# ATMSC-Net: Deep Unfolding Network for Anchor-based Tensorial Multi-View Subspace Clustering


1st Tianhao Huang
Jiangnan University
Wuxi, China
6233112019@jiangnan.edu.cn

2nd Xiaojun Wu
Jiangnan University
Wuxi, China
wu_xiaojun@jiangnan.edu.cn

3rd Wenhua Dong
Jiangnan University
Wuxi, China
wenhua_dong_jnu@jiangnan.edu.cn

4th Tianyang Xu
Jiangnan University
Wuxi, China
tianyang_xu@163.com



***Abstract*—Anchor-based and tensor-based approaches have emerged as two promising directions for multi-view subspace clustering, offering linear-complexity scalability and high-order correlation modeling, respectively. However, existing deep methods typically incorporate these structures in a heuristic or task-agnostic manner, lacking principled integration with the underlying clustering optimization and failing to combine both paradigms effectively. To bridge this gap, we propose ATMSC-Net, a deep network architecture that systematically unfolds the ADMM iterative solution of anchor-based tensorial multi-view subspace clustering into principled network structures. The proposed model decomposes the clustering process into four interpretable modules, RepresentModule, NoiseModule, TensorModule, and AnchorModule, each derived by unfolding a specific optimization subproblem into a dedicated network component, providing structural clarity and optimization traceability. By introducing TensorModule into the deep unfolding framework, ATMSC-Net extends the joint anchor-tensor paradigm from shallow methods to deep learning. Additionally, hand-crafted hyperparameters are transformed into learnable parameters through end-to-end training, enabling automatic adaptation to diverse data distributions. Extensive experiments on multiple datasets demonstrate that ATMSC-Net achieves competitive clustering performance compared with state-of-the-art methods.**



***Index Terms*—multi-view, unfolding network, tensor learning, subspace clustering**


## I. Introduction

Multi-view data, acquired from diverse sources or feature extractors, has become ubiquitous in real-world applications such as image analysis, social network mining, and medical diagnosis [1]–[3]. Multi-view clustering seeks to partition such data into meaningful groups by leveraging both consistency and complementarity across views [4], [5]. Among various approaches, subspace clustering has attracted considerable attention, which assumes that data points lie in a union of low-dimensional subspaces and employs self-representation to reveal the underlying structure [6], [7]. A central challenge in multi-view subspace clustering lies in jointly modeling heterogeneous feature distributions while preserving global structural consistency, all in an unsupervised manner.

To improve computational efficiency for large-scale scenarios, anchor-based methods have been developed, which approximate sample-level relationships using a compact set of representative anchors [8]–[10]. These methods construct anchor graphs to capture local structure and significantly reduce computational overhead. However, since anchor-based approaches typically focus on pairwise relationships within or between views, they struggle to capture high-order correlations among all views simultaneously. To address this limitation, tensorial multi-view clustering (TMC) methods have emerged, which stack the representation matrices from different views into a third-order tensor and impose low-rank constraints for consistent structure discovery [11]–[13]. Recently, Ji et al. [14] successfully integrated anchor-based representation with tensor low-rank constraints, achieving both computational efficiency and high-order correlation modeling. Despite their effectiveness, these shallow methods depend on manually tuned hyperparameters and are inherently limited in extracting deep feature representations.

To overcome the limitations of shallow methods, deep multi-view clustering approaches have been proposed, which leverage neural networks to learn latent embeddings through joint feature transformation and alignment [15]–[19]. These methods employ architectures such as autoencoders, graph neural networks, and contrastive learning, offering enhanced modeling flexibility and the capability to capture complex non-linear relationships. Along this line, anchor-based deep methods [20], [21] have been developed to combine learning capability with computational efficiency; meanwhile, tensor-based deep methods [22], [23] incorporate tensor learning into neural architectures for high-order modeling. Nevertheless, these deep approaches typically adopt heuristic network designs without explicit correspondence to the underlying clustering objective, limiting the interpretability of the learned representations. Moreover, while the integration of anchor representation and tensor learning has achieved promising results in shallow methods, this paradigm remains unexplored in deep multi-view subspace clustering.

To bridge this gap, we propose ATMSC-Net, a deep unfolding network that integrates anchor-based representation learning with low-rank tensor constraints for multi-view subspace clustering. Starting from an ADMM-based optimization formulation, we systematically unfold each iterative step into a

differentiable network layer with learnable parameters. Specifically, **RepresentModule** replaces the fixed linear projection with a parameterized layer and applies soft-thresholding for sparse anchor representation. **NoiseModule** performs column-wise shrinkage with an adaptive threshold for noise separation. **TensorModule** enforces low-rank tensor structure through differentiable singular value thresholding with a learnable threshold. **AnchorModule** maintains orthogonal anchor alignment via SVD-based projection. This principled design establishes explicit correspondence between network components and optimization subproblems, while converting hand-crafted hyperparameters into trainable parameters for automatic adaptation. The main contributions of this paper are summarized as follows:

- The designed ATMSC-Net systematically maps ADMM iterations to interpretable modules, and transforms hand-crafted hyper-parameters into learnable structures.
- The proposed deep anchor-tensor paradigm offers linear-complexity scalability and high-order correlation modeling capability.
- Extensive experiments on multiple multi-view benchmarks demonstrate that ATMSC-Net achieves competitive clustering performance compared with state-of-the-art shallow and deep methods.

## II. Related Work

### A. Multi-view Subspace Clustering

Multi-view subspace clustering has attracted considerable attention due to its ability to reveal the underlying structure of high-dimensional data. The core idea is based on the self-representation property, which assumes that each data point can be represented as a linear combination of other points within the same subspace. Given multi-view data $\{X^v \in \mathbb{R}^{n\times d_v}\}_{v=1}^m$ with $n$ samples and $m$ views, the general framework of multi-view subspace clustering can be formulated as:

$$\min_{\{Z^v\},\{E^v\}} \sum_{v=1}^{m} \|E^v\|_p + \lambda \mathcal{R}(\{Z^v\}_{v=1}^m) \quad s.t.\ X^v = Z^v X^v + E^v, \tag{1}$$

where $Z^v \in \mathbb{R}^{n\times n}$ is the self-representation matrix of the $v$-th view, $E^v$ denotes the reconstruction error, and $\mathcal{R}(\cdot)$ represents regularization terms imposed on the representation matrices. Early methods typically constrain representation matrices within individual views or between pairs of views through co-regularization strategies [24], which limits their ability to capture global structural information across all views.

To address this limitation, tensorial multi-view clustering (TMC) methods have emerged, which stack the representation matrices from different views into a third-order tensor and impose low-rank constraints for consistent structure discovery. The general framework can be expressed as:

$$\begin{aligned} &\min_{\mathcal{Z},\{E^v\}} \quad \|\mathcal{Z}\|_{TNN} + \lambda \sum_{v=1}^{m} \|E^v\|_p \\ &s.t. \quad X^v = Z^v X^v + E^v, \quad \mathcal{Z} = \Phi(Z^1, \ldots, Z^m), \end{aligned} \tag{2}$$

TABLE I
Essential notations and descriptions.

| Notation | Description |
|---|---|
| $\mathbb{R}$ | The real number space. |
| $n, m$ | The number of samples and views. |
| $d_v$ | The dimension of the $v$-th view. |
| $t, l$ | The anchor and projection dimensions. |
| $c$ | The number of clusters. |
| $X^v \in \mathbb{R}^{n\times d_v}$ | The $v$-th view matrix. |
| $Z^v_{anc} \in \mathbb{R}^{n\times t}$ | The $v$-th anchor representation. |
| $E^v \in \mathbb{R}^{n\times d_v}$ | The $v$-th noise matrix. |
| $P^v \in \mathbb{R}^{l\times d_v}$ | The $v$-th anchor indicator. |
| $A \in \mathbb{R}^{t\times l}$ | The shared anchor dictionary. |
| $\mathcal{Z}_{anc}$ | Tensor stacked from $\{Z^v_{anc}\}_{v=1}^m$. |

where $\mathcal{Z} \in \mathbb{R}^{n\times m\times n}$ is the representation tensor, $\Phi(\cdot)$ denotes the stacking and rotation operation, and $\|\cdot\|_{TNN}$ is the tensor nuclear norm based on tensor singular value decomposition (t-SVD). The low-rank tensor constraint serves as a view-level regularization that enforces consistent block-diagonal structures across all views, enabling the capture of high-order correlations that matrix-level fusion cannot achieve. Various tensor rank surrogates have been proposed, from convex relaxations based on t-SVD [12], [25] to non-convex alternatives that mitigate over-penalization of large singular values [26].

Despite the effectiveness of TMC methods in capturing high-order correlations, they incur high computational cost due to $\mathcal{O}(n^2)$ storage complexity for representation matrices and $\mathcal{O}(n^3)$ computational complexity for tensor operations, making them impractical for large-scale datasets. To overcome this limitation, recent works [14], [27] integrate anchor representation learning with low-rank tensor learning into a unified framework. Instead of constructing sample-level self-representation matrices, anchor-based methods learn a compact set of representative anchors to characterize all samples:

$$X^v = Z^v_{anc} A P^v + E^v, \tag{3}$$

where $A \in \mathbb{R}^{t\times l}$ is the shared anchor dictionary in a unified space, $P^v \in \mathbb{R}^{l\times d_v}$ is the view-specific projection matrix, and $Z^v_{anc} \in \mathbb{R}^{n\times t}$ is the anchor representation matrix with $t \ll n$. By stacking anchor representations into a space-saving tensor $\mathcal{Z}_{anc} \in \mathbb{R}^{t\times m\times n}$, the computational complexity is significantly reduced from $O(n^2 \log n)$ to $O(nmt \log(nm))$, enabling efficient processing of large-scale multi-view data.

However, despite their effectiveness, these shallow methods rely on manually tuned hyperparameters that require careful adjustment for different datasets. The iterative optimization procedures lack adaptability to varying data distributions, limiting their practical applicability.

### B. Deep Multi-view Clustering

To overcome the limitations of shallow methods, deep multi-view clustering approaches leverage neural networks for learning expressive latent representations. Zhu et al. [28] jointly learn view-specific and shared self-representation matrices through diversity and universality sub-networks in an end-to-end manner. To improve scalability, Cui et al. [20] combine anchor graphs with spectral clustering-based self-supervised learning and perform contrastive learning on pseudo-labels

TABLE II
TIME AND SPACE COMPLEXITY COMPARISON. $L$ IS THE NUMBER OF UNFOLDING LAYERS, $s$ IS THE MAX HIDDEN NEURONS, $p$ IS THE GRANULARITY PARAMETER, $V$ AND $D$ ARE INTERMEDIATE DIMENSIONS.

| Method | Time Complexity | Space Complexity |
|---|---|---|
| EOMSC-CA (AAAI'22) [29] | $O(n)$ | $O((d+t)n)$ |
| t-SVD-MSC (IJCV'18) [12] | $O(n^2 \log n)$ | $O(n^2)$ |
| TLS$_p$NM-MSC (TPAMI'23) [26] | $O(n^2 \log n)$ | $O(n^2)$ |
| TBGL (TPAMI'23) [30] | $O(n \log n)$ | $O(n^2)$ |
| ARLRR-TU (TPAMI'23) [31] | $O(n^2 \log n)$ | $O(n^2)$ |
| ESTMC-C$^2$ (TPAMI'25) [14] | $O(nmt \log(nm) + nd)$ | $O((d+mt)n)$ |
| DMCAG (IJCAI'23) [20] | $O(L(n(sD+msV+m^2+cV^2)))$ | $O(Vn(d+t))$ |
| SparseMvC (NeurIPS'25) [32] | $O(L \cdot Vn^2)$ | $O(n^2 + nd)$ |
| MGBCC (AAAI'25) [33] | $O(L \cdot V^2 n^2 d/p^2)$ | $O(V^2 n^2/p^2)$ |
| LargeMvC-Net (MM'25) [34] | $O(L(n(tD+t^2V)+t^2D))$ | $O(n(t+D)+tD+t^2V)$ |
| ATMSC-Net (Ours) | $O(L(nmt \log(nm) + nd))$ | $O((d+mt)n)$ |

to maintain cross-view consistency. Wang et al. [21] achieve end-to-end learnable anchors through a perturbation-driven mechanism and employ anchor graph convolution for linear-complexity clustering. For high-order correlation modeling, Liu and Song [22] integrate deep non-negative matrix factorization with weighted low-rank tensor constraints to capture complex nonlinear relationships in deep embedding space. Wang et al. [23] employ tensor transformers for cross-view information interaction and use self-expression coefficients to adaptively select contrastive sample pairs. However, the design of these methods is primarily driven by empirical observations rather than derived from clustering optimization formulations.

### C. Deep Unfolding Networks

Deep unfolding provides a principled framework that bridges optimization-based methods and deep learning by mapping iterative algorithms into feed-forward network architectures [35]. Gregor and LeCun [36] first introduced this paradigm by proposing LISTA, which unrolls ISTA for sparse coding. Unlike purely data-driven networks, deep unfolding preserves the interpretability of optimization methods, where each layer corresponds to one iteration with explicit mathematical semantics, and hand-crafted parameters become learnable through end-to-end training.

Deep unfolding has been successfully applied to various computer vision tasks such as compressive sensing and image restoration [37]–[40]. For subspace clustering, Li et al. [41] unfold ADMM with structure-preserving constraints and combine autoencoder-based feature extraction with interpretable optimization layers. Xie et al. [42] reformulate linearized iterations as a multi-block network for end-to-end learning of dictionary and affinity matrices. For multi-view clustering, LargeMvC-Net [34] unrolls anchor-based optimization into three interpretable modules with linear complexity for large-scale scenarios. Inspired by these advances, we introduce the joint anchor-tensor modeling into deep learning through principled ADMM unfolding.

## III. METHOD

The necessary notations are listed in Table I. To enable scalable multi-view subspace clustering with high-order correlation modeling, we adopt an anchor-based tensorial formulation that combines computational efficiency with cross-view structural consistency.

### A. Problem Formulation and Optimization

**Optimization Foundation for Deep Unfolding:** Let $\{X^v \in \mathbb{R}^{n\times d_v}\}_{v=1}^m$ denote the input multi-view data, where $n$ is the number of samples and $d_v$ is the feature dimensionality of the $v$-th view. Our goal is to learn anchor-based representations $\{Z_{anc}^v\}_{v=1}^m$ that capture both view-specific characteristics and cross-view consistency. Each view is reconstructed through a shared anchor dictionary $A \in \mathbb{R}^{t\times l}$ and view-specific anchor indicators $P^v \in \mathbb{R}^{l\times d_v}$. To capture high-order correlations among views, the representations are stacked into a third-order tensor $\mathcal{Z}_{anc} = \Phi(Z_{anc}^1, \ldots, Z_{anc}^m) \in \mathbb{R}^{t\times m\times n}$, upon which a tensor nuclear norm (TNN) constraint is imposed. The optimization problem is formulated as:

$$\begin{aligned} \min_{Z_{anc}^v, A, P^v, \mathcal{G}, E^v} \ & \gamma\|\mathcal{G}\|_{TNN} + \lambda\|E\|_{2,1} + \alpha\sum\nolimits_{v=1}^m \|Z_{anc}^v\|_1 \\ s.t.\ & X^v = Z_{anc}^v A P^v + E^v,\ \mathcal{Z}_{anc} = \mathcal{G}, \\ & E = [E^1; \ldots; E^m]^T,\ A^T A = I,\ (P^v)^T P^v = I, \end{aligned} \tag{4}$$

where $\|\mathcal{G}\|_{TNN}$ is the tensor nuclear norm defined as $\|\mathcal{G}\|_{TNN} = \frac{1}{n}\sum_{k=1}^n \|\bar{\mathcal{G}}^{(k)}\|_*$ with $\bar{\mathcal{G}} = \text{FFT}(\mathcal{G}, [], 3)$ being the Fourier transform along the third dimension. The first term enforces low-rank structure across views to capture high-order correlations. The $\ell_{2,1}$-norm on $E$ encourages column-sparse noise patterns to handle sample-level corruptions. The $\ell_1$-norm on $Z_{anc}^v$ promotes sparse anchor representations for enhanced discriminability. The orthogonality constraints on $A$ and $P^v$ stabilize the anchor alignment and avoid trivial solutions.

To solve Problem (4), we employ ADMM. Let $R^v = X^v - Z_{anc}^v A P^v - E^v$ denote the reconstruction residual. By introducing the auxiliary tensor variable $\mathcal{G}$, the augmented Lagrangian function is:

$$\begin{aligned} \mathcal{L} = & \gamma\|\mathcal{G}\|_{TNN} + \lambda\|E\|_{2,1} + \alpha\sum_{v=1}^m \|Z_{anc}^v\|_1 \\ & + \sum_{v=1}^m \left[\langle Y^v, R^v\rangle + \frac{\mu}{2}\|R^v\|_F^2\right] \\ & + \langle W, \mathcal{Z}_{anc} - \mathcal{G}\rangle + \frac{\rho}{2}\|\mathcal{Z}_{anc} - \mathcal{G}\|_F^2, \end{aligned} \tag{5}$$

where $Y^v$ and $W$ are Lagrangian multipliers, and $\mu$, $\rho$ are penalty parameters. For simplicity, we set $\mu = \rho = 1$, which yields a more elegant structure after unfolding. The optimization proceeds by alternately updating each variable while keeping others fixed.

**1) Optimization with Respect to $Z_{anc}^v$:** Given fixed $A$, $P^v$, $E^v$, and $\mathcal{G}$, the subproblem for updating $Z_{anc}^v$ is an $\ell_1$-regularized least squares problem. By taking the derivative and applying the proximal operator for the $\ell_1$-norm, the closed-form solution is:

$$Z_{anc}^{v,*} = \mathcal{S}_{\frac{\alpha}{2}}\left(\frac{1}{2}\left[(X^v - E^v + Y^v)(P^v)^T A^T + G^v - W^v\right]\right), \tag{6}$$

where $G^v$ and $W^v$ denote the $v$-th slices of $\mathcal{G}$ and $W$, and $\mathcal{S}_\theta(x) = \text{sign}(x)\cdot\max(|x| - \theta, 0)$ is the soft-thresholding operator.

**2) Optimization with Respect to $E$:** Fixing other vari-

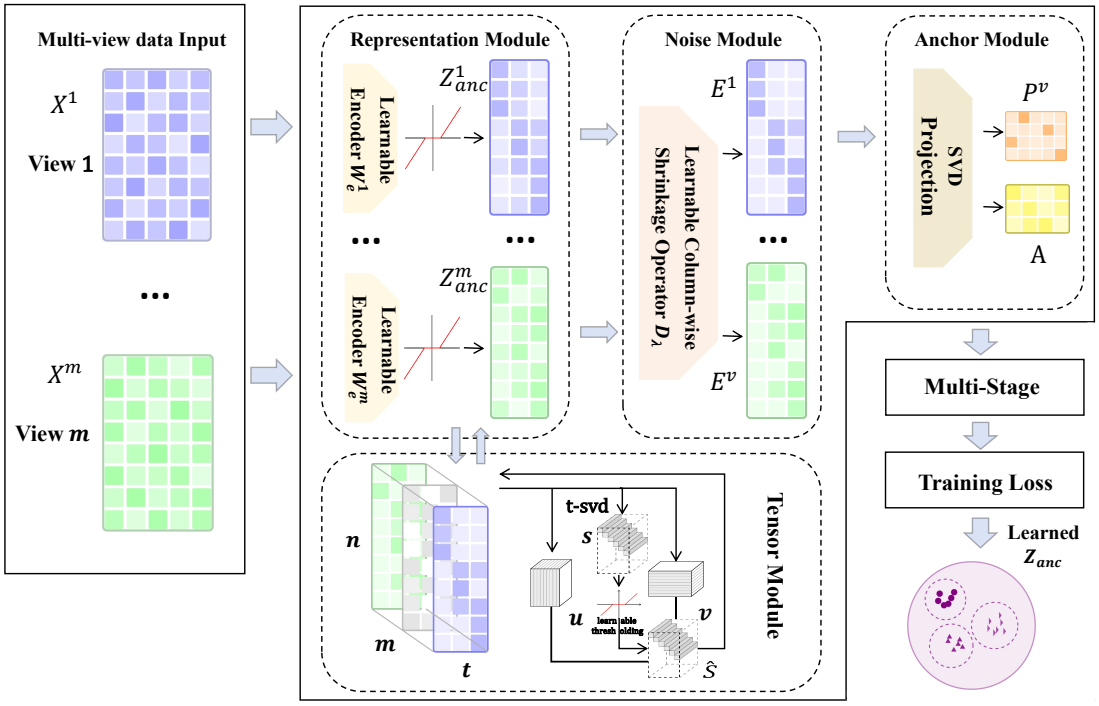



Fig. 1. The framework of ATMSC-Net. Each stage unfolds one ADMM iteration into four interpretable modules. The network repeats for $L$ stages, and the learned anchor representations $\{Z_{anc}^v\}_{v=1}^m$ are concatenated and clustered via $k$-means.

ables, the subproblem for $E$ is a standard $\ell_{2,1}$-norm proximal problem:

$$\min_E \lambda\|E\|_{2,1} + \frac{1}{2}\|E - \hat{E}\|_F^2, \tag{7}$$

where $\hat{E} = [X^1 - Z_{anc}^1 AP^1 + Y^1; \ldots; X^m - Z_{anc}^m AP^m + Y^m]^T$. This is solved via the column-wise shrinkage operator:

$$E_{:,j}^* = \begin{cases} \frac{\|\hat{E}_{:,j}\|_2 - \lambda}{\|\hat{E}_{:,j}\|_2} \cdot \hat{E}_{:,j}, & \text{if } \|\hat{E}_{:,j}\|_2 > \lambda \\ 0, & \text{otherwise.} \end{cases} \tag{8}$$

**3) Optimization with Respect to $\mathcal{G}$:** When other variables are fixed, the subproblem for $\mathcal{G}$ is:

$$\min_{\mathcal{G}} \gamma\|\mathcal{G}\|_{TNN} + \frac{1}{2}\|\mathcal{G} - (\mathcal{Z}_{anc} + W)\|_F^2. \tag{9}$$

Let $Q = \mathcal{Z}_{anc} + W$ with t-SVD $Q_f = \mathcal{U}_f * \mathcal{S}_f * \mathcal{V}_f^H$ in the Fourier domain. The optimal solution is given by the tensor singular value thresholding:

$$\mathcal{G}^* = \mathcal{U}_f * \mathcal{S}_\gamma(\mathcal{S}_f) * \mathcal{V}_f^H, \tag{10}$$

where $\mathcal{S}_\gamma(s) = \max(s - \gamma, 0)$ applies soft-thresholding to the singular values.

**4) Optimization with Respect to $P^v$:** Fixing other variables, the subproblem for $P^v$ is formulated as:

$$P^{v,*} = \underset{(P^v)^T P^v = I}{\arg\max} \operatorname{Tr}\left((P^v)^T N^v\right), \tag{11}$$

where $N^v = A^T(Z_{anc}^v)^T(X^v + Y^v - E^v)$. The optimal solution is $P^{v,*} = U_P^v (V_P^v)^T$, where $U_P^v$ and $V_P^v$ are the left and right singular matrices of $N^v$.

**5) Optimization with Respect to $A$:** Similarly, the subproblem for $A$ is:

$$A^* = \underset{A^T A = I}{\arg\max} \operatorname{Tr}\left(A^T M\right), \tag{12}$$

where $M = \sum_{v=1}^m (Z_{anc}^v)^T(X^v + Y^v - E^v)(P^v)^T$. The optimal solution is $A^* = U_A V_A^T$, where $U_A$ and $V_A$ are the left and right singular matrices of $M$.

**Lagrangian Multiplier Update:** Following the standard ADMM scheme, the multipliers are updated as:

$$\begin{aligned} Y^v &\leftarrow Y^v + (X^v - Z_{anc}^v AP^v - E^v), \\ W &\leftarrow W + (\mathcal{Z}_{anc} - \mathcal{G}). \end{aligned} \tag{13}$$

### *B. Deep Unfolding Network Architecture*

While the ADMM optimization in Subsection 3.1 provides an interpretable solution to anchor-based tensorial multi-view clustering, it relies on manually tuned hyperparameters and lacks adaptability to diverse data distributions. To integrate the strengths of optimization-based methods and deep representation learning, we propose to unfold the iterative ADMM procedure into a deep network architecture. This results in a layer-wise structure composed of four interpretable modules: **RepresentModule**, **NoiseModule**, **TensorModule**, and **AnchorModule**. The overall architecture is illustrated in Fig. 1, where the network unfolds for $L$ stages with each stage mimicking one ADMM iteration. We detail each module as follows:

**1) Representation Learning Module (RepresentModule):** This module is derived from the update rule of $Z_{anc}^v$ in (6), and is responsible for learning sparse anchor-based representations.

It transforms multi-view inputs into a unified latent space via a learnable soft-thresholding operation as:

$$Z_{anc}^{v,(l+1)} = \mathcal{S}_{\tilde{\alpha}}\left((X^v - E^{v,(l)} + Y^{v,(l)})W_e^v + G^{v,(l)} - W^{v,(l)}\right), \tag{14}$$

where $l$ denotes the layer index, $W_e^v \in \mathbb{R}^{d_v \times t}$ is a learnable encoder that replaces the fixed term $(P^v)^T A^T$, and $\tilde{\alpha} = \text{SoftPlus}(\alpha)$ is a learnable threshold ensuring non-negativity. The soft-thresholding operator is implemented using the SELU activation function as $\mathcal{S}_\theta(x) = \text{SELU}(x-\theta) - \text{SELU}(-x-\theta)$. RepresentModule captures cross-view latent alignment and promotes discriminative sparse representations.

**2) Noise Suppression Module (NoiseModule):** This module implements the update of $E^v$ in (8), aiming to isolate sample-level corruptions from meaningful signals:

$$\begin{aligned} E^{(l+1)} &= \mathcal{D}_{\tilde{\lambda}}\left([\hat{E}^{1,(l)}; \ldots; \hat{E}^{m,(l)}]^T\right), \\ \hat{E}^{v,(l)} &= X^v - Z_{anc}^{v,(l+1)} A^{(l)} P_v^{(l)} + Y^{v,(l)}, \end{aligned} \tag{15}$$

where $\tilde{\lambda} = \text{SoftPlus}(\lambda)$ is the learnable shrinkage threshold, and $\mathcal{D}_\lambda(\cdot)$ is the column-wise shrinkage operator defined in (8). The noise matrices are concatenated across all views before applying column-wise shrinkage, consistent with the original optimization in (7). NoiseModule adaptively suppresses corrupted samples and improves the robustness of ATMSC-Net.

**3) Tensor Low-Rank Module (TensorModule):** This module enforces the low-rank tensor constraint following (10):

$$\mathcal{G}^{(l+1)} = \text{t-SVT}_{\tilde{\gamma}}\left(\mathcal{Z}_{anc}^{(l+1)} + W^{(l)}\right), \tag{16}$$

where $\tilde{\gamma} = \text{SoftPlus}(\gamma)$ is the learnable singular value threshold, and t-SVT denotes the tensor singular value thresholding operator. TensorModule captures high-order correlations across views by enforcing a shared low-rank structure in the tensor space.

**4) Anchor Alignment Module (AnchorModule):** This module updates the anchor indicators $P^v$ and shared dictionary $A$ following (11)-(12). For notational simplicity, let $\tilde{X}^{v,(l)} = X^v + Y^{v,(l)} - E^{v,(l+1)}$ denote the noise-corrected reconstruction target. The update rules are:

$$P_v^{(l+1)} = \text{SVD-Proj}\left((A^{(l)})^T (Z_{anc}^{v,(l+1)})^T \tilde{X}^{v,(l)}\right), \tag{17}$$

$$A^{(l+1)} = \text{SVD-Proj}\left(\sum_{v=1}^{m} (Z_{anc}^{v,(l+1)})^T \tilde{X}^{v,(l)} (P_v^{(l+1)})^T\right), \tag{18}$$

where $\text{SVD-Proj}(M) = UV^T$ with $M = U\Sigma V^T$ projects a matrix onto the orthogonal manifold. AnchorModule aligns each view's anchor structure with the clustering representation while maintaining orthogonality constraints.

The Lagrangian multipliers are updated following (13):

$$\begin{aligned} Y^{v,(l+1)} &= Y^{v,(l)} + \left(X^v - Z_{anc}^{v,(l+1)} A^{(l+1)} P_v^{(l+1)} - E^{v,(l+1)}\right) \\ W^{(l+1)} &= W^{(l)} + \left(\mathcal{Z}_{anc}^{(l+1)} - \mathcal{G}^{(l+1)}\right). \end{aligned} \tag{19}$$

**Algorithm 1** ATMSC-Net

**Require:** Multi-view data $\{X^v\}_{v=1}^m$, training epochs $T$, the number of unfolding layers $L$, and learning rate $\eta$.
**Ensure:** $F$ as the representation for $k$-means.

```
Initialize network parameters Θ = {W_e^v, α, λ, γ};
Initialize anchor matrices A, {P^v} by orthogonalization;
for i = 1 → T do
    for l = 1 → L do
        Obtain Z_anc^(l) by RepresentModule (14);
        Calculate noise E^(l) by NoiseModule (15);
        Update tensor G^(l) by TensorModule (16);
        Update P^(l), A^(l) by AnchorModule (17)-(18);
        Update multipliers Y^(l), W^(l) by (19);
    end for
    Compute the training loss (20);
    Update Θ through backward propagation;
end for
return F as the representation for k-means.
```

### C. Multi-view Unsupervised Training Loss

To enable unsupervised training of ATMSC-Net, we adopt a reconstruction-based loss that encourages the learned representations to faithfully reconstruct each view's input:

$$\mathcal{L} = \sum_{v=1}^{m} \|X^v - Z_{anc}^v A P^v\|_F^2. \tag{20}$$

This loss ensures that the anchor-based latent space preserves sufficient information for reconstructing all views while aligning view-specific anchor indicators with the clustering structure. After training, the final clustering representation is obtained by concatenating the anchor representations across views, followed by SVD-based dimensionality reduction. The cluster assignments are then obtained via $k$-means clustering. The complete procedure is summarized in Algorithm 1.

## IV. Experiments

In this section, we conduct extensive experiments to demonstrate the effectiveness and superiority of the proposed ATMSC-Net. All experiments are implemented on a computer with an NVIDIA GeForce RTX 4070 Super GPU and an Intel Core i7-14650HX CPU.

TABLE IV
STATISTICS OF THE BENCHMARK DATASETS.

| Dataset | Samples | Views | Classes | Dimensions |
|---|---|---|---|---|
| NGs | 500 | 3 | 5 | 2000/2000/2000 |
| BBCSport | 544 | 2 | 5 | 3183/3203 |
| HW | 2000 | 6 | 10 | 216/76/64/6/240/47 |
| ALOI-100 | 11025 | 4 | 100 | 77/13/64/125 |
| NUS | 30000 | 5 | 31 | 65/226/145/74/129 |
| CIFAR10 | 50000 | 3 | 10 | 512/2048/1024 |

### A. Experimental Settings

**Datasets.** To verify the effectiveness of the proposed method, seven benchmark multi-view datasets are used for evaluation. The statistics of these datasets are summarized in Table IV.

**Baselines.** We compare ATMSC-Net with several state-of-the-art multi-view clustering methods. The shallow methods include SC-best [43], EOMSC-CA [29], t-SVD-MSC [12],

TABLE III
CLUSTERING PERFORMANCE COMPARISON. THE BEST RESULTS ARE IN **BOLD** AND THE SECOND-BEST ARE UNDERLINED.

| Dataset | Metric | $SC_{best}$ | EOMSC-CA | t-SVD-MSC | $TLS_p$NM-MSC | TBGL | ARLRR-TU | DMCAG | SparseMvC | MGBCC | LargeMvC-Net | ESTMC-$C^2$ | ATMSC-Net |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| NGs | ACC | 26.06 | 63.00 | 90.00 | **100.00** | 34.20 | **100.00** | 94.50 | 55.40 | 65.80 | 96.40 | **100.00** | **100.00** |
| | NMI | 1.84 | 46.17 | 76.64 | **100.00** | 16.17 | **100.00** | 92.35 | 30.18 | 46.45 | 89.26 | **100.00** | **100.00** |
| | ARI | 0.55 | 41.30 | 77.57 | **100.00** | 11.75 | **100.00** | 93.18 | 24.61 | 40.06 | 91.10 | **100.00** | **100.00** |
| BBCSport | ACC | 50.31 | 46.32 | 34.74 | 99.82 | 54.78 | 99.82 | 95.20 | 55.51 | 95.77 | 84.69 | 99.82 | **100.00** |
| | NMI | 21.01 | 21.57 | 1.84 | 99.29 | 27.75 | 99.38 | 93.45 | 37.29 | 87.21 | 72.19 | 99.38 | **100.00** |
| | ARI | 15.89 | 15.04 | 0.62 | 99.66 | 18.75 | 99.77 | 92.80 | 28.48 | 88.53 | 67.79 | 99.77 | **100.00** |
| HW | ACC | 73.85 | 76.00 | **100.00** | 99.95 | **100.00** | 99.90 | 97.90 | 91.70 | 85.85 | 75.01 | **100.00** | **100.00** |
| | NMI | 69.92 | 77.89 | **100.00** | 99.86 | **100.00** | 99.72 | 95.25 | 85.42 | 79.08 | 63.89 | **100.00** | **100.00** |
| | ARI | 61.20 | 69.94 | **100.00** | 99.89 | **100.00** | 99.78 | 95.40 | 82.97 | 74.00 | 56.31 | **100.00** | **100.00** |
| Aloi-100 | ACC | 64.61 | 22.48 | 71.99 | 83.59 | 66.10 | 84.44 | 63.75 | 87.53 | 88.39 | 60.03 | 89.45 | **90.83** |
| | NMI | 80.03 | 56.19 | 83.92 | 90.89 | 69.25 | 92.25 | 65.40 | 94.02 | **94.78** | 73.22 | 94.31 | 93.40 |
| | ARI | 54.52 | 6.76 | 61.71 | 77.50 | 10.74 | 79.76 | 62.15 | 83.39 | 81.73 | 38.07 | **84.76** | 80.22 |
| NUS | ACC | 11.50 | 19.51 | OM | OM | OM | OM | 13.15 | 18.48 | 15.18 | **27.79** | 21.07 | 23.84 |
| | NMI | 10.73 | 12.68 | OM | OM | OM | OM | 14.43 | 19.96 | 16.16 | 14.72 | 20.17 | **21.21** |
| | ARI | 4.10 | 6.92 | OM | OM | OM | OM | 2.74 | 8.92 | 5.94 | **10.83** | 8.38 | 9.73 |
| CIFAR10 | ACC | 90.70 | 98.85 | OM | OM | OM | OM | OM | 99.23 | 90.19 | 99.13 | **100.00** | **100.00** |
| | NMI | 80.53 | 97.03 | OM | OM | OM | OM | OM | 97.85 | 93.40 | 97.64 | **100.00** | **100.00** |
| | ARI | 80.75 | 97.50 | OM | OM | OM | OM | OM | 98.31 | 83.50 | 98.10 | **100.00** | **100.00** |

$TLS_p$NM-MSC [26], TBGL [30], ARLRR-TU [31], and ESTMC-$C^2$ [14]. The deep methods include DMCAG [20], SparseMvC [32], MGBCC [33], and LargeMvC-Net [34].

**Metrics.** Three widely used clustering metrics are adopted for evaluation: Accuracy (ACC), Normalized Mutual Information (NMI), and Adjusted Rand Index (ARI). Higher values indicate better clustering performance.

**Implementation Details.** For ATMSC-Net, the number of unfolding layers $L$ is searched in $\{1, 2, \ldots, 10\}$, with detailed settings discussed in Section IV-D. The anchor number $t$ and projection dimension $l$ are searched within $\{m, 2m, 3m\}$, where $m$ is the number of views, and we find that $t = l = m$ achieves satisfactory performance. The network is trained for 100 epochs with a learning rate of 0.001. For all baselines, parameters are tuned according to their original papers.

### *B. Clustering Results*

Table III presents the clustering performance comparison on six benchmark datasets. Overall, the proposed ATMSC-Net achieves state-of-the-art performance on four datasets (NGs, BBCSport, HW, and CIFAR10) and competitive results on the remaining two datasets. We discuss the detailed comparison from three perspectives:

**Comparison with Shallow Methods.** ATMSC-Net consistently outperforms anchor-based and matrix-oriented shallow methods (SC-best, EOMSC-CA, TBGL). Tensor-based shallow methods (t-SVD-MSC, $TLS_p$NM-MSC, ARLRR-TU) can model high-order correlations but fail to execute on large-scale datasets due to memory limitations (marked as OM in Table III). Compared with ESTMC-$C^2$, which shares the same anchor-tensor optimization foundation but employs a non-convex tensor rank surrogate, ATMSC-Net adopts the convex tensor nuclear norm to accommodate the deep unfolding framework and still achieves comparable or improved performance on all datasets, with ACC gains of 2.77% on NUS and 1.38% on ALOI-100.

**Comparison with Deep Methods.** We compare ATMSC-Net with deep multi-view clustering methods including DMCAG, SparseMvC, and MGBCC. SparseMvC and MGBCC exhibit inconsistent performance across datasets, which may stem from their heuristic network designs that lack explicit optimization guidance for adapting to diverse data distributions. DMCAG is the most relevant baseline as it also combines anchor graphs with deep learning for multi-view subspace clustering. ATMSC-Net outperforms DMCAG on all datasets where DMCAG can execute, with ACC improvements ranging from 2.10% on HW to 27.08% on ALOI-100, which validates the benefit of the proposed optimization-driven framework that integrates tensor low-rank constraints into anchor-based deep clustering.

**Comparison with Deep Unfolding Method.** LargeMvC-Net also adopts the deep unfolding paradigm but relies solely on anchor-based optimization without tensor learning. By further integrating tensor low-rank constraints into the unfolding framework, ATMSC-Net enables high-order correlation modeling across views and outperforms LargeMvC-Net on five out of six datasets, improving ACC from 60.03% to 90.83% on ALOI-100, which demonstrates the benefit of joint anchor-tensor modeling.

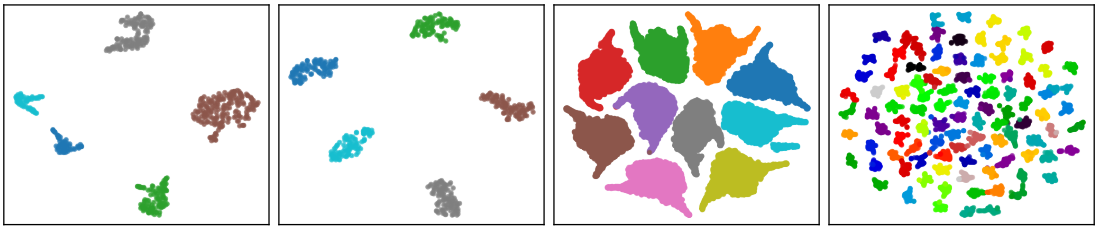
### *C. Ablation Study*

To validate the effectiveness of each module in our proposed framework, we conduct ablation experiments by progressively removing individual components. As shown in Table V, we compare the full model with four variants: (1) w/o Sparsity, which removes the SELU-based learnable soft-thresholding operator $\mathcal{S}_{\tilde{\alpha}}$ in RepresentModule; (2) w/o Noise Module, which eliminates the structured noise separation; (3) w/o Tensor Module, which discards the low-rank tensor constraint; and (4) w/o Anchor Module, which removes the orthogonal anchor alignment.

The results reveal several key observations. First, the Tensor Module plays a vital role in capturing high-order correlations across views, as its removal causes significant performance degradation on multiple datasets. Second, the Sparsity constraint and Noise Module contribute to consistent performance improvement, especially on the challenging dataset like NUS where the data contains more complex noise patterns. Third, removing the Anchor Module disrupts gradient flow to the learnable encoder $W_e^v$, leading to degraded anchor repre-

TABLE V
ABLATION STUDY ON THREE DATASETS.

| Dataset | Metric | w/o Sparsity | w/o Noise Module | w/o Tensor Module | w/o Anchor Module | Ours |
|---|---|---|---|---|---|---|
| NGs | ACC | 99.80 | 99.80 | 97.20 | 74.40 | **100.00** |
| | NMI | 99.30 | 99.30 | 91.19 | 55.28 | **100.00** |
| | ARI | 99.50 | 99.50 | 93.09 | 50.57 | **100.00** |
| BBCSport | ACC | 99.26 | 99.63 | 88.24 | 83.46 | **100.00** |
| | NMI | 97.53 | 98.60 | 75.16 | 64.19 | **100.00** |
| | ARI | 98.61 | 99.07 | 76.67 | 61.82 | **100.00** |
| NUS | ACC | 20.91 | 12.63 | 12.06 | 21.38 | **23.84** |
| | NMI | 20.40 | 10.56 | 9.82 | 19.21 | **21.21** |
| | ARI | 8.46 | 3.40 | 2.94 | 8.26 | **9.73** |



(a) BBCSport (b) NGs (c) CIFAR10 (d) ALOI-100

Fig. 2. Visualization of the clustering results on four datasets.

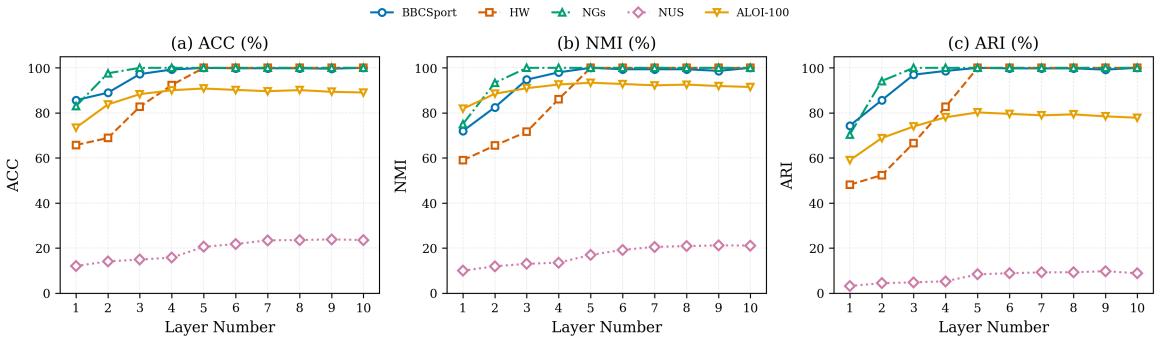



Fig. 3. Clustering performance with different numbers of unfolding layers on five datasets.

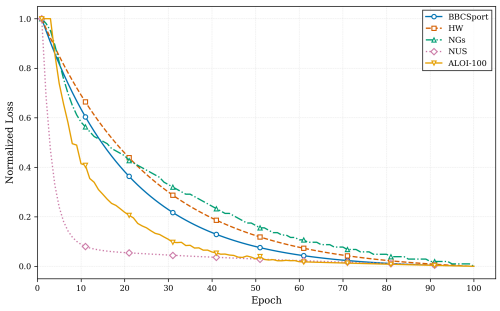



Fig. 4. Convergence analysis on five datasets.

sentation learning. These results collectively demonstrate the necessity of the complete unfolding framework.

### D. Parameter Sensitivity Analysis

As illustrated in Fig. 3, we vary the number of unfolding layers $L$ from 1 to 10. Insufficient layers limit the iterative refinement among coupled subproblems. The performance stabilizes once $L \geq 4$ across most datasets. For challenging datasets (NUS and ALOI-100), we adopt $L = 8$ to provide sufficient optimization depth. For the remaining datasets, we use $L = 5$.

### E. Convergence Analysis

Fig. 4 illustrates the convergence behavior of our method on five datasets. The normalized loss curves show that our model converges rapidly and consistently on all datasets, demonstrating the efficiency and stability of the deep unfolding framework for optimization.

### F. Visualization of Learned Representations

To intuitively demonstrate the quality of learned representations, we use t-SNE [44] to project the features into 2D space on four datasets. As shown in Fig. 2, different colors denote different cluster labels. The clustering structures are clearly visible across all datasets, and even on ALOI-100 with 100 categories, our method successfully separates different clusters into distinguishable regions, which further demonstrates the effectiveness of ATMSC-Net.

## V. CONCLUSION

In this paper, we propose ATMSC-Net, a deep unfolding network for anchor-based tensorial multi-view subspace

clustering. By mapping each ADMM subproblem to a dedicated network module with explicit optimization semantics, the proposed method bridges optimization-based methods and deep learning, transforming hand-crafted hyperparameters into learnable parameters through end-to-end training. The resulting architecture inherits the efficiency and high-order modeling capability of the anchor-tensor paradigm, while preserving the interpretability of the underlying optimization. Extensive experiments demonstrate competitive performance on multiple benchmarks.

## ACKNOWLEDGMENT

This work was supported in part by the National Natural Science Foundation of China (62332008, 62576152, 62336004), the Basic Research Program of Jiangsu (BK20250104), the Fundamental Research Funds for the Central Universities (JUSRP202504007), and Leverhulme Trust Emeritus Fellowship EM-2025-06-09.

## REFERENCES

[1] H. Su, S. Maji, E. Kalogerakis, and E. Learned-Miller, "Multi-view convolutional neural networks for 3D shape recognition," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, 2015, pp. 945–953.

[2] L. Tang, X. Wang, and H. Liu, "Community detection via heterogeneous interaction analysis," *Data Min. Knowl. Discov.*, vol. 25, no. 1, pp. 1–33, 2012.

[3] G. Litjens, T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, J. A. W. M. van der Laak, B. van Ginneken, and C. I. Sánchez, "A survey on deep learning in medical image analysis," *Med. Image Anal.*, vol. 42, pp. 60–88, 2017.

[4] C. Xu, D. Tao, and C. Xu, "A survey on multi-view learning," *arXiv preprint arXiv:1304.5634*, 2013.

[5] J. Zhao, X. Xie, X. Xu, and S. Sun, "Multi-view learning overview: Recent progress and new challenges," *Inf. Fusion*, vol. 38, pp. 43–54, 2017.

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

[7] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma, "Robust recovery of subspace structures by low-rank representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 171–184, 2013.

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

[9] S. Wang, X. Liu, X. Zhu, P. Zhang, Y. Zhang, F. Gao, and E. Zhu, "Fast parameter-free multi-view subspace clustering with consensus anchor guidance," *IEEE Trans. Image Process.*, vol. 31, pp. 556–568, 2022.

[10] Z. Kang, W. Zhou, Z. Zhao, J. Shao, M. Han, and Z. Xu, "Large-scale multi-view subspace clustering in linear time," in *Proc. AAAI Conf. Artif. Intell.*, vol. 34, no. 4, 2020, pp. 4412–4419.

[11] C. Zhang, H. Fu, S. Liu, G. Liu, and X. Cao, "Low-rank tensor constrained multiview subspace clustering," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, 2015, pp. 1582–1590.

[12] Y. Xie, D. Tao, W. Zhang, Y. Liu, L. Zhang, and Y. Qu, "On unifying multi-view self-representations for clustering by tensor multi-rank minimization," *Int. J. Comput. Vis.*, vol. 126, no. 11, pp. 1157–1179, 2018.

[13] C. Zhang, H. Fu, Q. Hu, X. Cao, Y. Xie, D. Tao, and D. Xu, "Generalized latent multi-view subspace clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 1, pp. 86–99, 2020.

[14] J. Ji and S. Feng, "Anchors crash tensor: Efficient and scalable tensorial multi-view subspace clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 47, no. 4, pp. 2660–2675, 2025.

[15] Q. Wang, J. Cheng, Q. Gao, G. Zhao, and L. Jiao, "Deep multi-view subspace clustering with unified and discriminative learning," *IEEE Trans. Multimedia*, vol. 23, pp. 3483–3493, 2020.

[16] C. Zhang, Y. Liu, and H. Fu, "AE2-Nets: Autoencoder in autoencoder networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2019, pp. 2577–2585.

[17] J. Xu, C. Li, Y. Ren, L. Peng, Y. Mo, X. Shi, and H. Zhu, "Deep incomplete multi-view clustering via mining cluster complementarity," in *Proc. AAAI Conf. Artif. Intell.*, vol. 36, no. 8, 2022, pp. 8761–8769.

[18] J. Xu, H. Tang, Y. Ren, L. Peng, X. Zhu, and L. He, "Multi-level feature learning for contrastive multi-view clustering," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2022, pp. 16 051–16 060.

[19] W. He and Z. Zhang, "Adaptive topological graph learning for generalized multi-view clustering," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*. IEEE, 2023, pp. 1–8.

[20] C. Cui, Y. Ren, J. Pu, X. Pu, and L. He, "Deep multi-view subspace clustering with anchor graph," in *Proc. Int. Joint Conf. Artif. Intell. (IJCAI)*, 2023, pp. 3577–3585.

[21] B. Wang, C. Zeng, M. Chen, and X. Li, "Towards learnable anchor for deep multi-view clustering," in *Proc. AAAI Conf. Artif. Intell.*, vol. 39, no. 20, 2025, pp. 21 044–21 052.

[22] Z. Liu and P. Song, "Deep low-rank tensor embedding for multi-view subspace clustering," *Expert Syst. Appl.*, vol. 234, p. 120913, 2024.

[23] Q. Wang, Z. Zhang, W. Feng, Z. Tao, and Q. Gao, "Contrastive multi-view subspace clustering via tensor transformers autoencoder," in *Proc. AAAI Conf. Artif. Intell.*, vol. 39, 2025.

[24] A. Kumar, P. Rai, and H. Daume, "Co-regularized multi-view spectral clustering," in *Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 24, 2011, pp. 1413–1421.

[25] W. Liu, L. Liu, L. Feng, and H. Deng, "Tensorized multi-view clustering via hyper-graph regularization," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*. IEEE, 2022, pp. 1–8.

[26] J. Guo, Y. Sun, J. Gao, B. Yin, Y. Hu, and B. Wen, "Logarithmic schatten-$p$ norm minimization for tensorial multi-view subspace clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 3, pp. 3396–3410, 2023.

[27] Z. Gu and S. Feng, "From dictionary to tensor: A scalable multi-view subspace clustering framework with triple information enhancement," in *Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 37, 2024, pp. 103 545–103 573.

[28] P. Zhu, X. Yao, Y. Wang, B. Hui, D. Du, and Q. Hu, "Multiview deep subspace clustering networks," *IEEE Trans. Cybern.*, vol. 54, no. 7, pp. 4280–4293, 2024.

[29] S. Liu, S. Wang, P. Zhang, K. Xu, X. Liu, C. Zhang, and F. Gao, "Efficient one-pass multi-view subspace clustering with consensus anchors," in *Proc. AAAI Conf. Artif. Intell.*, vol. 36, no. 7, 2022, pp. 7576–7584.

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

[31] Y. Tang, Y. Xie, and W. Zhang, "Affine subspace robust low-rank self-representation: From matrix to tensor," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 8, pp. 9357–9373, 2023.

[32] R. Liu, X. Zou, C. Tang, X. Zheng, X. Hu, K. Sun, and X. Liu, "Sparsemvc: Probing cross-view sparsity variations for multi-view clustering," in *Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2025.

[33] P. Su, S. Huang, W. Ma, D. Xiong, and J. Lv, "Multi-view granular-ball contrastive clustering," in *Proc. AAAI Conf. Artif. Intell.*, vol. 39, no. 19, 2025, pp. 20 637–20 645.

[34] S. Du, Z. Fang, Z. Wu, Y. Pi, S. Wang *et al.*, "LargeMvC-Net: Anchor-based deep unfolding network for large-scale multi-view clustering," in *Proc. ACM Int. Conf. Multimedia (MM)*, Dublin, Ireland, 2025.

[35] A. Balatsoukas-Stimming and C. Studer, "Deep unfolding for communications systems: A survey and some new directions," in *Proc. IEEE Int. Workshop Signal Process. Syst. (SiPS)*, 2019, pp. 266–271.

[36] K. Gregor and Y. LeCun, "Learning fast approximations of sparse coding," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2010, pp. 399–406.

[37] J. Zhang and B. Ghanem, "ISTA-Net: Interpretable optimization-inspired deep network for image compressive sensing," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2018, pp. 1828–1837.

[38] Y. Yang, J. Sun, H. Li, and Z. Xu, "Deep ADMM-Net for compressive sensing MRI," in *Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 29, 2016, pp. 10–18.

[39] Y. Lu and J. Zhang, "Federated deep unfolding for compressive sensing," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*. IEEE, 2024, pp. 1–7.

[40] S. Cui, F. Xu, X. Tang, and Q. Zheng, "Model driven deep unfolding network for extreme low-light image enhancement and denoising," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*. IEEE, 2023, pp. 1–8.

[41] X. Li, N. Nadisic, S. Huang, and A. Pizurica, "Unfolding ADMM for enhanced subspace clustering of hyperspectral images," in *Proc. Eur. Signal Process. Conf. (EUSIPCO)*, 2024.

[42] X. Xie, J. Wu, G. Liu *et al.*, "SSCNet: Learning-based subspace clustering," *Visual Intelligence*, vol. 2, no. 11, 2024.

[43] U. von Luxburg, "A tutorial on spectral clustering," *Stat. Comput.*, vol. 17, no. 4, pp. 395–416, 2007.

[44] L. Van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, no. 86, pp. 2579–2605, 2008.