z}}^{(k)})$ denotes the i-th singular value of the k-th frontal slice of $\overline{\mathcal{Z}}$ in the Fourier domain.

The parameter p provides a flexible way to control the spectral regularization imposed on the graph tensor. When $p = 1$, the formulation reduces to the tensor nuclear-norm-type regularization. For $0 < p < 1$, a stronger nonconvex sparsification effect is imposed on the singular values, whereas $p > 1$ produces a smoother spectral penalty. Therefore, rather than restricting the model to a particular range of p, we treat p as a tunable spectral regularization parameter. This generalized formulation makes the tensor regularization component flexible to different data characteristics.

3. Method

In this section, we present the proposed Correntropy-guided Matrix Factorization and Tensor Graph Learning (CMTGL) framework. The model is developed progressively by integrating label-constrained low-rank matrix factorization, maximum correntropy-based robust learning, tensor Schatten p-norm regularization, and consensus graph learning into a unified semi-supervised multi-view clustering formulation. We first introduce the label-constrained representation, followed by the tensor graph learning component and the correntropy-guided low-rank matrix factorization model. Finally, these components are integrated into the unified CMTGL objective function.

3.1. Label-Constrained Representation

Let $\mathbf{X}^v = [\mathbf{x}_1^v, \mathbf{x}_2^v, \ldots, \mathbf{x}_n^v] \in \mathbb{R}^{d_v \times n}$ denote the data matrix of the v-th view, where d_v is the feature dimensionality of the v-th view, n is the number of samples, and $v = 1, \ldots, V$. In the semi-supervised setting, only a subset of samples is labeled. To incorporate such partial label information into the representation learning process, we construct label constraint matrix $\mathbf{A}$.

Assume that n_l samples are labeled and the remaining $n_u = n - n_l$ samples are unlabeled. Let c be the number of classes appearing in the labeled samples, and let $\mathbf{C} \in \mathbb{R}^{c \times n_l}$ be the label indicator matrix of the labeled samples, where $C_{ij} = 1$ if the j-th labeled sample belongs to the i-th class and $C_{ij} = 0$ otherwise. The label constraint matrix is defined as

$$\mathbf{A} = \begin{bmatrix} \mathbf{C} & 0 \\ 0 & \mathbf{I}_{n_u} \end{bmatrix} \in \mathbb{R}^{(c+n_u) \times n}, \tag{9}$$

where $\mathbf{I}_{n_u}$ is the identity matrix for the unlabeled samples.

In the proposed model, the latent dimensionality is set equal to the number of classes, $r = c$. Accordingly, $\mathbf{M} \in \mathbb{R}^{c \times (c+n_u)}$, and the label-constrained representation becomes

$$\mathbf{MA} = [\mathbf{Ma}_1, \mathbf{Ma}_2, \ldots, \mathbf{Ma}_n] \in \mathbb{R}^{c \times n}. \tag{10}$$

Each row of $\mathbf{MA}$ therefore corresponds to one class, while each column represents the class-related latent representation of one sample. According to the construction of $\mathbf{A}$, labeled samples belonging to the same class share the same label-constrained component, whereas each unlabeled sample retains an independent representation variable. This construction allows the available label information to guide the representation learning process without assigning ground-truth labels to the unlabeled samples.

3.2. Constrained Tensor S_p-Norm Graph Learning

For each view, we learn a view-specific reconstruction graph $\mathbf{Z}^v \in \mathbb{R}^{n \times n}$. The i-th column of $\mathbf{Z}^v$, denoted by $\mathbf{z}_i^v$, represents the reconstruction coefficients of the i-th sample in the v-th view. A basic graph reconstruction model can be written as

$$\min_{\mathbf{Z}^v} \frac{1}{2} \sum_{v=1}^{V} \|\mathbf{X}^v \mathbf{Z}^v - \mathbf{X}^v\|_F^2. \tag{11}$$

This term preserves the local reconstruction structure of each view. However, learning each $\mathbf{Z}^v$ independently ignores the correlations among different views.

Since different views describe the same set of samples, their graph structures should be correlated. To capture such high-order correlations, we stack all view-specific graphs into a third-order tensor $\mathcal{Z} = \text{cat}(3, \mathbf{Z}^1, \mathbf{Z}^2, \ldots, \mathbf{Z}^V) \in \mathbb{R}^{n \times n \times V}$. Instead of using the conventional tensor nuclear norm or weighted tensor nuclear norm, we impose the tensor Schatten p-norm $\|\mathcal{Z}\|_{S_p}^p$ on the stacked graph tensor. This regularization provides a flexible spectral constraint on the singular values of the multi-view graph tensor, allowing the strength and shape of the spectral penalty to be adjusted through p. In this way, the common low-dimensional structural information shared across different views can be effectively incorporated into graph learning. Therefore, it is more suitable for preserving the intrinsic low-rank structure among multi-view graphs.

In addition to the view-specific graphs, we introduce a consensus graph $\mathbf{S} \in \mathbb{R}^{n \times n}$ to represent the common graph structure shared by all views. The consensus graph is expected to be consistent with each $\mathbf{Z}^v$. Meanwhile, the label-constrained representation $\mathbf{MA}$ should also be preserved through $\mathbf{S}$. Therefore, before introducing MCC, a squared-loss version of the constrained tensor graph learning model can be formulated as

$$\begin{aligned} \min_{\mathbf{M},\mathbf{Z}^v,\mathbf{S}} \; & \frac{1}{2}\sum_{v=1}^{V} \|\mathbf{X}^v\mathbf{Z}^v - \mathbf{X}^v\|_F^2 + \frac{\lambda}{2}\|\mathbf{MAS} - \mathbf{MA}\|_F^2 \\ & + \frac{\gamma}{2}\sum_{v=1}^{V} \|\mathbf{Z}^v - \mathbf{S}\|_F^2 + \rho\|\mathcal{Z}\|_{S_p}^p + \frac{\beta}{2}\|\mathbf{S}\|_1 \\ \text{s.t.} \; & \mathbf{M} \geq 0, \mathbf{S} \geq 0, \; S_{ii} = 0, \; i = 1, \ldots, n. \end{aligned} \tag{12}$$

where the first term learns the view-specific reconstruction graphs. The second term preserves the label-constrained representation on the consensus graph. The third term enforces the consistency between each view-specific graph and the consensus graph. The fourth term imposes the tensor S_p-norm low-rank regularization on the graph tensor $\mathcal{Z}$, and the last term encourages the sparsity of $\mathbf{S}$.

Although (12) provides a natural way to integrate view-specific graph learning, consensus graph learning, and label constraints, the squared loss is sensitive to outliers. To enhance robustness, we replace the squared reconstruction losses with MCC-induced similarity measures. Specifically, the graph reconstruction error, the label-constrained representation error, and the cross-view graph discrepancy are transformed by Gaussian kernels. The resulting robust constrained tensor graph learning model is

$$\begin{aligned} \max_{\mathbf{M},\mathbf{Z}^v,\mathbf{S}} \; & \frac{1}{2}\sum_{v=1}^{V}\sum_{i=1}^{n} \exp\left(-\frac{\|\mathbf{X}^v\mathbf{z}_i^v - \mathbf{x}_i^v\|_2^2}{2\sigma_2^2}\right) + \frac{\lambda}{2}\sum_{i=1}^{n} \exp\left(-\frac{\|\mathbf{MAs}_i - \mathbf{Ma}_i\|_2^2}{2\sigma_3^2}\right) \\ & + \frac{\gamma}{2}\sum_{v=1}^{V} \exp\left(-\frac{\|\mathbf{Z}^v - \mathbf{S}\|_F^2}{2\sigma_4^2}\right) - \rho\|\mathcal{Z}\|_{S_p}^p - \frac{\beta}{2}\|\mathbf{S}\|_1 \\ \text{s.t.} \; & \mathbf{M} \geq 0, \; \mathbf{S} \geq 0, \; s_{ii} = 0, \; i = 1, \ldots, n. \end{aligned} \tag{13}$$

where $\mathbf{s}_i$ is the i-th column of $\mathbf{S}$, and σ_2, σ_3, and σ_4 are the kernel bandwidths corresponding to the three MCC terms. The parameters λ and γ balance the label-constrained graph term and the graph consistency term, respectively. The parameter ρ controls the tensor S_p-norm regularization, and β controls the sparsity of the consensus graph.

For notational simplicity, we define the MCC-based constrained graph learning component as

$$\begin{aligned}\mathcal{J}_{\mathrm{CTL}} = &\frac{1}{2}\sum_{v=1}^{V}\sum_{i=1}^{n}\exp\left(-\frac{\|\mathbf{X}^v\mathbf{z}_i^v - \mathbf{x}_i^v\|_2^2}{2\sigma_2^2}\right) + \frac{\lambda}{2}\sum_{i=1}^{n}\exp\left(-\frac{\|\mathbf{MAs}_i - \mathbf{Ma}_i\|_2^2}{2\sigma_3^2}\right)\\ &+ \frac{\gamma}{2}\sum_{v=1}^{V}\exp\left(-\frac{\|\mathbf{Z}^v - \mathbf{S}\|_F^2}{2\sigma_4^2}\right) - \frac{\beta}{2}\|\mathbf{S}\|_1.\end{aligned} \tag{14}$$

Then (13) can be compactly rewritten as

$$\begin{aligned}&\max_{\mathbf{M},\mathbf{Z}^v,\mathbf{S}} \ \mathcal{J}_{\mathrm{CTL}}(\mathbf{M},\mathbf{Z}^v,\mathbf{S}) - \rho\|\mathcal{Z}\|_{S_p}^p\\ &\text{s.t. } \mathbf{M}\geq 0,\ \mathbf{S}\geq 0,\ s_{ii}=0,\ i=1,\ldots,n.\end{aligned} \tag{15}$$

3.3. Correntropy-Based Multi-View Low-Rank Matrix Factorization

Low-rank matrix factorization provides an effective way to learn compact representations from high-dimensional data. For the v-th view, let $\mathbf{W}^v \in \mathbb{R}^{d_v\times r}$ be the view-specific basis matrix. With the label-constrained representation $\mathbf{MA}$, the reconstruction of $\mathbf{X}^v$ is given by

$$\mathbf{X}^v \approx \mathbf{W}^v\mathbf{MA}. \tag{16}$$

A conventional squared-loss multi-view low-rank matrix factorization model can be formulated as

$$\begin{aligned}&\min_{\mathbf{W}^v,\mathbf{M}} \ \frac{1}{2}\sum_{v=1}^{V}\|\mathbf{X}^v - \mathbf{W}^v\mathbf{MA}\|_F^2\\ &\text{s.t. } (\mathbf{W}^v)^\top\mathbf{W}^v = \mathbf{I},\ \mathbf{M}\geq 0,\ v=1,\ldots,V.\end{aligned} \tag{17}$$

The orthogonality constraint on $\mathbf{W}^v$ avoids the arbitrary scaling ambiguity between $\mathbf{W}^v$ and $\mathbf{M}$, while the nonnegative constraint on $\mathbf{M}$ improves the interpretability of the learned representation.

However, the squared loss in (17) is also sensitive to large reconstruction errors caused by noise or outliers. Therefore, MCC is adopted to construct a robust multi-view low-rank matrix factorization model:

$$\begin{aligned}&\max_{\mathbf{W}^v,\mathbf{M}} \ \frac{1}{2}\sum_{v=1}^{V}\sum_{i=1}^{n}\exp\left(-\frac{\|\mathbf{x}_i^v - \mathbf{W}^v\mathbf{Ma}_i\|_2^2}{2\sigma_1^2}\right)\\ &\text{s.t. } (\mathbf{W}^v)^\top\mathbf{W}^v = \mathbf{I},\ \mathbf{M}\geq 0,\ v=1,\ldots,V,\end{aligned} \tag{18}$$

where σ_1 is the kernel bandwidth for the matrix factorization error. Compared with the squared-loss formulation, the MCC-based formulation assigns smaller influence to samples with large reconstruction errors and thus enhances the robustness of the learned latent representation.

3.4. Unified Objective Function

The correntropy-based low-rank matrix factorization model in (18) mainly captures the global latent representation structure of multi-view data. The constrained tensor graph learning model in (15) further captures the local geometric structure, the label-constrained

graph structure, and the high-order correlations among different view-specific graphs. By integrating these two components, we obtain the final objective function:

$$\begin{aligned}&\max_{\mathbf{W}^v,\mathbf{M},\mathbf{Z}^v,\mathbf{S}} \frac{1}{2}\sum_{v=1}^{V}\sum_{i=1}^{n}\exp\left(-\frac{\|\mathbf{x}_i^v-\mathbf{W}^v\mathbf{M}\mathbf{a}_i\|_2^2}{2\sigma_1^2}\right)+\alpha\mathcal{J}_{\mathrm{CTL}}(\mathbf{M},\mathbf{Z}^v,\mathbf{S})-\rho\|\mathcal{Z}\|_{S_p}^p \\ &\text{s.t. } (\mathbf{W}^v)^\top\mathbf{W}^v=\mathbf{I},\ \mathbf{M}\geq 0,\mathbf{S}\geq 0,\ s_{ii}=0,\ i=1,\ldots,n.\end{aligned} \tag{19}$$

where α is a trade-off parameter that balances the matrix factorization term and the constrained graph learning component.

The proposed formulation has three main advantages. First, the label constraint matrix explicitly embeds partial label information into the latent representation. Second, MCC reduces the influence of noisy samples and outliers by suppressing large reconstruction errors. Third, the tensor Schatten p-norm regularization provides a flexible spectral constraint on the multi-view graph tensor, while the consensus graph captures the common structural information shared across different views.

4. Optimization

This section presents the optimization procedure for the proposed CMTGL model. Since the objective function in (19) is nonconvex and involves several coupled variable blocks, direct optimization is difficult. We first employ the Fenchel conjugate (FC) [41] transformation to reformulate the correntropy terms into weighted quadratic forms. Based on the reformulated problem, an accelerated block coordinate update (BCU) [42] scheme is developed to update the variables alternately. Within each iteration, one variable block is updated while the remaining variables are fixed at their latest values. An extrapolation strategy is further introduced to accelerate the optimization process.

4.1. Fenchel Conjugate Reformulation

Let $\varphi(z)=z-z\ln(-z)$, according to FC, it is easy to verify that

$$\exp(-x)=\sup_z\{zx-\varphi(z)\}. \tag{20}$$

By setting $(\frac{\partial}{\partial(z)})(zx-\varphi(z))=x+\ln(-z)=0$, one can find that the maximum value is reached at $z=-\exp(-x)$, have

$$O_1^{MCC}=\frac{1}{2}\sum_{v=1}^{V}\sum_{i=1}^{n}(p_i^v\frac{||\mathbf{x}_i^v-\mathbf{W}^v\mathbf{M}\mathbf{a}_i||_2^2}{2\sigma_1^2}-\varphi(p_i^v)), \tag{21a}$$

$$O_2^{MCC}=\frac{1}{2}\sum_{v=1}^{V}\sum_{i=1}^{n}(b_i^v\frac{||\mathbf{X}^v\mathbf{z}_i^v-\mathbf{x}_i^v||_2^2}{2\sigma_2^2}-\varphi(b_i^v)), \tag{21b}$$

$$O_3^{MCC}=\frac{1}{2}\sum_{i=1}^{n}(c_i\frac{||\mathbf{M}\mathbf{A}\mathbf{s}_i-\mathbf{M}\mathbf{a}_i||_2^2}{2\sigma_3^2}-\varphi(c_i)), \tag{21c}$$

$$O_4^{MCC}=\frac{1}{2}\sum_{v=1}^{V}(d^v\frac{||\mathbf{Z}^v-\mathbf{S}||_F^2}{2\sigma_4^2}-\varphi(d^v)), \tag{21d}$$

$$\begin{aligned}f(\mathbf{W}^v,\mathbf{M},\mathbf{Z}^v,\mathbf{S},\mathbf{p}^v,\mathbf{b}^v,\mathbf{c},\mathbf{d}^v)&=O_1^{MCC}(\mathbf{W}^v,\mathbf{M},\mathbf{p}^v)+\alpha(O_2^{MCC}(\mathbf{Z}^v,\mathbf{b}^v)\\&+\lambda O_3^{MCC}(\mathbf{M},\mathbf{S},\mathbf{c})+\gamma O_4^{MCC}(\mathbf{Z}^v,\mathbf{S},\mathbf{d}^v)).\end{aligned} \tag{21e}$$

According to the definitions in (21a)–(21e) and (20), the problem (19) in is equivalent to

$$\begin{aligned}&\max\ f(\mathbf{W}^v,\mathbf{M},\mathbf{Z}^v,\mathbf{S},\mathbf{p}^v,\mathbf{b}^v,\mathbf{c},\mathbf{d}^v)-\rho\|\mathcal{Z}\|_{S_p}^p-\frac{\alpha\beta}{2}\|\mathbf{S}\|_1\\&\text{s.t. } (\mathbf{W}^v)^\top\mathbf{W}^v=\mathbf{I},\ \mathbf{M}\geq 0,\mathbf{S}\geq 0,\ s_{i,i}=0,\ \forall i=1,\ldots,n,\end{aligned} \tag{22}$$

where $\mathbf{p}^v = (p_1^v, p_2^v, \ldots, p_n^v)$, $\mathbf{b}^v = (b_1^v, b_2^v, \ldots, b_n^v)$, $\mathbf{c} = (c_1, c_2, \ldots, c_n)$ and $\mathbf{d} = (d^1, d^2, \ldots, d^V)$.

4.2. Iterative Updates by Accelerated BCU

By accelerating BCU, one variable in (22) is updated in each iteration of the algorithm, while the other variables are fixed with their latest values. Specifically, these variables are updated as follows:

$$\{\mathbf{p}^v\}^{t+1} = \underset{\mathbf{p}^v}{\operatorname{argmax}} O_1^{MCC}(\{\mathbf{p}^v\}^t) \tag{23a}$$

$$\{\mathbf{b}^v\}^{t+1} = \underset{\mathbf{b}^v}{\operatorname{argmax}} O_2^{MCC}(\{\mathbf{b}^v\}^t) \tag{23b}$$

$$\mathbf{c}^{t+1} = \underset{\mathbf{c}}{\operatorname{argmax}} O_3^{MCC}(\mathbf{c}^t) \tag{23c}$$

$$\{\mathbf{d}^v\}^{t+1} = \underset{\mathbf{d}^v}{\operatorname{argmax}} O_4^{MCC}(\{\mathbf{d}^v\}^t) \tag{23d}$$

$$\{\mathbf{Z}^v\}^{t+1} = \underset{\mathbf{Z}^v}{\operatorname{argmax}} \left\langle \nabla_{\mathbf{Z}^v} f(\{\hat{\mathbf{Z}}^v\}^t),\ \mathbf{Z}^v - \{\hat{\mathbf{Z}}^v\}^t \right\rangle - \frac{L_{\mathbf{Z}}^{v,t}}{2} ||\mathbf{Z}^v - \{\hat{\mathbf{Z}}^v\}^t||_F^2 - \rho \|\mathcal{Z}\|_{S_p}^p \tag{23e}$$

$$\mathbf{M}^{t+1} = \underset{\mathbf{M} \geq 0}{\operatorname{argmax}} \langle \nabla_{\mathbf{M}} f(\hat{\mathbf{M}}^t), \mathbf{M} - \hat{\mathbf{M}}^t \rangle - \frac{L_{\mathbf{M}}^t}{2} ||\mathbf{M} - \hat{\mathbf{M}}^t||_F^2 \tag{23f}$$

$$\mathbf{S}^{t+1} = \underset{\mathbf{S} \geq 0,\ s_{i,i}=0}{\operatorname{argmax}} \langle \nabla_{\mathbf{S}} f(\hat{\mathbf{S}}^t), \mathbf{S} - \hat{\mathbf{S}}^t \rangle - \frac{L_{\mathbf{S}}^t}{2} ||\mathbf{S} - \hat{\mathbf{S}}^t||_F^2 - \frac{\alpha\beta}{2} \|\mathbf{S}\|_1 \tag{23g}$$

$$\{\mathbf{W}^v\}^{t+1} = \underset{\mathbf{W}^{v\top}\mathbf{W}^v = \mathbf{I}}{\operatorname{argmax}} f(\{\mathbf{W}^v\}^t) \tag{23h}$$

where $\{\hat{\mathbf{Z}}^v\}^t = \{\mathbf{Z}^v\}^t + \omega_{\mathbf{Z}^v}^t(\{\mathbf{Z}^v\}^t - \{\mathbf{Z}^v\}^{t-1})$, $\hat{\mathbf{M}}^t = \mathbf{M}^t + \omega_{\mathbf{M}}^t(\mathbf{M}^t - \mathbf{M}^{t-1})$, $\hat{\mathbf{S}}^t = \mathbf{S}^t + \omega_{\mathbf{S}}^t(\mathbf{S}^t - \mathbf{S}^{t-1})$. Here, the partial gradients are $\nabla_{\mathbf{Z}^v} f(\cdot)$, $\nabla_{\mathbf{M}} f(\cdot)$ and $\nabla_{\mathbf{S}} f(\cdot)$ for $\mathbf{Z}^v$, $\mathbf{M}$ and $\mathbf{S}$, respectively. $L_{\mathbf{Z}^v}^t$, $L_{\mathbf{M}}^t$ and $L_{\mathbf{S}}^t$ are the Lipschitz constants of $\nabla_{\mathbf{Z}^v} f(\cdot)$, $\nabla_{\mathbf{M}} f(\cdot)$ and $\nabla_{\mathbf{S}} f(\cdot)$ with respect to $\mathbf{Z}^v$, $\mathbf{M}$ and $\mathbf{S}$ (see Appendix A for the detailed expressions of gradients and Lipschitz constants). For acceleration [42], the weight is the extrapolation point of $\omega_{\mathbf{Z}^v}^t$, $\omega_{\mathbf{M}}^t$ and $\omega_{\mathbf{S}}^t \in [0,1)$. In the following, we will discuss how to solve each subproblem from (23a)–(23f) explicitly.

4.2.1. Update $\mathbf{p}^v$, $\mathbf{b}^v$, $\mathbf{c}$, and $\mathbf{d}^v$

The optimal solution of can be obtained when $z = -\exp(-x)$. Therefore the solutions of (23a)–(23c), for any $i = 1, \ldots, n$, can be obtained explicitly as

$$\{p_i^v\}^{t+1} = -\exp\left(-\frac{\|\mathbf{x}_i^v - \{\mathbf{W}^v\}^t \mathbf{M}^t \mathbf{a}_i\|_2^2}{2\sigma_1^2}\right), \tag{24a}$$

$$\{b_i^v\}^{t+1} = -\exp\left(-\frac{\|\mathbf{X}^v \{\mathbf{z}_i^v\}^t - \mathbf{x}_i^v\|_2^2}{2\sigma_2^2}\right), \tag{24b}$$

$$c_i^{t+1} = -\exp\left(-\frac{\|\mathbf{M}\mathbf{A}\mathbf{s}_i^t - \mathbf{M}\mathbf{a}_i\|_2^2}{2\sigma_3^2}\right), \tag{24c}$$

$$\{d^v\}^{t+1} = -\exp\left(-\frac{\|\{\mathbf{Z}^v\}^t - \mathbf{S}^t\|_F^2}{2\sigma_4^2}\right). \tag{24d}$$

4.2.2. Update $\mathcal{Z}$

Removing the terms that are irrelevant to $\mathcal{Z}$, the subproblem of updating the view-specific graph tensor in (23e) can be written as

$$\min_{\mathcal{Z}} \frac{1}{2} \|\mathcal{Z} - \mathcal{Q}\|_F^2 + \rho \|\mathcal{Z}\|_{S_p}^p, \tag{25}$$

where $\mathcal{Z} = \text{cat}(3, \mathbf{Z}^1, \mathbf{Z}^2, \ldots, \mathbf{Z}^V) \in \mathbb{R}^{n\times n\times V}$. Each frontal slice of $\mathcal{Q}$ is constructed by $\mathbf{Q}^v = \{\hat{\mathbf{Z}^v}\}^t + \frac{1}{L^t_{\mathbf{Z}^v}}\nabla_{\mathbf{Z}^v} f(\{\hat{\mathbf{Z}^v}\}^t)$ for $v = 1, \ldots, V$, and $\mathcal{Q} = \text{cat}(3, \mathbf{Q}^1, \mathbf{Q}^2, \ldots, \mathbf{Q}^V)$.

Problem (25) is a generalized tensor optimization problem regularized by the tensor Schatten p-norm. Instead of re-deriving the slice-by-slice optimization, we directly apply the Generalized Tensor Optimization Method [43], which efficiently solves this objective via the t-SVD framework. According to Theorem 4 in [43], the optimal solution to (25) can be obtained in the Fourier domain as

$$\mathcal{Z}^* = \Gamma_{\rho,p}(\mathcal{Q}) = \mathcal{U} * \text{ifft}(\mathcal{P}_{\rho,p}(\bar{\mathcal{Q}})) * \mathcal{V}^\top, \tag{26}$$

where $\bar{\mathcal{Q}} = \text{fft}(\mathcal{Q}, [], 3)$ is the tensor in the Fourier domain, and its t-SVD is $\bar{\mathcal{Q}} = \mathcal{U} * \mathcal{S} * \mathcal{V}^\top$.

The operator $\mathcal{P}_{\rho,p}(\cdot)$ applies the Generalized Soft-Thresholding (GST) algorithm [43] to each singular value σ_{ij} of the frontal slices of $\bar{\mathcal{Q}}$. Specifically, each updated singular value $\hat{\sigma}_{ij}$ corresponds to the analytical solution of the scalar L_p-minimization problem:

$$\hat{\sigma}_{ij} = \arg\min_{\xi\geq 0} \frac{1}{2}(\xi - \sigma_{ij})^2 + \rho\xi^p. \tag{27}$$

Based on the GST operator outlined in [43], $\hat{\sigma}_{ij}$ is efficiently computed using a precise thresholding condition:

$$\hat{\sigma}_{ij} = \mathcal{T}_{p,\rho}(\sigma_{ij}) = \begin{cases} 0, & \text{if } \sigma_{ij} \leq \tau_p^{GST}(\rho) \\ \delta^*_{ij}, & \text{otherwise} \end{cases} \tag{28}$$

where the threshold is defined as $\tau_p^{GST}(\rho) = (2\rho(1-p))^{\frac{1}{2-p}} + \rho p(2\rho(1-p))^{\frac{p-1}{2-p}}$, and δ^*_{ij} is the exact non-negative root satisfying the fixed-point equation $\delta^*_{ij} - \sigma_{ij} + \rho p(\delta^*_{ij})^{p-1} = 0$.

The scalar problem above is solved according to the value of p. For $0 < p < 1$, the Generalized Soft-Thresholding (GST) operator in [43] can be applied. When $p = 1$, the solution reduces to the conventional soft-thresholding operator:

$$\hat{\sigma}_{ij} = \max(\sigma_{ij} - \rho, 0). \tag{29}$$

For $p > 1$, the objective is convex with respect to ξ, and the unique non-negative solution satisfies

$$\hat{\sigma}_{ij} - \sigma_{ij} + \rho p\hat{\sigma}_{ij}^{p-1} = 0. \tag{30}$$

Therefore, the tensor update can be implemented using the corresponding p-dependent singular-value proximal mapping.

By applying this generalized optimization theorem, we circumvent the redundant formulation and obtain the optimal graph tensor directly. The specific tensor update procedure, incorporating dimensional rotation and conjugate symmetry for computational efficiency, is summarized in Algorithm 1.

4.2.3. Update $\mathbf{M}$

Removing items that have no relation to $\mathbf{M}$, the subproblem in (23f) for updating $\mathbf{M}$ is equivalent to

$$\min_{\mathbf{M}\geq 0} \frac{1}{2}\|\mathbf{M} - \mathbf{M}^*\|_F^2, \tag{31}$$

where $\mathbf{M}^* = \hat{\mathbf{M}}^t + \frac{1}{L^t_{\mathbf{M}}}\nabla_{\mathbf{M}} f(\cdot)$. The optimal solution of (31) is $\mathbf{M}^{t+1} = \max(0, \mathbf{M}^*)$.

Algorithm 1 Solution to (25)

Input: Tensor $\mathcal{Q} \in \mathbb{R}^{n\times n\times V}$, regularization parameter ρ, Schatten parameter p;
Output: Tensor $\mathcal{Z} \in \mathbb{R}^{n\times n\times V}$;

Rotate the tensor: $\mathcal{B} = \text{shiftdim}(\mathcal{Q}, 1)$;
Compute FFT: $\bar{\mathcal{B}} = \text{fft}(\mathcal{B}, [], 3)$;
for $j = 1, \ldots, \lceil \frac{n+1}{2} \rceil$ **do**
  Compute compact singular value decomposition. $\bar{\mathcal{B}}(:,:,j) = \mathbf{U}_j \mathbf{\Sigma}_j \mathbf{V}_j^\top$;
  Update each singular value using the corresponding p-dependent proximal operator;
  Form diagonal matrix: $\hat{\mathbf{\Sigma}}_j = \text{diag}(\hat{\sigma}_{1j}, \hat{\sigma}_{2j}, \ldots)$;
  Reconstruct slice: $\bar{\mathcal{Y}}(:,:,j) = \mathbf{U}_j \hat{\mathbf{\Sigma}}_j \mathbf{V}_j^\top$;
  if $j > 1$ and $j \neq n - j + 2$ **then**
    Recover the conjugate slice: $\bar{\mathcal{Y}}(:,:,n-j+2) = \text{conj}(\bar{\mathcal{Y}}(:,:,j))$;
  end if
end for
Compute IFFT: $\mathcal{Y} = \text{ifft}(\bar{\mathcal{Y}}, [], 3)$;
Recover the graph tensor: $\mathcal{Z} = \text{shiftdim}(\mathcal{Y}, 2)$;

4.2.4. Update $\mathbf{S}$

Removing items that have no relation to $\mathbf{S}$, the subproblem in (23g) for updating $\mathbf{S}$ is equivalent to

$$\min_{\mathbf{S}\geq 0,\, s_{i,i}=0} \frac{1}{2}\|\mathbf{S} - \mathbf{S}^*\|_F^2 + \eta\|\mathbf{S}\|_1, \tag{32}$$

where $\mathbf{S}^* = \hat{\mathbf{S}}^t + \frac{1}{L_{\mathbf{S}}^t}\nabla_{\mathbf{S}} f(\cdot)$ and $\eta = (\alpha\beta/2L_{\mathbf{S}}^t) > 0$. Due to the constraint that $s_{i,i} = 0$, the problem in (32) is equivalent to

$$\min_{\mathbf{S}\geq 0} \frac{1}{2}\|\mathbf{S} - \tilde{\mathbf{S}}^*\|_F^2 + \eta\|\mathbf{S}\|_1, \tag{33}$$

where $\tilde{\mathbf{S}}^*$ equals $\mathbf{S}^*$ except for the diagonal elements, and $\tilde{s}_{i,i}^* = 0, \forall i = 1, \ldots, n$. The optimization problem in (33) has the closed-form solution as $\mathbf{S} = \max(\tilde{\mathbf{S}}^* - \eta, 0)$.

4.2.5. Update $\mathbf{W}^v$

Removing items that have no relation to $\mathbf{W}^v$, the subproblem in (23h) is equivalent to

$$\max_{\mathbf{W}^{v\top}\mathbf{W}^v = \mathbf{I}} \frac{1}{2}\sum_{v=1}^{V} \text{Tr}(\mathbf{X}^v - \mathbf{W}^v\mathbf{M}\mathbf{A})\mathbf{P}^v(\mathbf{X}^v - \mathbf{W}^v\mathbf{M}\mathbf{A})^\top, \tag{34}$$

where $\mathbf{P}^v$ is a diagonal matrix with $\mathbf{p}^v/2\sigma_1^2$ as its diagonal element. Here, $\mathbf{W}$ in each view can be optimized independently. The optimal solution can be easily obtained by Theorem 1 in the following:

Theorem 1. *The problem in (34) has an optimal solution* $\mathbf{W}^v = -\mathbf{V}\mathbf{G}^T$, *where* $\mathbf{V}$ *and* $\mathbf{G}$ *are obtained by compact singular value decomposition (SVD) with* $\mathbf{M}\mathbf{A}\mathbf{P}^v\mathbf{X}^{v\top} = \mathbf{G}\mathbf{D}\mathbf{V}^T$.

4.3. Parameters Setting

It should be noted that the updating of the variables $\mathcal{Z}$ and $\mathbf{M}$ in (23) requires the value of the Lipschitz constant $L_{\mathbf{Z}^v}^t$, $L_{\mathbf{M}}^t$ and the extrapolation weights $\omega_{\mathbf{Z}^v}^t, \omega_{\mathbf{M}}^t$. In the proposed

algorithm, the value of the Lipschitz constants are set to $L^t_{\mathbf{Z}^v} = \max(\tilde{L}^t_{\mathbf{Z}^v}, L_{\mathbf{Z}^v}), L^t_{\mathbf{M}} = \max(\tilde{L}^t_{\mathbf{M}}, L_{\mathbf{M}})$, where $L_{\mathbf{Z}^v} = \lambda\|\mathbf{X}^{v\top}\mathbf{X}^v\|_F, L_{\mathbf{M}} = \sum_{v=1}^{V}\|\mathbf{A}\|_F^2$ and

$$\begin{aligned}\widetilde{L}^{v,t}_{\mathbf{Z}} &= \alpha(\|\mathbf{X}^{v\top}\mathbf{X}^v\|_F\|\mathbf{B}^{v,t+1}\|_2 + \gamma)\\ \widetilde{L}^t_{\mathbf{M}} &= \|\mathbf{A}\|_F^2(\alpha\lambda\|\mathbf{C}\|_2\|\mathbf{I}-\mathbf{S}\|_F^2 + \sum_{v=1}^{V}\|\mathbf{P}^v\|_2)\\ \widetilde{L}^t_{\mathbf{S}} &= \alpha(\lambda\|\mathbf{A}^\top\mathbf{M}^\top\mathbf{M}\mathbf{A}\|_F\|\mathbf{C}\|_2 + \gamma\|\mathbf{d}\|_1)\end{aligned} \tag{35}$$

In the above, $\mathbf{B}^v$, $\mathbf{C}$ are diagonal matrices with $\mathbf{b}/2\sigma_2^2, \mathbf{c}/2\sigma_3^2$ as the diagonal elements respectively. The extrapolation weights are set according to [42] as follows:

$$\omega^t_{\mathbf{Z}^v} = \min(\hat{\omega}^t, \delta_\omega\sqrt{\frac{L^{t-1}_{\mathbf{Z}^v}}{L^t_{\mathbf{Z}^v}}}) \tag{36a}$$

$$\omega^t_{\mathbf{M}} = \min(\hat{\omega}^t, \delta_\omega\sqrt{\frac{L^{t-1}_{\mathbf{M}}}{L^t_{\mathbf{M}}}}) \tag{36b}$$

$$\omega^t_{\mathbf{S}} = \min(\hat{\omega}^t, \delta_\omega\sqrt{\frac{L^{t-1}_{\mathbf{S}}}{L^t_{\mathbf{S}}}}) \tag{36c}$$

where $\delta_\omega < 1$ is predetermined and $\hat{\omega}^t = (\tau_{t-1} - 1)/\tau_t$ with

$$\tau_0 = 1, \tau_t = \frac{1}{2}\left(1 + \sqrt{1 + 4\tau^2_{t-1}}\right). \tag{37}$$

Combining the updated rules above , the complete optimization procedure of CMTGL is summarized in Algorithm 2. As shown in Lines 11–13, if an extrapolated update decreases the objective value, the extrapolated variables are discarded and the corresponding blocks are recomputed from the previous iterates. Therefore, only non-decreasing objective updates are accepted, which provides the monotonicity required in the subsequent convergence analysis.

Algorithm 2 CMTGL for Multi-View Clustering

Require: Data matrix $\mathbf{X}^v \in \mathbb{R}^{d^v \times n}$, constraint matrix $\mathbf{A}$, parameters $\alpha, \beta, \lambda, \gamma, \rho$, and p.
Ensure: Clustering indicator (representation) matrix $\mathbf{M}$.

1: Initialize $\mathbf{W}^{v,0}$, $\mathbf{M}^0$, $\mathbf{Z}^{v,0}$ and $\mathbf{S}^0$, choose a positive number $\delta_\omega < 1$, and set $t = 0$.
2: **while** Not convergent **do**
3: Update $\{\mathbf{p}^v\}^{t+1}, \{\mathbf{b}^v\}^{t+1}, \mathbf{c}^{t+1}, \{\mathbf{d}^v\}^{t+1} \leftarrow$(24).
4: Compute $L^t_{\mathbf{Z}^v}$, $L^t_{\mathbf{M}}$, $L^t_{\mathbf{S}}$ and $\omega^t_{\mathbf{Z}^v}$, $\omega^t_{\mathbf{M}}$, $\omega^t_{\mathbf{S}}$ according to (35) and (36), respectively.
5: Let $\{\hat{\mathbf{Z}^v}\}^t = \{\mathbf{Z}^v\}^t + \omega^t_{\mathbf{Z}^v}(\{\mathbf{Z}^v\}^t - \{\mathbf{Z}^v\}^{t-1})$.
6: Let $\hat{\mathbf{M}}^t = \mathbf{M}^t + \omega^t_{\mathbf{M}}(\mathbf{M}^t - \mathbf{M}^{t-1})$.
7: Let $\hat{\mathbf{S}}^t = \mathbf{S}^t + \omega^t_{\mathbf{S}}(\mathbf{S}^t - \mathbf{S}^{t-1})$.
8: Update $\mathcal{Z}^{t+1}$ by solving (25) with Algorithm 1.
9: Update $\mathbf{M}^{t+1}$ by (31).
10: Update $\mathbf{S}^{t+1}$ by (33).
11: **if** $f(\{\mathbf{W}^v\}^t, \mathbf{M}^{t+1}, \{\mathbf{Z}^v\}^{t+1}, \{\mathbf{p}^v\}^{t+1}, \{\mathbf{b}^v\}^{t+1}, \mathbf{c}^{t+1}) - \rho\|\mathcal{Z}^{t+1}\|^p_{S_p} - \frac{\alpha\beta}{2}\|\mathbf{S}^{t+1}\|_1 \leq f(\{\mathbf{W}^v\}^t, \mathbf{M}^t, \{\mathbf{Z}^v\}^t, \{\mathbf{p}^v\}^t, \{\mathbf{b}^v\}^t, \mathbf{c}^{t+1}) - \rho\|\mathcal{Z}^t\|^p_{S_p} - \frac{\alpha\beta}{2}\|\mathbf{S}^t\|_1$ **then**
12: Set $\hat{\mathcal{Z}}^t = \mathcal{Z}^t, \hat{\mathbf{M}}^t = \mathbf{M}^t$, and back to **Step** 8.
13: **end if**
14: Compute SVD of $\mathbf{MAPX}^{vT} = \mathbf{GDV}^T$.
15: Update $\{\mathbf{W}^v\}^{t+1} = -\mathbf{VG}^T$
16: Let $t \leftarrow t + 1$.
17: **end while**

4.4. Convergence Analysis

For the following objective-sequence analysis, the kernel bandwidths are assumed to remain positive and are treated as fixed quantities within each block-update cycle. Since the proposed CMTGL model is nonconvex and contains several coupled variable blocks, establishing the convergence of the entire iterate sequence is nontrivial. In this work, we focus on the convergence of the objective-value sequence. Similar to the objective-sequence convergence analysis of the correntropy-based low-rank matrix factorization method in [9], the key observation is that the accepted updates of Algorithm 2 preserve the monotonicity of the objective function, while the MCC-based objective is upper bounded.

Let $\mathcal{J}^t$ denote the value of the objective function in (19) at the t-th iteration. The auxiliary variables introduced by the Fenchel conjugate transformation are updated to their optimal values, while the variables $\mathbf{Z}^v$, $\mathbf{M}$, $\mathbf{S}$, and $\mathbf{W}^v$ are updated according to their corresponding subproblems. Moreover, the restart strategy in Algorithm 2 rejects the extrapolated variables whenever they lead to a decrease in the objective value. Therefore, the accepted objective sequence satisfies $\mathcal{J}^{t+1} \geq \mathcal{J}^t$. We next show that the objective function is upper bounded. For any $x \geq 0$ and $\sigma > 0$, $0 < \exp(-x/2\sigma^2) \leq 1$. Hence, the MCC-based matrix factorization term in (19) satisfies

$$\frac{1}{2}\sum_{v=1}^{V}\sum_{i=1}^{n}\exp\left(-\frac{\|\mathbf{x}_i^v - \mathbf{W}^v\mathbf{M}\mathbf{a}_i\|_2^2}{2\sigma_1^2}\right) \leq \frac{Vn}{2}. \tag{38}$$

Similarly, the three MCC terms in the constrained tensor graph learning component are respectively upper bounded by $Vn/2, \lambda n/2, \gamma V/2$. Moreover, $\rho\|\mathcal{Z}\|_{S_p}^p \geq 0, \alpha\beta/2\|\mathbf{S}\|_1 \geq 0$. Therefore, the objective function in (19) satisfies $\mathcal{J}^t \leq 1/2[Vn + \alpha(Vn + \lambda n + \gamma V)]$. Consequently, $\mathcal{J}^t$ is a monotonically non-decreasing sequence with a finite upper bound. Therefore, there exists a finite value $\mathcal{J}^*$ such that $\lim_{t\to\infty}\mathcal{J}^t = \mathcal{J}^*$. Thus, the objective-value sequence generated by Algorithm 2 is convergent. It should be emphasized that the above result establishes the convergence of the objective values only and does not imply that the generated iterate sequence necessarily converges to a local or global optimum.

4.5. Computational Complexity

In general, we analyze the computational complexity of Algorithm 2 without considering the particular structure of the data matrix $\mathbf{X}^v$. If the $\mathbf{X}^v$ is sparse, the computational complexity can be even lower. We assume that clusters $k \ll \min(d^v, n)$ and $\mathbf{Z}^v$ are sparse, with the number of non-zero elements in each column $l \ll n$. The main cost of Algorithm 2 is the updating of $\mathbf{p}^v$, $\mathbf{b}^v$ and $\mathbf{c}$, $\mathbf{W}^v$, $\mathbf{M}$, and $\mathbf{Z}^v$. The costs of updating $\mathbf{p}^v$, $\mathbf{b}^v$ and $\mathbf{c}$ by (24) are $O(Vndk)$, $O(Vndl)$, and $O(Vnkl)$, respectively. For the update of $\mathbf{M}$, the main cost is the calculation of $\nabla_{\mathbf{M}}f$. The total calculated amount of $\nabla_{\mathbf{M}}f$ in Appendix A is taken as $O(Vnk^2 + Vndk + Vnkl)$. Updating $\mathcal{Z}$ needs to calculate the FFT and IFFT, which takes $O(Vn^2log(n))$. In the Fourier domain, the SVD of rotated $\mathcal{Z}$ with a size of $n \times V \times n$ needs to be computed, and it takes $O(V^2n^2)$. So we need $O(Vn^2log(n) + V^2n^2)$ in total when updating $\mathcal{Z}$. For the $\mathbf{W}^v$ update, the calculation is $O(Vndk)$. Therefore, the computational complexity of each iteration is $O(Vn^2log(n) + V^2n^2)$; because of $k, l \ll \min\{n, d^v\}$.

5. Experimental Results and Analysis

In this section, we evaluate the proposed CMTGL framework on six public multi-view datasets. CMTGL is compared with nine representative multi-view clustering and semi-supervised multi-view learning methods. The experiments include clustering performance comparison, robustness analysis, ablation study, parameter sensitivity analysis, view-weight analysis, and empirical convergence evaluation.

5.1. Data Sets

Six data sets were used in the experimental study, and the statistical results of these data sets are shown in Table 2.

Table 2. The datasets utilized in the experiments.

Dataset	Views	Classes	Sample Size	d1	d2	d3	d4	d5	d6
COIL20	3	20	1440	1024	3304	6750			
MSRCv1	5	7	210	576	512	256	254	24	
NUS-2400	6	12	2400	64	144	73	128	225	500
ORL	4	40	400	512	59	864	254		
Scene-8	4	8	2688	512	432	256	48		
Yale	3	15	165	4096	3304	6750			

1. **COIL20 [44]**: It consists of 1440 images of 20 classes. For each image, 1024-dimensional intensity, 3304-dimensional LBP and 6750-dimensional Gabor features were extracted.
2. **MSRCv1 [45]**: This dataset consists of seven categories with 210 data points in total. Each category has 30 data points and each data point is recorded by five features, including 254-D CENTRIST feature, 24-D color moment, 512-D GIST feature, 256-D LBP feature and 576-D HOG feature.
3. **ORL [46]**: The ORL face dataset contains 400 grayscale images of 40 subjects, with 10 images per subject captured under variations in facial expression, lighting, and the presence or absence of glasses.
4. **Yale [47]**: The Yale face dataset contains 165 grayscale face images of 15 subjects, with 11 images per subject captured under different facial expressions and illumination conditions.
5. **Scene-8 [45]**: This dataset is the image dataset for object recognition problem. The dataset has 2688 images consisting of eight groups. Following For each image, we choose four different feature vectors, including 512-D GIST, 432-D color moment, 256-D HOG, and 48-D LBP.
6. **NUS-2400**: The NUS dataset (also referred to as NUS-WIDE-OBJECT-12) is a widely used subset of NUS-WIDE, which contains 2400 images divided into 12 categories. Each sample is represented by six handcrafted features, i.e., color histogram (64D), color correlation (144D), edge distribution (73D), wavelet texture (128D), color moment (225D), and bag-of-SIFT (500D).

5.2. Competitors

For a comprehensive comparison, we compare the proposed CMTGL with nine representative multi-view clustering and semi-supervised multi-view learning methods. These compared methods cover nonconvex low-rank multi-view subspace clustering, tensor-based multi-view subspace clustering, constrained K-means based semi-supervised multi-view learning, graph-regularized NMF-based multi-view clustering, hypergraph-based semi-supervised multi-view clustering, embedding-regularization based semi-supervised multi-view learning, and scalable anchor-graph-based semi-supervised multi-view subspace clustering. The compared methods are summarized as follows.

1. **NLRSC-MvSC [48]**: A nonconvex multi-view subspace clustering method that introduces nonconvex low-rank regularization to capture the intrinsic global low-rank structures of multi-view data.
2. **TC-MVSC [43]**: A tensorized and compressed multi-view subspace clustering method that adopts compressed dictionary representation and low-rank tensor learning to

model the high-order correlations among multiple views. It is used to evaluate the effectiveness of tensor-based low-rank multi-view representation learning.

3. **MLCK [49]**: A constrained K-means-based semi-supervised multi-view learning method that can be regarded as a multi-view extension of constrained K-means. It incorporates the available label information into the clustering indicator matrix and learns view-specific centroid matrices, while treating all views equally.
4. **DACK [49]**: A data-driven auto-weighted constrained K-means-based semi-supervised multi-view learning method. It extends MLCK by assigning adaptive weights to different views according to their data fitting errors, so as to learn a more reliable indicator matrix from multi-view data.
5. **LACK [49]**: A label-driven auto-weighted constrained K-means-based semi-supervised multi-view learning method. Different from DACK, LACK evaluates the importance of each view from the label perspective by measuring whether the learned view-specific centroids can correctly predict the labels of labeled samples, which helps reduce the negative influence of low-quality views.
6. **GPSNMF [50]**: A graph-regularized partially shared non-negative matrix factorization method for semi-supervised multi-view clustering. It preserves the local geometric structure of each view by graph regularization and exploits both common and view-specific information through partially shared matrix factorization.
7. **MVCHSS [51]**: a multi-view clustering method with hypergraph semi-supervised learning, which exploits high-order relationships among samples through hypergraph learning.
8. **ERL-MVSC [52]**: An embedding regularizer learning method for multi-view semi-supervised learning. It learns view-specific embedding regularizers and adaptively integrates complementary information from multiple views to improve the discriminative ability of the learned representation.
9. **FSSMSC [53]**: A fast and scalable semi-supervised multi-view subspace clustering method that constructs a consensus anchor graph across multiple views and jointly performs graph learning and label propagation. It is adopted as a recent scalable semi-supervised multi-view subspace clustering baseline.

5.3. Evaluation Indicators

To comprehensively evaluate the clustering performance, six widely used metrics are adopted, including Accuracy (ACC), Normalized Mutual Information (NMI), Purity, Adjusted Rand Index (ARI), F-score, and Recall. For detailed definitions of these metrics, see [54,55]. A larger value of each metric indicates better clustering performance.

5.4. Experimental Settings

For the semi-supervised setting, a fixed proportion of samples is first selected as labeled samples, and the remaining samples are treated as unlabeled. Unless otherwise stated, the labeling rate is set to 20%. Let $\mathcal{L}$ and $\mathcal{U}$ denote the labeled and unlabeled index sets, respectively. The samples are reordered by placing the labeled samples before the unlabeled samples, such that $\mathbf{X}^v = [\mathbf{X}^v_{\mathcal{L}}, \mathbf{X}^v_{\mathcal{U}}]$ After the labeled subset is determined, synthetic outliers are introduced to evaluate robustness. For a prescribed outlier ratio, the replacement positions are randomly generated for each view, and the selected original samples are replaced by normalized outlier samples. The outlier positions are generated independently across different views; therefore, different views of the same multi-view dataset may be corrupted at different sample positions. Since the outlier replacement is performed after the labeled subset has been determined, the corrupted positions are not restricted to the unlabeled subset and may also include labeled samples. This setting

is intentionally retained to simulate a more general semi-supervised scenario in which corruption may affect both labeled and unlabeled observations.

For each dataset setting, one fixed corruption realization, including the synthetic outlier vectors and the view-specific replacement positions, is generated and saved. Ten independent class-stratified labeled subsets are then sampled under the same fixed corrupted observations. All compared methods use exactly the same saved corruption and the same labeled split in each corresponding trial. The reported results are the mean and standard deviation over these ten labeled-split trials.

The maximum number of iterations of CMTGL is set to 100. The principal trade-off parameters are determined through a coarse parameter exploration on a representative realization. This procedure is used only to identify a reasonable and stable parameter range rather than to optimize the parameters separately for every reported test realization. Once determined, the same parameter configuration is kept fixed throughout the repeated experiments, specifically $\beta \in \{1, 10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}\}, \lambda \in \{10^{-1}, 1, 10, 100\}, \gamma \in \{10^{-1}, 1, 10, 100\}$. The parameter α is fixed to 100 after simple analysis and value-selection experiments, and the parameter ρ is set to 0.0005. Within the explored candidate parameter ranges, we configure fixed parameter combinations adapted to different datasets. For the six experimental datasets, the corresponding (β, λ, γ) values are set to (0.1, 100, 10), (1, 100, 1), (1, 100, 10), (1, 100, 1), (0.001, 1, 10), and (0.1, 100, 0.1) in sequence.

The Schatten parameter p is investigated separately. To isolate its influence, all other parameters, the labeled-data split, and the corrupted multi-view samples are kept unchanged while only p is varied. We evaluate $p \in \{0.1, 0.5, 0.7, 1, 1.5, 2\}$. As reported in Table 3, the clustering performance varies only slightly across different p values. Moreover, the value producing the highest score differs across datasets and evaluation metrics. This indicates that CMTGL is relatively insensitive to p within the investigated range. We therefore set $p = 1$ in the main experiments because it provides a simple and stable configuration and corresponds to the standard tensor nuclear-norm-type spectral regularization. The broader range of p is retained as an extensible mechanism for adapting the spectral regularization to different datasets when necessary.

For each MCC term, the kernel bandwidth is adaptively determined by its corresponding reconstruction residual: $\sigma_1 = ((\theta/n)\|\mathbf{X}^v - \mathbf{W}^v\mathbf{M}\mathbf{A}\|_F^2))^{1/2}, \sigma_2 = ((\theta/n)\|\mathbf{X}^v - \mathbf{X}^v\mathbf{Z}^v\|_F^2))^{1/2}, \sigma_3 = ((\theta/n)\|\mathbf{M}\mathbf{A} - \mathbf{M}\mathbf{A}\mathbf{S}\|_F^2))^{1/2}, \sigma_4 = ((\theta/V)\sum_{v=1}^{V}\|\mathbf{Z}^v - \mathbf{S}\|_F^2))^{1/2}$, where $\theta = 1$. After optimization, the class-related representation $\mathbf{M}\mathbf{A} \in \mathbb{R}^{c\times n}$ is used to generate the final clustering assignments each row of $\mathbf{M}\mathbf{A}$ corresponds to one class. The predicted label of the i-th sample is therefore given by

$$\hat{y}_i = \arg\max_{1\le j\le c} (\mathbf{M}\mathbf{A})_{ji}. \tag{39}$$

For the comparison algorithms, publicly available code implementations and parameter settings recommended in the original reference papers were used. All algorithms employed identical semi-supervised sample splits and contaminated data. Furthermore, clustering evaluation metrics were computed on the complete dataset, including both labeled and unlabeled samples, with all algorithms following this unified evaluation protocol.

Table 3. Clustering performance with different p values on benchmark datasets

Dataset	p	ACC (%)	NMI (%)	Purity (%)	ARI (%)	F-Score (%)	Recall (%)
MSRCv1	0.1	88.10	80.36	88.10	76.81	80.06	81.05
	0.5	88.10	79.20	88.10	76.53	79.84	81.08
	0.7	88.10	79.20	88.10	76.53	79.84	81.08
	1	88.10	80.79	88.10	76.84	80.11	81.54
	1.5	87.62	79.38	87.62	76.06	79.42	80.43
	2	87.14	79.11	87.14	75.45	78.92	80.43

Table 3. *Cont.*

Dataset	p	ACC (%)	NMI (%)	Purity (%)	ARI (%)	F-Score (%)	Recall (%)
ORL	0.1	89.00	91.42	89.00	78.52	79.02	81.83
	0.5	89.00	91.45	89.00	78.40	78.91	81.78
	0.7	89.00	91.58	89.00	78.41	78.92	81.94
	1	89.00	91.27	89.00	78.39	78.00	81.61
	1.5	88.50	91.32	88.50	77.74	78.26	81.39
	2	88.75	91.51	88.75	78.06	78.57	81.78
Scene-8	0.1	63.32	45.16	63.32	37.98	46.29	48.98
	0.5	63.24	45.01	63.24	37.88	46.20	48.88
	0.7	63.24	45.01	63.24	37.88	46.20	48.88
	1	62.98	44.95	63.24	37.76	46.05	48.81
	1.5	63.06	44.69	63.06	37.60	45.96	48.62
	2	62.95	44.56	62.95	37.47	45.86	48.64
Yale	0.1	79.39	76.37	79.39	59.02	61.57	63.03
	0.5	79.39	76.37	79.39	59.02	61.57	63.03
	0.7	79.39	76.37	79.39	59.02	61.57	63.03
	1	79.39	76.37	79.39	59.02	61.57	63.03
	1.5	79.39	76.37	79.39	59.02	61.57	63.03
	2	79.39	76.37	79.39	59.02	61.57	63.03

All values show mean ± standard deviation (%).

5.5. Comparison of Clustering Performance

All compared algorithms are repeated ten times on these datasets and we present the clustering results in Table 4. The best and second-best results are highlighted in bold and underlined, respectively. From the results, several observations can be made.

First, CMTGL achieves the best ACC and Purity on all six datasets, which demonstrates that the proposed method can effectively improve the overall clustering accuracy and class-wise clustering consistency. In particular, CMTGL obtains ACC values of 95.12%, 88.86%, 51.75%, 88.88%, 63.43%, and 77.88% on COIL20, MSRCv1, NUS-2400, ORL, Scene-8, and Yale, respectively. These results show that the proposed method performs consistently well on both face image datasets and general image datasets.

Second, CMTGL achieves the best ARI and F-score on most datasets. On COIL20, ORL, Scene-8, and Yale, CMTGL outperforms all compared methods in terms of ACC, Purity, ARI, F-score, and Recall. This indicates that the proposed label-constrained low-rank representation and consensus graph learning strategy can produce more reliable clustering partitions. The improvement is especially clear on Yale and Scene-8, where CMTGL achieves superior performance across all six evaluation metrics. This suggests that CMTGL can effectively exploit the available label information while preserving the local and global structures of multi-view data.

Third, the tensor spectral regularization contributes to the structural modeling capability of CMTGL. By stacking the view-specific graphs into a third-order tensor, CMTGL explicitly exploits high-order dependencies across multiple views, while the Schatten parameter p provides additional flexibility in controlling the singular-value penalty. The parameter study further shows that the overall clustering performance is relatively stable across different p values, suggesting that the effectiveness of CMTGL does not rely on a narrowly tuned Schatten parameter. Instead, the tensor regularization mainly serves to preserve the shared low-dimensional graph structure, while p provides an extensible mechanism for adapting the spectral penalty when necessary.

Table 4. Experimental results obtained by our method, CMTGL, with other clustering methods on these benchmark datasets.

Methods/COIL20	ACC	NMI	Purity	ARI	F-Score	Recall
NLRSC-MvSC	71.56 ± 1.63	81.75 ± 0.84	75.34 ± 1.80	63.87 ± 1.56	65.79 ± 1.46	71.17 ± 1.37
TC-MVSC	65.17 ± 2.21	74.92 ± 1.47	66.62 ± 1.96	40.30 ± 3.84	44.22 ± 3.42	66.07 ± 1.95
MLCK	83.07 ± 1.35	84.55 ± 0.90	83.07 ± 1.35	74.07 ± 1.55	75.37 ± 1.47	76.87 ± 1.40
DACK	83.42 ± 1.44	85.18 ± 1.04	83.49 ± 1.52	74.96 ± 1.85	76.22 ± 1.76	77.72 ± 1.66
LACK	82.81 ± 1.15	84.34 ± 1.00	82.88 ± 1.23	73.80 ± 1.59	75.12 ± 1.51	76.54 ± 1.42
GPSNMF	92.98 ± 1.16	91.07 ± 1.16	92.98 ± 1.16	86.35 ± 2.03	87.03 ± 1.93	87.50 ± 1.85
MVCHSS	90.78 ± 4.35	94.92 ± 1.93	91.31 ± 3.84	87.67 ± 4.64	88.32 ± 4.39	93.23 ± 3.16
ERL-MVSC	92.17 ± 1.05	91.30 ± 1.07	92.17 ± 1.05	86.43 ± 1.68	87.10 ± 1.60	87.36 ± 1.61
FSSMSC	94.64 ± 0.95	93.74 ± 0.99	94.64 ± 0.95	89.85 ± 1.89	90.35 ± 1.79	90.73 ± 1.56
CMTGL	**95.12 ± 0.85**	**94.28 ± 0.81**	**95.12 ± 0.85**	**90.92 ± 1.49**	**91.37 ± 1.42**	**91.49 ± 1.36**
Methods/MSRCv1	**ACC**	**NMI**	**Purity**	**ARI**	**F-Score**	**Recall**
NLRSC-MvSC	78.29 ± 1.74	69.53 ± 1.88	78.38 ± 1.57	63.05 ± 2.40	68.31 ± 2.08	70.13 ± 2.42
TC-MVSC	60.48 ± 3.92	50.09 ± 4.24	60.81 ± 3.58	35.70 ± 3.89	45.94 ± 3.21	53.87 ± 3.69
MLCK	77.95 ± 4.99	68.87 ± 3.90	78.52 ± 4.36	62.18 ± 5.75	67.66 ± 4.87	70.66 ± 4.27
DACK	77.52 ± 4.45	68.32 ± 3.25	78.05 ± 4.05	61.41 ± 5.18	66.99 ± 4.38	69.83 ± 3.80
LACK	82.38 ± 4.84	72.76 ± 4.97	82.52 ± 4.55	67.29 ± 6.48	71.99 ± 5.50	74.36 ± 4.77
GPSNMF	78.76 ± 3.34	63.78 ± 4.37	78.76 ± 3.34	56.99 ± 6.30	63.12± 5.29	64.58 ± 4.24
MVCHSS	88.33 ± 6.53	**80.31 ± 5.50**	88.67 ± 5.55	**77.41 ± 7.73**	**80.58 ± 6.60**	**81.18 ± 5.97**
ERL-MVSC	85.52 ± 2.19	74.32 ± 2.89	85.52 ± 2.19	70.25 ± 3.77	74.45 ± 3.20	75.78 ± 2.59
FSSMSC	76.24 ± 1.88	61.98 ± 2.54	76.24 ± 1.88	54.41 ± 3.25	60.92 ± 2.78	62.70 ± 3.04
CMTGL	**88.86 ± 2.59**	79.71 ± 4.52	**88.86 ± 2.59**	76.60 ± 5.02	79.90 ± 4.31	81.12 ± 4.23
Methods/NUS-2400	**ACC**	**NMI**	**Purity**	**ARI**	**F-Score**	**Recall**
NLRSC-MvSC	22.72 ± 0.59	10.66 ± 0.36	24.48 ± 0.63	5.46 ± 0.26	13.52 ± 0.23	13.90 ± 0.23
TC-MVSC	20.65 ± 0.98	12.41 ± 0.77	22.03 ± 0.82	1.83 ± 0.43	15.06 ± 0.13	**40.61 ± 4.51**
MLCK	42.71 ± 0.46	22.84 ± 0.41	42.71 ± 0.46	16.14 ± 0.39	23.29 ± 0.35	23.94 ± 0.40
DACK	42.67 ± 0.65	22.96 ± 0.48	42.67 ± 0.65	16.04 ± 0.55	23.24 ± 0.48	24.02 ± 0.48
LACK	43.22 ± 0.62	23.33 ± 0.36	43.22 ± 0.62	16.60 ± 0.52	23.69 ± 0.43	24.30 ± 0.34
GPSNMF	36.37 ± 1.12	16.55 ± 1.01	36.37 ± 1.12	10.38 ± 1.00	18.06 ± 0.83	18.65 ± 0.73
MVCHSS	36.71 ± 2.30	21.11 ± 1.10	37.92 ± 1.90	15.48 ± 1.57	22.60 ± 1.45	22.98 ± 1.50
ERL-MVSC	48.93 ± 0.58	26.19 ± 0.59	48.93 ± 0.58	21.68 ± 0.57	28.32 ± 0.53	28.99 ± 0.58
FSSMSC	43.10 ± 0.65	20.83 ± 0.54	43.10 ± 0.65	15.85 ± 0.58	22.87 ± 0.53	23.00 ± 0.51
CMTGL	**51.75 ± 0.92**	**29.51 ± 0.82**	**51.75 ± 0.92**	**24.29 ± 0.83**	**30.79 ± 0.75**	31.91 ± 0.81
Methods/ORL	**ACC**	**NMI**	**Purity**	**ARI**	**F-Score**	**Recall**
NLRSC-MvSC	76.05 ± 2.99	87.30 ± 1.68	78.17 ± 2.91	66.50 ± 3.27	67.30 ± 3.19	72.12 ± 3.63
TC-MVSC	59.05 ± 1.52	77.84 ± 0.53	59.40 ± 1.46	44.57 ± 1.25,	45.83 ± 1.23	46.26 ± 1.24
MLCK	74.65 ± 3.78	83.68 ± 2.41	74.78 ± 3.73	56.03 ± 5.36	57.19 ± 5.19	68.52 ± 4.01
DACK	74.23 ± 3.25	83.56 ± 1.87	74.42 ± 3.23	55.50 ± 4.25	56.68 ± 4.11	68.30 ± 3.05
LACK	74.90 ± 4.04	83.72 ± 2.60	74.98 ± 3.99	56.10 ± 5.87	57.26 ± 5.68	68.66 ± 4.32
GPSNMF	74.35 ± 2.59	80.73 ± 1.80	74.38 ± 2.59	56.55 ± 3.50	57.58 ± 3.41	60.42 ± 3.30
MVCHSS	80.75 ± 2.74	90.21 ± 1.23	82.10 ± 2.14	73.60 ± 3.08	74.22 ± 3.00	78.16 ± 2.83
ERL-MVSC	79.85 ± 2.48	84.63 ± 1.77	79.85 ± 2.48	64.37 ± 4.13	65.20 ± 4.02	67.85 ± 3.55
FSSMSC	85.55 ± 2.31	90.35 ± 1.14	85.60 ± 2.21	75.22 ± 2.94	75.81 ± 2.87	79.03 ± 2.49
CMTGL	**88.88 ± 2.14**	**91.20 ± 1.54**	**88.88 ± 2.14**	**78.84 ± 3.75**	**79.33 ± 3.66**	**80.91 ± 3.32**
Methods/Scene-8	**ACC**	**NMI**	**Purity**	**ARI**	**F-Score**	**Recall**
NLRSC-MvSC	34.62 ± 1.83	22.49 ± 1.95	35.26 ± 1.71	15.65 ± 1.47	26.68 ± 1.24	27.48 ± 1.19
TC-MVSC	30.97 ± 0.29	20.15 ± 0.20	33.91 ± 0.19	7.04 ± 0.22	23.59 ± 0.12	39.49 ± 0.43
MLCK	47.19 ± 2.47	28.63 ± 2.93	47.19 ± 2.47	20.55 ± 2.19	30.78 ± 1.95	31.26 ± 2.24
DACK	46.17 ± 2.34	27.63 ± 2.54	46.17 ± 2.34	19.21 ± 2.07	29.64 ± 1.86	30.18 ± 2.14
LACK	53.08 ± 2.48	35.76 ± 1.38	53.24 ± 2.49	27.40 ± 1.93	36.71 ± 1.66	37.06 ± 1.66
GPSNMF	42.27 ± 2.11	17.22 ± 1.58	42.27 ± 2.11	13.13 ± 1.73	24.25 ± 1.48	24.45 ± 1.44
MVCHSS	54.25 ± 4.15	38.19 ± 2.64	55.79 ± 3.45	30.60 ± 2.89	39.41 ± 2.51	39.42 ± 2.44
ERL-MVSC	60.99 ± 1.03	39.77 ± 1.12	60.99 ± 1.03	34.34 ± 1.27	42.76 ± 1.11	43.19 ± 1.145
FSSMSC	54.37 ± 1.15	34.15 ± 0.78	54.37 ± 1.15	26.33 ± 1.18	35.70 ± 1.02	35.76 ± 0.98
CMTGL	**63.43 ± 1.08**	**45.10 ± 1.15**	**63.43 ± 1.08**	**38.20 ± 1.32**	**46.56 ± 1.12**	**49.88 ± 1.25**
Methods/Yale	**ACC**	**NMI**	**Purity**	**ARI**	**F-Score**	**Recall**
NLRSC-MvSC	66.61 ± 2.45	69.39 ± 1.80	66.67 ± 2.37	49.23 ± 2.77	52.47 ± 2.57	54.92 ± 2.43
TC-MVSC	43.45 ± 2.16	47.01 ± 2.50	44.24 ± 1.92	12.23 ± 2.11	20.04 ± 1.64	37.27 ± 3.38
MLCK	65.15 ± 4.10	66.29 ± 3.32	65.21 ± 4.06	43.59 ± 5.23	47.31 ± 4.79	51.38 ± 3.82
DACK	65.03 ± 3.84	66.44 ± 3.25	65.03 ± 3.84	43.78 ± 4.96	47.48 ± 4.54	51.52 ± 3.58
LACK	65.64 ± 3.90	66.61 ± 3.08	65.70 ± 3.90	44.16 ± 4.87	47.83 ± 4.45	51.73 ± 3.29
GPSNMF	68.00 ± 4.01	67.83 ± 3.08	68.30 ± 3.63	42.29 ± 4.90	46.01 ± 4.54	49.96 ± 4.44
MVCHSS	66.00 ± 3.88	68.60 ± 2.33	66.42 ± 3.85	49.18 ± 3.51	52.34 ± 3.29	53.36 ± 3.33
ERL-MVSC	61.26 ± 4.23	61.30 ± 3.02	61.56 ± 4.20	35.68 ± 3.70	39.77 ± 3.48	42.69 ± 4.17
FSSMSC	71.88 ± 3.52	69.23 ± 2.94	71.88 ± 3.52	47.40 ± 4.33	50.80 ± 4.02	54.12 ± 4.00
CMTGL	**77.88 ± 3.64**	**74.39 ± 3.59**	**77.88 ± 3.64**	**57.25 ± 5.30**	**59.93 ± 4.97**	**61.67 ± 5.08**

Bold values indicate the best results, and underlined values indicate the second-best results. All values show mean ± standard deviation (%).

Fourth, compared with semi-supervised baselines such as MLCK, DACK, LACK, GPSNMF, ERL-MVSC, MVCHSS, and FSSMSC, CMTGL shows more stable performance across datasets. On ORL, CMTGL improves the ACC from 85.55% achieved by FSSMSC to 88.88%. On Yale, CMTGL improves the ACC from 71.88% achieved by FSSMSC to 77.88%. On Scene-8, CMTGL achieves 63.43% ACC, which is higher than the second-best result of 60.99% obtained by ERL-MVSC. These improvements demonstrate that the integration of MCC, low-rank matrix factorization, label constraints, and tensor graph learning is beneficial for semi-supervised multi-view clustering.

It is worth noting that the proposed CMTGL exhibits relatively lower performance on NUS-2400 and Scene-8 datasets. This is mainly attributed to the intrinsic characteristics of these two scene datasets: different views correspond to heterogeneous low-level visual features with significantly weaker inter-view semantic correlation compared to object datasets like COIL20 and ORL. The core advantage of CMTGL lies in exploiting high-order low-rank correlations among views via tensor Schatten p-norm regularization, which is less effective when cross-view consistency is insufficient. Additionally, scene datasets inherently suffer from large intra-class variations and limited discriminative power of handcrafted features, further challenging the semi-supervised clustering performance. It should also be noted that CMTGL already contains a partial safeguard against unreliable views. Specifically, the correntropy-based graph consistency term adaptively reduces the influence of a view when its learned graph deviates substantially from the consensus graph. Therefore, severely inconsistent views are not treated equally during consensus learning. Nevertheless, the current tensor regularization still relies on exploitable shared low-rank correlations across views and does not explicitly separate shared and view-specific structures. For weakly correlated multi-view data, a promising extension is to decompose each view into shared and private graph components and learn the cross-view coupling strength adaptively.

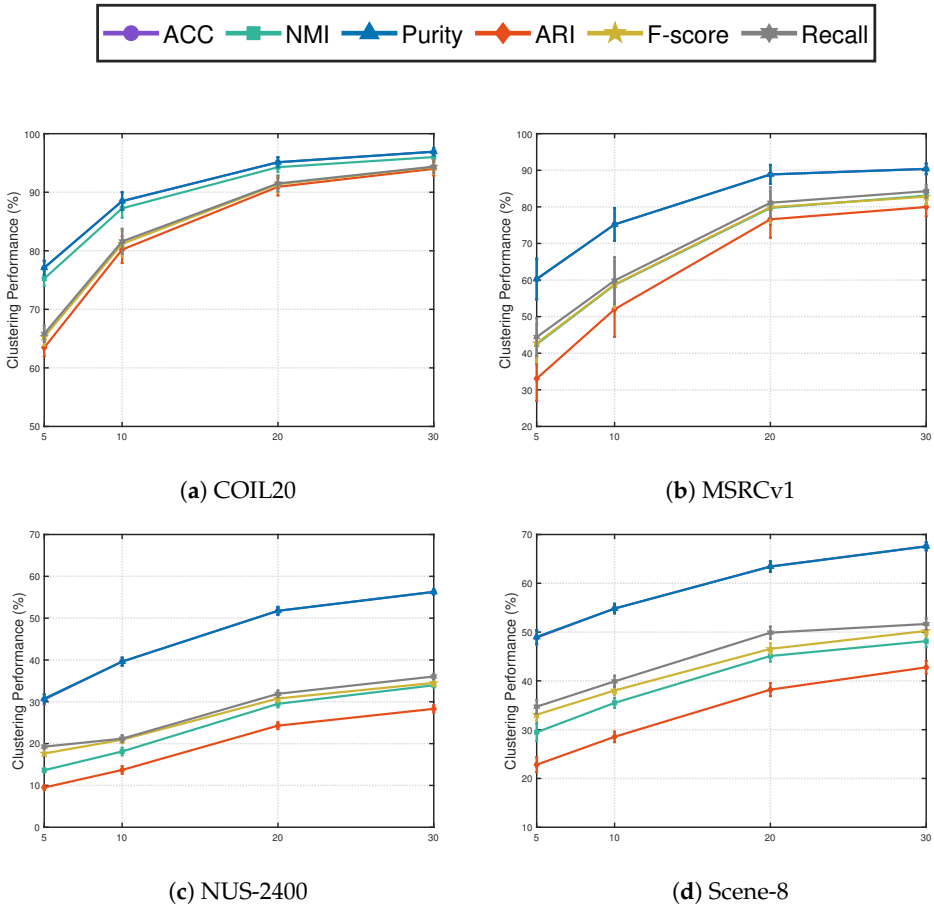
Finally, although some compared methods obtain competitive results on individual metrics, CMTGL provides the most balanced performance across all metrics. For example, MVCHSS obtains competitive NMI, ARI, F-score, and Recall on MSRCv1, while CMTGL still achieves the best ACC and Purity on this dataset. On NUS-2400, TC-MVSC obtains a higher Recall, but its ACC, NMI, Purity, ARI, and F-score are significantly lower than those of CMTGL. Therefore, considering all evaluation metrics together, the proposed CMTGL demonstrates stronger and more consistent clustering performance on the evaluated multi-view datasets.

5.6. Influence of the Labeling Rate

To further evaluate the behavior of CMTGL under different amounts of supervisory information, we vary the labeling rate over $\{5\%, 10\%, 20\%, 30\%\}$ on four representative datasets, while keeping the remaining experimental settings unchanged. The results are shown in Figure 3.

As the labeling rate increases, the clustering performance generally improves across all four datasets. The improvement is particularly evident when the labeling rate increases from 5% to 10% or 20%, indicating that the label-constrained representation can effectively exploit additional supervisory information. When the labeling rate further increases, the performance gains become relatively moderate on several datasets, suggesting that the model remains effective even when only a limited proportion of labeled samples is available.

Overall, these results demonstrate that CMTGL can benefit from increasing label information while maintaining reasonable clustering performance under relatively low labeling rates.



(**a**) COIL20 (**b**) MSRCv1

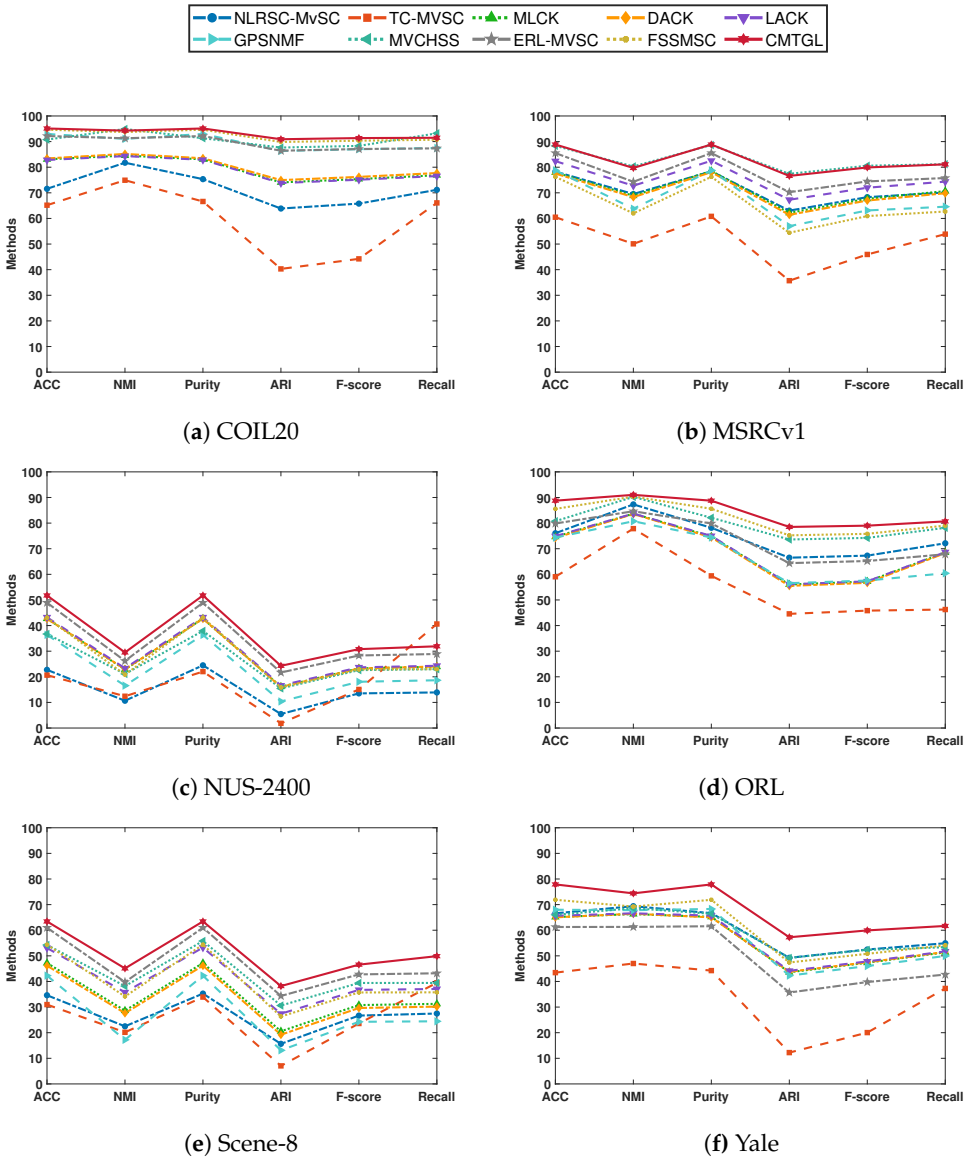
(**c**) NUS-2400 (**d**) Scene-8

Figure 3. Clustering performance of CMTGL under different labeling rates on four benchmark datasets. The labeling rate varies from 5% to 30%, while the remaining experimental settings are kept unchanged.

5.7. Visualization of Clustering Performance

To provide a more intuitive comparison of different methods, we further visualize the clustering results reported in Table 4. For each dataset, six evaluation metrics (ACC, NMI, Purity, ARI, F-score, and Recall) are arranged along the horizontal axis, with performance scores plotted for all competitors. Each curve corresponds to one compared algorithm, including several state-of-the-art multi-view clustering approaches and the proposed CMTGL. This visualization enables simultaneous observation of performance variations of different methods across multiple metrics.

As shown in Figure 4, CMTGL generally maintains a higher and more stable performance profile across the six metrics on most datasets. Compared with methods that perform well only on specific metrics or datasets, the proposed method shows more balanced behavior in terms of clustering accuracy, information consistency, pairwise agreement, and cluster purity. This further demonstrates that the integration of label-constrained low-rank matrix factorization, MCC-based robust weighting, and tensor S_p-norm graph learning can effectively improve the overall clustering quality.



(**a**) COIL20 (**b**) MSRCv1

(**c**) NUS-2400 (**d**) ORL

(**e**) Scene-8 (**f**) Yale

Figure 4. Visualization of clustering performance on six benchmark datasets. The horizontal axis denotes the six evaluation metrics, and each curve corresponds to one compared method. The vertical axis denotes the clustering score in percentage.

5.8. View Weight and Robust Representation Analysis

To further verify the adaptive view weighting capability of the proposed CMTGL, we present the normalized weights of all views across six benchmark datasets in Figure 5. It can be observed that the model automatically assigns differentiated weights to different views according to their discriminative contributions. Views with stronger clustering informativeness obtain higher weights, while less informative views are assigned lower weights, which is consistent with the correntropy-induced robust weighting mechanism. Specifically, on COIL20 and Yale datasets, the intensity feature and Gabor feature achieve significantly higher weights than the LBP feature, indicating that grayscale and Gabor descriptors provide more discriminative information for object and face clustering tasks. On the NUS-2400 dataset, the weights of six handcrafted features are relatively balanced, reflecting the complementary effect of multi-modal features on complex scene datasets. This adaptive weighting strategy enables the model to suppress the interference of low-quality views and fully exploit effective information from high-quality views, thus boosting the overall clustering performance.

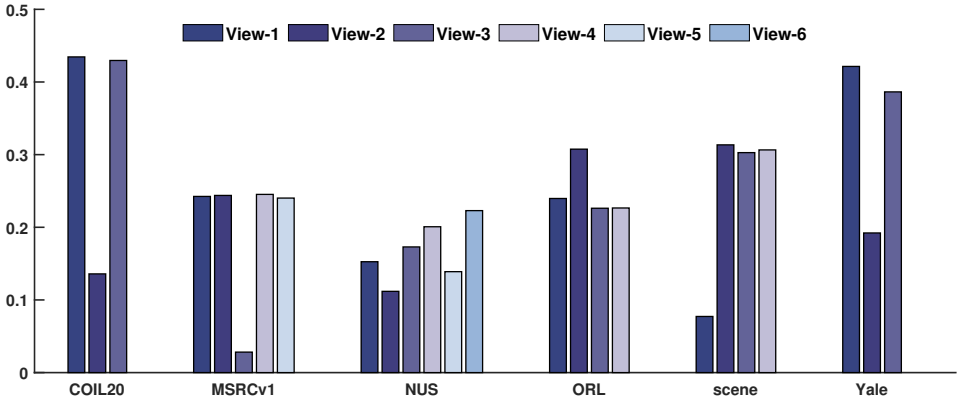


Figure 5. Adaptive view weights learned by the proposed CMTGL on six benchmark datasets. Bars with different colors correspond to different views of each dataset, and the height of each bar represents the normalized contribution weight of the corresponding view.

To further provide a qualitative view of how data corruption affects the learned representations, we randomly select one fixed corrupted realization of the Yale dataset and visualize representative basis or anchor patterns learned by several methods. The first row of Figure 6 shows representative corrupted input samples, while the subsequent rows visualize representative patterns extracted from NLRSC-MvSC, TC-MVSC, GPSNMF, FSSMSC, and CMTGL, respectively.

Since the compared methods employ different internal representations, their basis or anchor structures are converted into the same number of representative patterns only for visualization. For CMTGL, the columns of the learned view-specific basis matrix $\mathbf{W}^{(1)}$ are visualized directly. For methods with a larger number of anchors or latent bases, their internal representatives are grouped into c clusters and the corresponding centers are displayed.

Figure 6. The first row shows representative corrupted input samples. The remaining rows show representative basis or anchor patterns obtained from NLRSC-MvSC, TC-MVSC, GPSNMF, FSSMSC, and CMTGL, respectively. For methods whose internal representation contains more than c bases or anchors, c representative centers are obtained for visualization.

As shown in Figure 6, the injected outliers introduce visually irregular and noise-dominated patterns into the corrupted data. In contrast, CMTGL learns relatively structured and interpretable representative patterns despite the presence of corruption. This qualitative observation is consistent with the role of the maximum correntropy criterion, which reduces the influence of samples associated with large reconstruction residuals. The visualization therefore provides complementary evidence that the proposed robust representation learning mechanism can alleviate the influence of corrupted observations.

5.9. Ablation Study

To investigate the contribution of the major components of CMTGL, we conduct ablation experiments by removing the maximum correntropy criterion, tensor regularization, and label constraint, respectively. We denote these three components as C_1 (MCC-based robust learning), C_2 (tensor Schatten p-norm regularization), and C_3 (label-constrained representation learning).

All ablation variants use exactly the same labeled splits and corrupted data realizations as the complete model. In particular, the same ten corrupted realizations are used for every variant so that the performance differences are attributable to the removed component rather than random data corruption. When the tensor regularization is removed, the corresponding regularization coefficient ρ is set to zero. When the label constraint is removed, the constraint matrix is replaced by the identity matrix, eliminating the class-wise label tying in the representation.

Tables 5 and 6 report the ACC and NMI results on all six datasets. Removing any component generally degrades the clustering performance, while the complete CMTGL achieves the most consistent overall results. The removal of MCC leads to noticeable performance degradation, particularly on MSRCv1 and several noisy datasets, confirming the importance of robust residual weighting. Removing tensor regularization also reduces performance on most datasets, indicating that high-order cross-view structural information contributes to the learned representation. The largest deterioration occurs when the label constraint is removed, demonstrating that partial label information plays a central role in the proposed semi-supervised framework.

Table 5. Ablation study on COIL20, MSRCv1, and NUS-2400.

			COIL20		**MSRCv1**		**NUS-2400**	
***C*1**	***C*2**	***C*3**	**ACC**	**NMI**	**ACC**	**NMI**	**ACC**	**NMI**
	✓	✓	94.25 ± 0.89	92.52 ± 1.00	84.24 ± 1.77	71.82 ± 2.71	49.05 ± 0.55	26.01 ± 0.68
✓		✓	95.12 ± 0.85	94.31 ± 0.79	86.76 ± 2.36	76.61 ± 3.41	50.71 ± 0.80	28.76 ± 0.59
✓	✓		9.86 ± 0.00	4.33 ± 0.00	19.52 ± 0.00	2.78 ± 0.00	11.62 ± 0.00	1.38 ± 0.00
✓	✓	✓	$\mathbf{95.12 \pm 0.85}$	$\mathbf{94.28 \pm 0.81}$	$\mathbf{88.86 \pm 2.59}$	$\mathbf{79.71 \pm 4.52}$	$\mathbf{51.75 \pm 0.92}$	$\mathbf{29.51 \pm 0.82}$

✓ denotes inclusion. Bold values indicate the best results. All values show mean ± standard deviation (%).

Table 6. Ablation study on ORL, Scene-8, and Yale.

			ORL		**Scene-8**		**Yale**	
***C*1**	***C*2**	***C*3**	**ACC**	**NMI**	**ACC**	**NMI**	**ACC**	**NMI**
	✓	✓	82.00 ± 3.28	85.21 ± 2.57	61.42 ± 0.68	41.33 ± 0.73	77.58 ± 4.07	74.24 ± 3.87
✓		✓	87.45 ± 1.69	90.09 ± 1.26	62.37 ± 0.87	43.73 ± 0.87	77.03 ± 3.56	73.55 ± 3.53
✓	✓		15.50 ± 0.00	39.80 ± 0.04	14.69 ± 0.00	0.48 ± 0.00	20.61 ± 0.00	23.38 ± 0.00
✓	✓	✓	$\mathbf{88.88 \pm 2.14}$	$\mathbf{91.20 \pm 1.54}$	$\mathbf{63.43 \pm 1.08}$	$\mathbf{45.10 \pm 1.15}$	$\mathbf{77.88 \pm 3.64}$	$\mathbf{74.39 \pm 3.59}$

✓ denotes inclusion. Bold values indicate the best results. All values show mean ± standard deviation (%).

These results indicate that MCC-based robust learning, tensor graph regularization, and label-constrained representation learning provide complementary contributions to the complete CMTGL model.

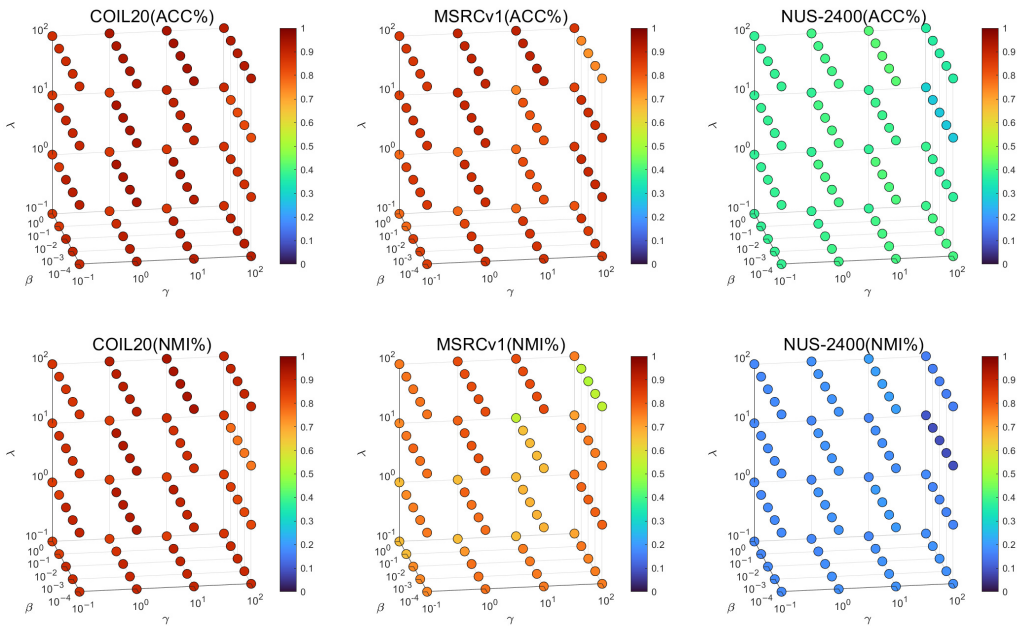
5.10. Parameter Sensitivity Analysis

The proposed CMTGL framework contains several trade-off parameters, including β, λ, γ, and the Schatten parameter p. Among them, β controls the sparsity regularization of the consensus graph, λ balances the label-constrained representation propagation term,

and γ controls the consistency between the view-specific graphs and the consensus graph. To investigate the influence of these parameters, we conduct a sensitivity analysis on all six benchmark datasets by varying $\beta \in \{1, 10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}\}$, $\lambda \in \{10^{-1}, 1, 10, 100\}$, $\gamma \in \{10^{-1}, 1, 10, 100\}$. The clustering performance is evaluated by ACC and NMI, while the other experimental settings are kept unchanged. As shown in Figure 7, CMTGL exhibits relatively stable performance within a broad range of parameter combinations. This indicates that the proposed model is not overly sensitive to a single parameter. In particular, appropriate values of λ and γ help balance label-guided representation propagation and cross-view graph consistency. When these two terms are too weak, the model may not sufficiently exploit the supervisory information and the common structure shared by multiple views. On the other hand, excessively large regularization may overemphasize graph consistency or label propagation, which can reduce the flexibility of view-specific graph learning. The parameter β controls the sparsity degree of the consensus graph; a moderate sparsity constraint is beneficial for removing redundant connections while preserving the intrinsic neighborhood structure. A coarse parameter exploration is first conducted to identify reasonable operating ranges for β, λ, and γ. For each dataset, one configuration within the stable region is then fixed before the repeated trials and is kept unchanged over all ten corrupted realizations.

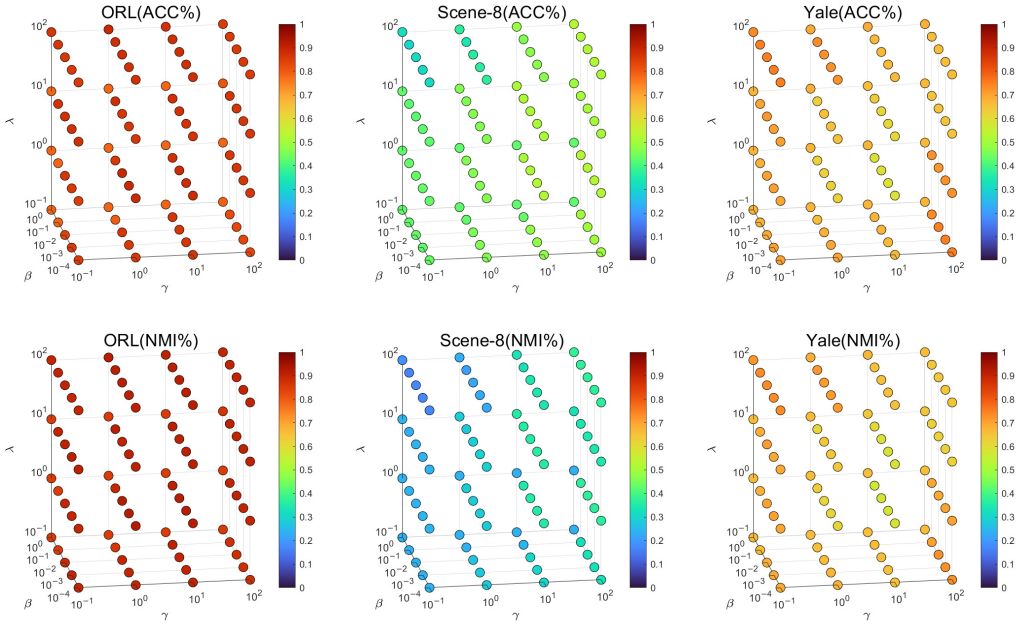
5.11. Empirical Convergence Behavior

The objective-value sequence generated by Algorithm 2 is theoretically convergent under the stated assumptions. To further illustrate its numerical behavior, Figure 8 shows the objective-value curves of CMTGL on the six benchmark datasets. The objective values gradually stabilize after a limited number of iterations on all datasets, which is consistent with the theoretical objective-sequence convergence analysis.



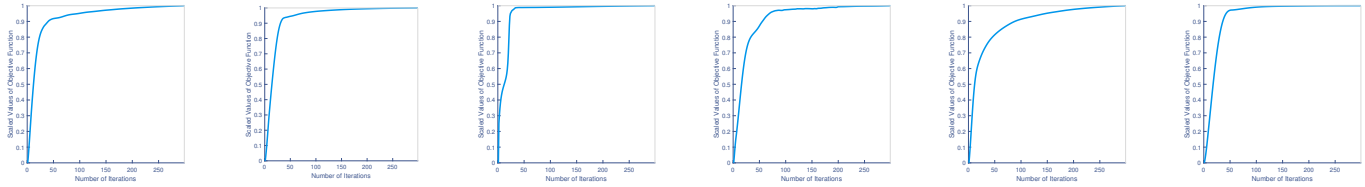
(**a**) (**b**) (**c**)

Figure 7. *Cont.*



(**d**) (**e**) (**f**)

Figure 7. Parameter sensitivity visualization of CMTGL on the six benchmark datasets. Each point corresponds to one combination of β, λ, and γ, and its color indicates the corresponding ACC or NMI. This experiment is used to examine the stability of the model with respect to parameter variations.



(**a**) (**b**) (**c**) (**d**) (**e**) (**f**)

Figure 8. Convergence curve of CMTGL method over six datasets. Subplots (**a**–**f**) correspond to the COIL20, MSRCv1, NUS-2400, ORL, Scene-8, and Yale datasets, respectively.

6. Conclusions

In this paper, we propose a robust semi-supervised multi-view clustering method named CMTGL, which combines correntropy-based low-rank matrix factorization with tensor low-rank graph learning. The method embeds partial label information into representation learning via a constraint matrix, adopts the maximum correntropy criterion to enhance noise robustness, and imposes tensor Schatten p-norm regularization on stacked view graphs to exploit high-order cross-view correlations. Comprehensive experiments on six public benchmark datasets verify that the proposed method achieves competitive clustering performance and maintains good stability under noisy conditions. For future work, we intend to design more adaptive view fusion mechanisms for weakly correlated multi-view data, and further optimize the model to achieve better efficiency and performance on large-scale big data scenarios.

Author Contributions: L.H.: Writing—original draft, resources; S.J.: Writing—original draft, methodology, visualization, investigation; X.L.: Conceptualization, methodology, visualization, writing—review and editing; P.S.: Supervision, writing—review and editing; Y.Y.: Data curation, writing—review and editing; N.Z.: Writing—review and editing, supervision. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported, in part, by the Sichuan Science and Technology Program under Grant 2025ZNSFSC1483.

Data Availability Statement: The datasets used in this study are publicly available from their original repositories or established multi-view learning benchmark sources. Specifically, COIL20 is available from the Columbia University Image Library at https://www.cs.columbia.edu/CAVE/software/softlib/coil-20.php; MSRCv1 is available from the public MSRCv1 repository at https://mldta.com/dataset/msrc-v1/home; the NUS-WIDE database, from which the NUS-2400 benchmark subset is derived, is available from the National University of Singapore at https://lms.comp.nus.edu.sg/research/NUS-WIDE.htm; the ORL face database is available from the Database of Faces at https://cam-orl.co.uk/facedatabase.html; and the Yale Face Database is available from Yale University at https://vision.ucsd.edu/content/yale-face-database. The multi-view feature representations corresponding to the NUS-2400 and Scene-8 benchmark settings used in this study are catalogued in a public multi-view dataset repository at https://github.com/zskong/Multi-view-data. All datasets and benchmark resources were accessed on 10 September 2026.

Acknowledgments: The authors acknowledged the use of the AI tool ChatGPT (GPT-5.6, OpenAI) during the preparation of the manuscript, solely for language polishing purposes, including improving grammar, readability, and clarity of expression.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A. Derivations of Gradients and Lipschitz Constant

Appendix A.1. The Derivation of $\nabla_{\mathbf{Z}^v} f$, $\nabla_{\mathbf{M}} f$ *and* $\nabla_{\mathbf{S}} f$

The original semi-supervised multi-view clustering optimization problem based on the maximum correntropy criterion is formulated as follows:

$$\begin{aligned}\max_{\mathbf{W}^v,\mathbf{M},\mathbf{Z}^v,\mathbf{S}} & \frac{1}{2}\sum_{v=1}^{V}\sum_{i=1}^{n}\exp\left(-\frac{||\mathbf{x}_i^v-\mathbf{W}^v\mathbf{M}\mathbf{a}_i||_2^2}{2\sigma_1^2}\right)+\frac{\alpha}{2}\sum_{v=1}^{V}\sum_{i=1}^{n}\exp\left(-\frac{||\mathbf{X}^v\mathbf{z}_i^v-\mathbf{x}_i^v||_2^2}{2\sigma_2^2}\right)\\ &+\frac{\alpha\lambda}{2}\sum_{i=1}^{n}\exp\left(-\frac{||\mathbf{M}\mathbf{A}\mathbf{s}_i-\mathbf{M}\mathbf{a}_i||_2^2}{2\sigma_3^2}\right)+\frac{\alpha\gamma}{2}\sum_{v=1}^{V}\exp\left(-\frac{\|\mathbf{Z}^v-\mathbf{S}\|_F^2}{2\sigma_4^2}\right)\\ &-\rho\|\mathcal{Z}\|_{S_p}^p-\frac{\alpha\beta}{2}\|\mathbf{S}\|_1\\ \text{s.t. } & (\mathbf{W}^v)^\top\mathbf{W}^v=\mathbf{I},\ \mathbf{M}\geq 0,\ \mathbf{S}\geq 0,\ s_{i,i}=0,\ \forall i=1,\ldots,n,\end{aligned}\tag{A1}$$

Directly optimizing the objective with exponential kernel terms is inconvenient for gradient derivation and block coordinate update. According to the Fenchel conjugate property introduced in Section IV-A, we introduce adaptive weight variables to equivalently transform each correntropy term into a weighted quadratic form, where the weights are adaptively determined by the corresponding reconstruction errors. After the equivalent reformulation, the objective can be rewritten in the compact trace form:

$$\begin{aligned}\max_{\mathbf{W}^v,\mathbf{M},\mathbf{Z}^v,\mathbf{S}} & \frac{1}{2}\sum_{v=1}^{V}\mathrm{Tr}(\mathbf{X}^v-\mathbf{W}^v\mathbf{M}\mathbf{A})\mathbf{P}^v(\mathbf{X}^v-\mathbf{W}^v\mathbf{M}\mathbf{A})^\top+\frac{\alpha}{2}\sum_{v=1}^{V}\mathrm{Tr}(\mathbf{X}^v\mathbf{Z}^v-\mathbf{X}^v)\mathbf{B}^v(\mathbf{X}^v\mathbf{Z}^v-\mathbf{X}^v)^\top\\ &+\frac{\alpha\lambda}{2}\mathrm{Tr}(\mathbf{M}\mathbf{A}\mathbf{S}-\mathbf{M}\mathbf{A})\mathbf{C}(\mathbf{M}\mathbf{A}\mathbf{S}-\mathbf{M}\mathbf{A})^\top+\frac{\alpha\gamma}{2}\sum_{v=1}^{V}d^v(\mathbf{Z}^v-\mathbf{S})(\mathbf{Z}^v-\mathbf{S})^\top\\ &-\rho\|\mathcal{Z}\|_{S_p}^p-\frac{\alpha\beta}{2}\|\mathbf{S}\|_1\\ \text{s.t. } & \mathbf{W}^\top\mathbf{W}^v=\mathbf{I},\ \mathbf{M}\geq 0,\ \mathbf{S}\geq 0,\ s_{i,i}=0,\ \forall i=1,\ldots,n,\end{aligned}\tag{A2}$$

Appendix A.2. The Derivation of $L^t_{\mathbf{Z}^v}$

From the definition of f in (21e), $\nabla_{\mathbf{Z}^v} f$ can be computed as

$$\nabla_{\mathbf{Z}^v} f = \alpha\Big[\mathbf{X}^{v\top}\mathbf{X}^v(\mathbf{Z}^v - \mathbf{I})\mathbf{B}^v + \gamma d^v(\mathbf{Z}^v - \mathbf{S})\Big]. \tag{A3}$$

Since $d^v = -\exp\big(-\|\mathbf{Z}^v - \mathbf{S}\|_F^2/2\sigma_4^2\big) \in [-1, 0)$, it is obvious that $|d^v| \leq 1$. Therefore, we can bound the term as follows. For any $\mathbf{Z}_1^v$ and $\mathbf{Z}_2^v$, we have

$$\begin{aligned}
&\|\nabla_{\mathbf{Z}^v} f(\mathbf{Z}_1^v) - \nabla_{\mathbf{Z}^v} f(\mathbf{Z}_2^v)\|_F \\
=&\alpha\|\mathbf{X}^{v\top}\mathbf{X}^v(\mathbf{Z}_1^v - \mathbf{Z}_2^v)\mathbf{B}^v + \gamma(\mathbf{Z}_1^v - \mathbf{Z}_2^v)\|_F \\
\leq&\alpha(\|\mathbf{X}^{v\top}\mathbf{X}^v\|_F\|\mathbf{B}^v\|_2 + \gamma)\|\mathbf{Z}_1^v - \mathbf{Z}_2^v\|_F.
\end{aligned}$$

Therefore, the Lipschitz constant

$$L^t_{\mathbf{Z}^v} = \alpha(\|\mathbf{X}^{v\top}\mathbf{X}^v\|_F\|\mathbf{B}^{v,t+1}\|_2 + \gamma). \tag{A4}$$

Appendix A.3. The Derivation of $L^t_{\mathbf{M}}$

From the definition of f in (21e), $\nabla_{\mathbf{M}} f$ can be computed as

$$\nabla_{\mathbf{M}} f = \alpha\lambda\mathbf{M}(\mathbf{A} - \mathbf{AS})\mathbf{C}(\mathbf{A} - \mathbf{AS})^\top + \sum_{v=1}^{V}(\mathbf{W}^{v\top}\mathbf{W}^v\mathbf{MAP}^v\mathbf{A}^\top - \mathbf{W}^{v\top}\mathbf{X}^v\mathbf{P}^v\mathbf{A}^\top). \tag{A5}$$

For any $\mathbf{M}_1$ and $\mathbf{M}_2$, we have

$$\begin{aligned}
&\|\nabla_{\mathbf{M}} f(\mathbf{M}_1) - \nabla_{\mathbf{M}} f(\mathbf{M}_2)\|_F \\
=&\|\alpha\lambda(\mathbf{M}_1 - \mathbf{M}_2)(\mathbf{A} - \mathbf{AS})\mathbf{C}(\mathbf{A} - \mathbf{AS})^\top + \sum_{v=1}^{V}(\mathbf{W}^v)^\top\mathbf{W}^v(\mathbf{M}_1 - \mathbf{M}_2)\mathbf{AP}^v\mathbf{A}^\top\|_F \\
\leq&\alpha\lambda\|(\mathbf{A} - \mathbf{AS})\mathbf{C}(\mathbf{A} - \mathbf{AS})^\top\|_F\|\mathbf{M}_1 - \mathbf{M}_2\|_F + \sum_{v=1}^{V}\|\mathbf{AP}^v\mathbf{A}^\top\|_F\|\mathbf{M}_1 - \mathbf{M}_2\|_F \\
\leq&\|\mathbf{A}\|_F^2(\alpha\lambda\|\mathbf{C}\|_2\|\mathbf{I} - \mathbf{S}\|_F^2 + \sum_{v=1}^{V}\|\mathbf{P}^v\|_2)\|\mathbf{M}_1 - \mathbf{M}_2\|_F.
\end{aligned}$$

Therefore, the Lipschitz constant

$$L^t_{\mathbf{M}} = \|\mathbf{A}\|_F^2(\alpha\lambda\|\mathbf{C}\|_2\|\mathbf{I} - \mathbf{S}\|_F^2 + \sum_{v=1}^{V}\|\mathbf{P}^v\|_2). \tag{A6}$$

Appendix A.4. The Derivation of $L^t_{\mathbf{S}}$

From the definition of f in (21e), $\nabla_{\mathbf{S}} f$ can be computed as

$$\nabla_{\mathbf{S}} f = \alpha\left[\lambda\mathbf{A}^\top\mathbf{M}^\top\mathbf{MA}(\mathbf{S} - \mathbf{I})\mathbf{C} + \gamma\sum_{v=1}^{V} d^v(\mathbf{S} - \mathbf{Z}^v)\right]. \tag{A7}$$

For any $\mathbf{S}_1$ and $\mathbf{S}_2$, we have

$$\begin{aligned}
&\|\nabla_{\mathbf{S}} f(\mathbf{S}_1) - \nabla_{\mathbf{S}} f(\mathbf{S}_2)\|_F \\
=&\alpha\|\lambda\mathbf{A}^\top\mathbf{M}^\top\mathbf{MA}(\mathbf{S}_1 - \mathbf{S}_2)\mathbf{C} + \gamma\sum_{v=1}^{V} d^v(\mathbf{S}_1 - \mathbf{S}_2)\|_F \\
\leq&\alpha(\lambda\|\mathbf{A}^\top\mathbf{M}^\top\mathbf{MA}\|_F\|\mathbf{C}\|_2 + \gamma\|\mathbf{d}\|_1)\|\mathbf{S}_1 - \mathbf{S}_2\|_F.
\end{aligned}$$

Therefore, the Lipschitz constant

$$L^t_{\mathbf{S}} = \alpha(\lambda\|\mathbf{A}^\top\mathbf{M}^\top\mathbf{MA}\|_F\|\mathbf{C}\|_2 + \gamma\|\mathbf{d}\|_1). \tag{A8}$$

References

1. Zhou, N.; Jika, L.; Wang, W.; Xie, Y.; Du, Y.; Soh, Y.C. BERN: A Novel Framework for Enhanced Emotion Recognition Through the Integration of EEG and Eye Movement Features. *IEEE Trans. Cogn. Dev. Syst.* **2026**, *18*, 263–275. https://doi.org/10.1109/TCDS.2025.3590031.
2. Xie, Y.; Ou, J.; Wen, B.; Yu, Z.; Tian, W. A joint learning method for low-light facial expression recognition. *Complex Intell. Syst.* **2025**, *11*, 139.
3. Liu, J.; Liu, X.; Yang, Y.; Guo, X.; Kloft, M.; He, L. Multiview subspace clustering via co-training robust data representation. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**, *33*, 5177–5189.
4. Xu, H.; Zhang, X.; Xia, W.; Gao, Q.; Gao, X. Low-rank tensor constrained co-regularized multi-view spectral clustering. *Neural Netw.* **2020**, *132*, 245–252.
5. Li, Z.; Tang, C.; Liu, X.; Zheng, X.; Zhang, W.; Zhu, E. Consensus graph learning for multi-view clustering. *IEEE Trans. Multimed.* **2021**, *24*, 2461–2472.
6. Li, X.; Zhang, H.; Wang, R.; Nie, F. Multiview clustering: A scalable and parameter-free bipartite graph fusion method. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *44*, 330–344.
7. Wang, S.; Liu, X.; Zhu, X.; Zhang, P.; Zhang, Y.; Gao, F.; Zhu, E. Fast parameter-free multi-view subspace clustering with consensus anchor guidance. *IEEE Trans. Image Process.* **2021**, *31*, 556–568.
8. Xiao, Q.; Du, S.; Song, J.; Yu, Y.; Huang, Y. Hyper-Laplacian regularized multi-view subspace clustering with a new weighted tensor nuclear norm. *IEEE Access* **2021**, *9*, 118851–118860.
9. Ma, L.; Zhou, N.; Du, Y.; Wang, W.; Shi, K.; Pedrycz, W. Low-Rank Matrix Factorization Induced Adaptive Divergent Graph Learning for Fuzzy Clustering. *IEEE Trans. Fuzzy Syst.* **2026**, *34*, 705–718. https://doi.org/10.1109/TFUZZ.2025.3643471.
10. Sun, M.; Wang, S.; Zhang, P.; Liu, X.; Guo, X.; Zhou, S.; Zhu, E. Projective multiple kernel subspace clustering. *IEEE Trans. Multimed.* **2021**, *24*, 2567–2579.
11. Weng, W.; Zhou, W.; Chen, J.; Peng, H.; Cai, H. Enhancing multi-view clustering through common subspace integration by considering both global similarities and local structures. *Neurocomputing* **2020**, *378*, 375–386.
12. Ke, G.; Hong, Z.; Yu, W.; Zhang, X.; Liu, Z. Efficient multi-view clustering networks. *Appl. Intell.* **2022**, *52*, 14918–14934.
13. Xia, W.; Wang, S.; Yang, M.; Gao, Q.; Han, J.; Gao, X. Multi-view graph embedding clustering network: Joint self-supervision and block diagonal representation. *Neural Netw.* **2022**, *145*, 1–9.
14. Lv, J.; Kang, Z.; Wang, B.; Ji, L.; Xu, Z. Multi-view subspace clustering via partition fusion. *Inf. Sci.* **2021**, *560*, 410–423.
15. Tang, C.; Liu, X.; Zhu, X.; Zhu, E.; Luo, Z.; Wang, L.; Gao, W. CGD: Multi-view clustering via cross-view graph diffusion. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; Volume 34, pp. 5924–5931.
16. Zheng, Y.; Zhang, X.; Xu, Y.; Qin, M.; Ren, Z.; Xue, X. Robust multi-view subspace clustering via weighted multi-kernel learning and co-regularization. *IEEE Access* **2020**, *8*, 113030–113041.
17. Fu, Y.; Hu, D.; Wang, Z. Nonconvex Low-Rank Tensor Representation with Deep Priors for Multiview Subspace Clustering. In Proceedings of the Forty-third International Conference on Machine Learning, Seoul, Republic of Korea, 6–11 July 2026.
18. Zhou, N.; Deng, Q.; Luo, W.; Huang, X.; Du, Y.; Chen, B.; Pedrycz, W. Correntropy meets cross-entropy: A robust loss against noisy labels. *Eng. Appl. Artif. Intell.* **2026**, *167*, 113830.
19. Zhou, N.; Luo, W.; Wu, Z.; Du, Y.; Shi, K.; Chen, B. Bipartite graph regularized robust low-rank matrix factorization for fast semi-supervised image clustering. *Appl. Intell.* **2025**, *56*, 22. https://doi.org/10.1007/s10489-025-07040-w.
20. Xin, X.; Wang, J.; Xie, R.; Zhou, S.; Huang, W.; Zheng, N. Semi-supervised person re-identification using multi-view clustering. *Pattern Recognit.* **2019**, *88*, 285–297.
21. Nie, F.; Cai, G.; Li, X. Multi-view clustering and semi-supervised classification with adaptive neighbours. In Proceedings of the AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017; Volume 31, pp. 2408–2414.
22. Cai, H.; Liu, B.; Xiao, Y.; Lin, L. Semi-supervised multi-view clustering based on constrained nonnegative matrix factorization. *Knowl.-Based Syst.* **2019**, *182*, 104798.
23. Jiang, G.; Peng, J.; Wang, H.; Mi, Z.; Fu, X. Tensorial multi-view clustering via low-rank constrained high-order graph learning. *IEEE Trans. Circuits Syst. Video Technol.* **2022**, *32*, 5307–5318.
24. Yu, Z.; Yu, C.; Yang, K.; Chen, X.; Shi, Y.; Philip Chen, C.L. Tensor-Based Semi-Supervised Multiview Subspace Clustering With Pairwise Constraint Propagation. *IEEE Trans. Cybern.* **2026**, *56*, 867–879. https://doi.org/10.1109/TCYB.2025.3607287.
25. Yu, Y.; She, K.; Shi, K.; Cai, X.; Kwon, O.M.; Soh, Y. Analysis of medical images super-resolution via a wavelet pyramid recursive neural network constrained by wavelet energy entropy. *Neural Netw.* **2024**, *178*, 106460. https://doi.org/10.1016/j.neunet.2024.106460.
26. Kilmer, M.E.; Martin, C.D. Factorization strategies for third-order tensors. *Linear Algebra Its Appl.* **2011**, *435*, 641–658.
27. Martinez, A.M. Recognizing imprecisely localized, partially occluded, and expression variant faces from a single sample per class. *IEEE Trans. Pattern Anal. Mach. Intell.* **2002**, *24*, 748–763.
28. Fidler, S.; Skocaj, D.; Leonardis, A. Combining reconstructive and discriminative subspace methods for robust classification and regression by subsampling. *IEEE Trans. Pattern Anal. Mach. Intell.* **2006**, *28*, 337–350.

29. Wright, J.; Yang, A.Y.; Ganesh, A.; Sastry, S.S.; Ma, Y. Robust face recognition via sparse representation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2008**, *31*, 210–227.
30. Tang, H.; Li, H. Information Theoretic Learning: Reny's Entropy and Kernel Perspectives (Principe, J.; 2010)[Book Review]. *IEEE Comput. Intell. Mag.* **2011**, *6*, 60–62.
31. Liu, W.; Pokharel, P.P.; Principe, J.C. Correntropy: Properties and applications in non-Gaussian signal processing. *IEEE Trans. Signal Process.* **2007**, *55*, 5286–5298.
32. Pokharel, P.P.; Liu, W.; Principe, J.C. A low complexity robust detector in impulsive noise. *Signal Process.* **2009**, *89*, 1902–1909.
33. Principe, J.C.; Xu, D. An introduction to information theoretic learning. In *Proceedings of the IJCNN'99. International Joint Conference on Neural Networks. Proceedings (Cat. No. 99CH36339)*; IEEE: Piscataway, NJ, USA, 1999; Volume 3, pp. 1783–1787.
34. Vapnik, V.N. *The Nature of Statistical Learning Theory*; Springer: New York, NY, USA, 1995.
35. Wang, H.; Yang, Y.; Liu, B. GMC: Graph-based multi-view clustering. *IEEE Trans. Knowl. Data Eng.* **2019**, *32*, 1116–1129.
36. Zhang, C.; Fu, H.; Liu, S.; Liu, G.; Cao, X. Low-rank tensor constrained multiview subspace clustering. In *Proceedings of the IEEE International Conference on Computer Vision*; IEEE: Piscataway, NJ, USA, 2015; pp. 1582–1590.
37. Yang, Z.; Zhang, H.; Liang, N.; Li, Z.; Sun, W. Semi-supervised multi-view clustering by label relaxation based non-negative matrix factorization. *Vis. Comput.* **2023**, *39*, 1409–1422.
38. Wang, Z.; Hu, D.; Liu, Z.; Gao, C.; Wang, Z. Iteratively Capped Reweighting Norm Minimization with Global Convergence Guarantee for Low-Rank Matrix Learning. *IEEE Trans. Pattern Anal. Mach. Intell.* **2024**, *47*, 1923–1940. https://doi.org/10.1109/TPAMI.2024.3512458.
39. Hu, D.; Qu, Q.; Liu, Z.; Chen, W.; Wang, Z. Subspace clustering based on latent low-rank representation with transformed Schatten-1 penalty function. *Knowl.-Based Syst.* **2024**, *304*, 112538. https://doi.org/10.1016/j.knosys.2024.112538.
40. Fu, Y.; Wang, Z.; Hu, D.; Jia, T.; Gao, C.; Wang, Z. A Unified Framework With Capped Tensor Norm Minimization for Multiview Subspace Learning. *IEEE Trans. Knowl. Data Eng.* **2026**, *38*, 5607–5621. https://doi.org/10.1109/TKDE.2026.3694749.
41. Rockafellar, R.T. *Convex Analysis*; Princeton University Press: Princeton, NJ, USA, 1997; Volume 11.
42. Xu, Y.; Yin, W. A Block Coordinate Descent Method for Regularized Multiconvex Optimization with Applications to Nonnegative Tensor Factorization and Completion. *SIAM J. Imaging Sci.* **2013**, *6*, 1758–1789. https://doi.org/10.1137/120887795.
43. Chang, W.; Chen, H.; Nie, F.; Wang, R.; Li, X. Tensorized and Compressed Multi-View Subspace Clustering via Structured Constraint. *IEEE Trans. Pattern Anal. Mach. Intell.* **2024**, *46*, 10434–10451. https://doi.org/10.1109/TPAMI.2024.3446537.
44. Nene, S.A.; Nayar, S.K.; Murase, H. Columbia Object Image Library (COIL-20); Technical Report CUCS-005-96; Columbia University: New York, NY, USA, 1996.
45. Nie, F.; Cai, G.; Li, J.; Li, X. Auto-Weighted Multi-View Learning for Image Clustering and Semi-Supervised Classification. *IEEE Trans. Image Process.* **2018**, *27*, 1501–1511. https://doi.org/10.1109/TIP.2017.2754939.
46. Wang, S.; Gu, X.; Lu, J.; Yang, J.Y.; Wang, R.; Yang, J. Unsupervised discriminant canonical correlation analysis for feature fusion. In *Proceedings of the 2014 22nd International Conference on Pattern Recognition*; IEEE: Piscataway, NJ, USA, 2014; pp. 1550–1555.
47. Chen, S.; Zhao, H.; Kong, M.; Luo, B. 2D-LPP: A two-dimensional extension of locality preserving projections. *Neurocomputing* **2007**, *70*, 912–921.
48. Wang, Z.; Liu, Z.; Hu, D.; Jia, T. Nonconvex multiview subspace clustering framework with efficient method designs and theoretical analysis. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence (IJCAI-24)*, Jeju, Republic of Korea, 3–9 August 2024; International Joint Conferences on Artificial Intelligence: Jeju, Republic of Korea, 2024; pp. 5162–5170. https://doi.org/10.24963/ijcai.2024/571.
49. Yu, Y.; Zhou, G.; Huang, H.; Xie, S.; Zhao, Q. Multi-view Data Classification with a Label-driven Auto-weighted Strategy. *arXiv* **2022**, arXiv:2201.00714. https://doi.org/10.48550/arXiv.2201.00714.
50. Liang, N.; Yang, Z.; Li, Z.; Xie, S.; Su, C.Y. Semi-supervised multi-view clustering with Graph-regularized Partially Shared Non-negative Matrix Factorization. *Knowl.-Based Syst.* **2020**, *190*, 105185. https://doi.org/10.1016/j.knosys.2019.105185.
51. Peng, S.; Yin, J.; Yang, Z.; Chen, B.; Lin, Z. Multiview Clustering via Hypergraph Induced Semi-Supervised Symmetric Nonnegative Matrix Factorization. *IEEE Trans. Circuits Syst. Video Technol.* **2023**, *33*, 5510–5524. https://doi.org/10.1109/TCSVT.2023.3258926.
52. Huang, A.; Wang, Z.; Zheng, Y.; Zhao, T.; Lin, C.W. Embedding regularizer learning for multi-view semi-supervised classification. *IEEE Trans. Image Process.* **2021**, *30*, 6997–7011.
53. Ling, H.; Bao, C.; Song, J.; Shi, Z. Fast and Scalable Semi-Supervised Learning for Multi-View Subspace Clustering. *arXiv* **2024**, arXiv:2408.05707. https://doi.org/10.48550/arXiv.2408.05707.
54. Shi, H.J.M.; Tu, S.; Xu, Y.; Yin, W. A primer on coordinate descent algorithms. *arXiv* **2016**, arXiv:1610.00040.
55. Hubert, L.; Arabie, P. Comparing partitions. *J. Classif.* **1985**, *2*, 193–218.