PRESERVING CONSISTENCY AND PROGRESSIVELY INTEGRATING DIVERSITY FOR CONTINUAL MULTI-VIEW CLUSTERING

Anonymous authors
Paper under double-blind review

ABSTRACT

Continual multi-view clustering (CMVC) updates clustering representations as new views arrive sequentially, requiring both the preservation of shared structures and the integration of complementary information. Existing methods typically preserve historical knowledge by constraining current representations to remain compatible with previous ones. Such compatibility constraints provide limited semantic guidance and often underexploit view-specific diversity. To address these limitations, we propose **Preserving Consistency and Progressively Integrating Diversity for Continual Multi-View Clustering (PC-PID)**. PC-PID decomposes adjacent-view representations into a shared consistency component and view-specific diversity components. Contrastive knowledge transfer strengthens semantic dependence between current and historical consistency representations while suppressing redundant dependence between adjacent diversity representations, thereby preserving shared structures and separating view-specific information. Progressive tensor fusion constructs a local tensor from adjacent diversity representations and captures their high-order correlations as new views arrive. These two mechanisms are incorporated into a unified objective and optimized through an alternating algorithm. Experiments on eight benchmark datasets against 14 representative static and continual multi-view clustering methods demonstrate the effectiveness and efficiency of PC-PID.

1 INTRODUCTION

Multi-view clustering (MVC) partitions unlabeled data by exploiting complementary information from multiple views. Most existing MVC methods assume that all views are available simultaneously (Lu et al., 2024; Wang & Feng, 2024; Wu et al., 2024; Chen et al., 2026). However, this assumption is often impractical. For example, cameras may be deployed progressively in intelligent vehicles, medical images may be collected across multiple clinical stages, and recommender systems may continually acquire new multimodal features. In these scenarios, views cannot be obtained at once; retraining on all views is costly, while storage and privacy requirements may also prevent retaining all historical raw views. These practical requirements motivate continual multi-view clustering (CMVC), which updates clustering representations as new views arrive using compact historical knowledge.

Existing CMVC methods preserve previous knowledge through anchors, prototypes, distribution alignment, structure adaptation, or anti-degradation constraints (Zhang et al., 2024; Wang et al., 2025; Yuan et al., 2025; Li et al., 2026). Although effective for maintaining learning stability, these methods commonly encourage compatibility between historical and current representations, without explicitly distinguishing transferable common structures from newly introduced view-specific information. Meanwhile, contrastive learning has been used to model semantic relations among views in static MVC (Lu et al., 2024; Luo et al., 2024; Cai et al., 2026), while tensor-based methods exploit high-order cross-view correlations through low-rank modeling (Wu et al., 2024; Ji et al., 2024). Their direct use in CMVC remains challenging because views evolve across stages and historical raw views are not fully accessible. Therefore, an effective CMVC model should preserve reliable common knowledge while progressively integrating complementary information from incoming views.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

To address these challenges, we propose **Preserving Consistency and Progressively Integrating Diversity for Continual Multi-View Clustering (PC-PID)**. PC-PID decomposes adjacent-view representations into a shared consistency component and view-specific diversity components. Contrastive knowledge transfer preserves shared structures by strengthening the dependence between current and historical consistency representations while suppressing redundant dependence between adjacent diversity representations. Progressive tensor fusion integrates view-specific diversity by organizing adjacent-stage diversity representations into a local tensor and exploiting their high-order correlations during continual updating. The overall framework of PC-PID is shown in Figure 1.

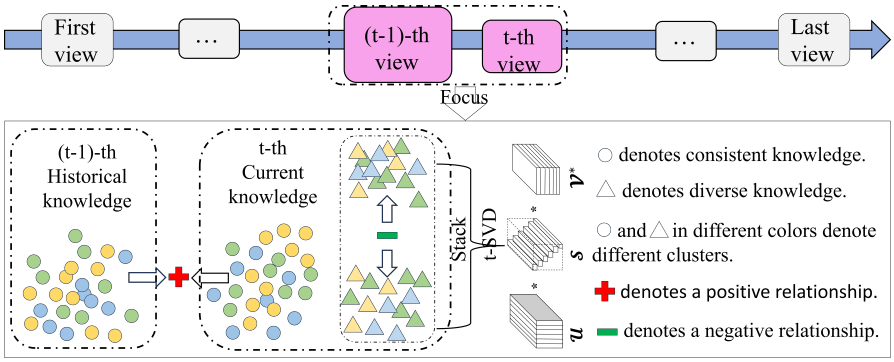

Figure 1: Overview of PC-PID. At stage t, PC-PID jointly processes the current view and its preceding view and decomposes their latent representations into a shared consistency representation (circles) and view-specific diversity representations (triangles). Contrastive knowledge transfer establishes a positive relation between the current and historical consistency representations (red "+") and a negative relation between the adjacent diversity representations (green "−"). Progressive tensor fusion then stacks the adjacent diversity representations into a local tensor and applies t-SVD to capture their high-order correlations. Different colors denote different clusters.

The main contributions of this paper are summarized as follows:

- We propose a contrastive knowledge transfer mechanism for CMVC that strengthens the dependence between historical and current consistency representations while suppressing redundant dependence between view-specific diversity representations.
- We introduce progressive tensor fusion to incrementally integrate complementary information from incoming views by modeling high-order correlations between adjacent-stage diversity representations.
- We develop a unified optimization framework for continual multi-view clustering and validate PC-PID on eight benchmark datasets against 14 representative static and continual multi-view clustering methods.

2 Related Work

2.1 Static Multi-View Clustering

Static multi-view clustering (MVC) assumes that all views are available simultaneously and aims to exploit their shared semantics and complementary information for clustering. Recent methods improve representation learning by constructing contrastive relations across views (Lu et al., 2024; Luo et al., 2024; Cai et al., 2026), modeling inter-view interactions (Wang & Feng, 2024), separating shared and private representations (Chen et al., 2026), or imposing low-rank tensor structures to capture high-order correlations (Li et al., 2023; Wu et al., 2024; Ji et al., 2024). These studies demonstrate the importance of modeling both consistency and complementarity among views. However, their objectives are typically defined on a fixed and complete view set, and thus cannot directly handle views that arrive sequentially.

2.2 CONTINUAL MULTI-VIEW CLUSTERING

Continual multi-view clustering updates clustering representations as new views arrive while retaining useful knowledge learned at previous stages. Existing methods mainly preserve historical information through anchors, prototypes, distribution alignment, or anti-degradation constraints (Zhang et al., 2024; Wang et al., 2025; Li et al., 2026; Yuan et al., 2025). These methods have shown that maintaining historical knowledge is essential for reducing performance degradation in dynamic multi-view scenarios. Nevertheless, most existing approaches focus on preserving the stability of previous representations, while the distinction between transferable common information and newly arriving view-specific information remains insufficiently explored.

3 MOTIVATION AND PROPOSED METHOD

3.1 MOTIVATION

In CMVC, new views arrive sequentially, requiring the model to preserve historical common structures while incorporating complementary information from incoming views. Existing methods usually transfer knowledge by enforcing similarity or alignment between consecutive representations. Although effective for mitigating forgetting, these strategies provide limited semantic guidance because they do not distinguish transferable consistency information from newly introduced view-specific diversity. This raises a natural question: can historical knowledge be transferred in a more semantic and discriminative manner than through simple distance preservation?

Meanwhile, incoming view-specific information may exhibit high-order correlations across continual stages, which are difficult to capture through independent processing or simple matrix-level aggregation. This raises a second question: how can such information be progressively fused without retaining all historical raw views? These questions motivate contrastive knowledge transfer and progressive tensor fusion under a consistency–diversity decomposition framework.

3.2 PROBLEM FORMULATION

Assume that a multi-view dataset contains n samples and its views arrive incrementally. Let $\mathbf{X}^{(t)} \in \mathbb{R}^{n\times d_t}$ denote the view arriving at stage t, where d_t is its feature dimension. We employ an m-dimensional latent space, with consistency and diversity representations in $\mathbb{R}^{n\times m}$ and projection matrices in $\mathbb{R}^{m\times d_t}$. The goal is to update the clustering representation using the incoming view, its immediately preceding view, and compact historical knowledge, without retaining or revisiting the complete set of historical raw views.

First stage. Since no historical knowledge is available, we initialize the consistency representation from the first view:

$$\begin{aligned} \min_{\mathbf{C}^{(1)},\mathbf{P}^{(1)}} \quad & \mathcal{L}_{rec}^{(1)} = \left\| \mathbf{X}^{(1)} - \mathbf{C}^{(1)}\mathbf{P}^{(1)} \right\|_F^2 \\ \text{s.t.} \quad & \mathbf{P}^{(1)}\mathbf{P}^{(1)\top} = \mathbf{I}. \end{aligned} \tag{1}$$

Subsequent stages. For $t \geq 2$, let $\mathcal{I}_t = \{t-1, t\}$ and $\Theta^{(t)} = \{\mathbf{C}^{(t)}, \mathbf{D}^{(t-1)}, \mathbf{D}^{(t)}, \mathbf{P}^{(t-1)}, \mathbf{P}^{(t)}\}$. We update the model by jointly reconstructing the two adjacent views:

$$\begin{aligned} \min_{\Theta^{(t)}} \quad & \mathcal{L}_{rec}^{(t)} = \sum_{i\in\mathcal{I}_t} \left\| \mathbf{X}^{(i)} - \left(\mathbf{C}^{(t)} + \mathbf{D}^{(i)}\right)\mathbf{P}^{(i)} \right\|_F^2 \\ \text{s.t.} \quad & \mathbf{P}^{(i)}\mathbf{P}^{(i)\top} = \mathbf{I}, \quad i \in \mathcal{I}_t. \end{aligned} \tag{2}$$

Here, $\mathbf{C}^{(t)}$ captures the common cluster structure shared by the adjacent views, while $\mathbf{D}^{(i)}$ preserves the complementary information specific to view i. The orthogonality constraints prevent degenerate projections and stabilize the latent representation scale.

3.3 CONTRASTIVE KNOWLEDGE TRANSFER

A common strategy for historical knowledge transfer is to minimize $\|\mathbf{C}^{(t)} - \mathbf{C}^{(t-1)}\|_F^2$ (Kirkpatrick et al., 2017; Li & Hoiem, 2018; De Lange et al., 2022). However, point-wise distance preservation

cannot explicitly distinguish transferable common structures from newly introduced view-specific information. Previous studies have shown that separating shared and private representations is important for preserving invariant information while retaining view-specific factors (Bousmalis et al., 2016; Ke et al., 2024). Since consistency representations encode common clustering structures across stages, we regard $(\mathbf{C}^{(t)}, \mathbf{C}^{(t-1)})$ as a positive knowledge pair, following the principle that semantically shared information should remain dependent across views or stages (Tian et al., 2020).

In contrast, strong dependence between diversity representations may indicate redundant information rather than complementary factors. Therefore, $(\mathbf{D}^{(i)}, \mathbf{D}^{(v)})$, $v \neq i$, is treated as a negative knowledge pair whose linear dependence should be suppressed. Similar dependence minimization has been used to reduce redundancy and promote complementarity among multi-view representations (Cao et al., 2015; Ke et al., 2024; Lu et al., 2024; Luo et al., 2024; Wang & Feng, 2024; Guo et al., 2024; Hu et al., 2025).

Definition 1. Given $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{n\times m}$, where n is the number of samples and $m > 1$ is the latent dimension, their linear-kernel Hilbert–Schmidt independence criterion (HSIC) (Gretton et al., 2005; Song et al., 2012) is defined as

$$\mathrm{HSIC}(\mathbf{A}, \mathbf{B}) = \frac{1}{(m-1)^2} \operatorname{tr}\left(\mathbf{K}_A \mathbf{K}_B\right), \tag{3}$$

where

$$\mathbf{K}_A = \mathbf{H}\mathbf{A}^\top \mathbf{A}\mathbf{H}, \qquad \mathbf{K}_B = \mathbf{H}\mathbf{B}^\top \mathbf{B}\mathbf{H}, \qquad \mathbf{H} = \mathbf{I} - \frac{1}{m}\mathbf{1}\mathbf{1}^\top. \tag{4}$$

Here, $\mathbf{H} \in \mathbb{R}^{m\times m}$ centers the latent dimensions, while $\mathbf{K}_A$ and $\mathbf{K}_B$ characterize their centered second-order structures. Centered kernel alignment provides a principled measure of structural similarity between kernel matrices and learned representations (Cortes et al., 2012; Kornblith et al., 2019).

Based on Definition 1, we formulate the contrastive knowledge transfer regularizer as

$$\mathcal{L}_{ckt}^{(t)} = \frac{\displaystyle\sum_{i=t-1}^{t} \sum_{\substack{v=t-1 \\ v\neq i}}^{t} \mathrm{HSIC}\left(\mathbf{D}^{(i)}, \mathbf{D}^{(v)}\right)}{\mathrm{HSIC}\left(\mathbf{C}^{(t)}, \mathbf{C}^{(t-1)}\right) + \epsilon}, \tag{5}$$

where $\epsilon > 0$ avoids numerical instability. The denominator measures the structural dependence between the current and historical consistency representations, whereas the numerator measures redundant linear dependence between diversity representations associated with different views. The ordered-pair summation provides a symmetric formulation, and its constant factor is absorbed into the trade-off parameter λ_1.

Assumption 1. The historical consistency representation $\mathbf{C}^{(t-1)}$ provides a reliable encoding of the common clustering structure learned before stage t.

Proposition 1. For any $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{n\times m}$, the HSIC in Definition 1 satisfies

$$\mathrm{HSIC}(\mathbf{A}, \mathbf{B}) = \frac{\left\|\mathbf{A}\mathbf{H}\mathbf{B}^\top\right\|_F^2}{(m-1)^2}. \tag{6}$$

Furthermore, for nonzero $\mathbf{K}_A$ and $\mathbf{K}_B$,

$$0 \leq \frac{\operatorname{tr}\left(\mathbf{K}_A \mathbf{K}_B\right)}{\|\mathbf{K}_A\|_F \|\mathbf{K}_B\|_F} \leq 1. \tag{7}$$

The upper bound is attained if and only if $\mathbf{K}_A = \alpha \mathbf{K}_B$ for some $\alpha > 0$, while $\mathrm{HSIC}(\mathbf{A}, \mathbf{B}) = 0$ if and only if $\mathbf{A}\mathbf{H}\mathbf{B}^\top = \mathbf{0}$.

Proof. Since $\mathbf{H}^\top = \mathbf{H}$ and $\mathbf{H}^2 = \mathbf{H}$, cyclic invariance of the trace gives

$$\operatorname{tr}\left(\mathbf{K}_A \mathbf{K}_B\right) = \operatorname{tr}\left(\mathbf{H}\mathbf{A}^\top \mathbf{A}\mathbf{H}\mathbf{B}^\top \mathbf{B}\mathbf{H}\right) = \operatorname{tr}\left(\mathbf{A}\mathbf{H}\mathbf{B}^\top \mathbf{B}\mathbf{H}\mathbf{A}^\top\right) = \left\|\mathbf{A}\mathbf{H}\mathbf{B}^\top\right\|_F^2. \tag{8}$$

Substituting this identity into Eq. (3) proves Eq. (6) and establishes non-negativity. It also shows that HSIC equals zero exactly when $\mathbf{A}\mathbf{H}\mathbf{B}^\top = \mathbf{0}$.

Because $\mathbf{K}_A$ and $\mathbf{K}_B$ are positive semi-definite, their Frobenius inner product is non-negative. By the Cauchy–Schwarz inequality,

$$\operatorname{tr}(\mathbf{K}_A\mathbf{K}_B) \le \|\mathbf{K}_A\|_F\|\mathbf{K}_B\|_F. \tag{9}$$

Equality holds if and only if the two nonzero kernel matrices are positively linearly dependent, i.e., $\mathbf{K}_A = \alpha\mathbf{K}_B$ for some $\alpha > 0$. This proves Eq. (7).

Proposition 1 provides a relational interpretation of the proposed transfer mechanism. For the positive pair, increasing $\mathrm{HSIC}(\mathbf{C}^{(t)}, \mathbf{C}^{(t-1)})$ aligns their centered second-order structures. At the upper bound, the two consistency kernels preserve the same latent relational geometry up to scale, which is consistent with the centered-kernel alignment interpretation (Cortes et al., 2012; Kornblith et al., 2019). Under Assumption 1, this alignment transfers the common clustering structure encoded by the historical representation.

For a negative pair, decreasing $\mathrm{HSIC}(\mathbf{D}^{(i)}, \mathbf{D}^{(v)})$ reduces $\|\mathbf{D}^{(i)}\mathbf{H}\mathbf{D}^{(v)\top}\|_F^2$ and suppresses linearly shared variation between view-specific components, thereby promoting their complementarity (Cao et al., 2015; Ke et al., 2024). Consequently, minimizing Eq. (5) biases the model toward preserving historical consistency while separating redundant diversity information, providing a structural basis for contrastive knowledge transfer.

3.4 Progressive Tensor Fusion

To capture high-order correlations among view-specific complementary information, we construct a local tensor from the diversity representations of the previous and current views:

$$\mathcal{D}^{(t)} = \operatorname{cat}\left(\mathbf{D}^{(t-1)}, \mathbf{D}^{(t)}\right), \tag{10}$$

where $\operatorname{cat}(\cdot)$ stacks matrices along the third mode. Although only two adjacent representations are involved at each stage, repeated local fusion progressively incorporates complementary information as new views arrive.

Definition 2. Let $\widehat{\mathcal{D}}^{(t)} = \operatorname{fft}(\mathcal{D}^{(t)}, [], 3)$ denote the tensor transformed along its third mode. Its tensor nuclear norm is defined as (Kilmer & Martin, 2011; Lu et al., 2020)

$$\left\|\mathcal{D}^{(t)}\right\|_{\circledast} = \sum_{k=1}^{2} \left\|\widehat{\mathcal{D}}^{(t)}(:,:,k)\right\|_*, \tag{11}$$

where $\|\cdot\|_*$ denotes the matrix nuclear norm.

The progressive tensor fusion regularizer is formulated as

$$\mathcal{L}_{ptf}^{(t)} = \left\|\mathcal{D}^{(t)}\right\|_{\circledast}. \tag{12}$$

Since the tensor nuclear norm is a convex surrogate for tensor multi-rank (Kilmer & Martin, 2011; Lu et al., 2020), minimizing $\mathcal{L}_{ptf}^{(t)}$ promotes a compact low-rank structure in the Fourier tensor space. This captures high-order correlations between adjacent diversity representations without constructing a global tensor over all historical stages.

3.5 Overall Objective

Combining reconstruction, contrastive knowledge transfer, and progressive tensor fusion gives the following objective for stage $t \ge 2$:

$$\begin{aligned} \min_{\Theta^{(t)}} \quad & \mathcal{L}^{(t)} = \mathcal{L}_{rec}^{(t)} + \lambda_1\mathcal{L}_{ckt}^{(t)} + \lambda_2\mathcal{L}_{ptf}^{(t)} \\ & \text{s.t.} \quad \mathbf{P}^{(i)}\mathbf{P}^{(i)\top} = \mathbf{I}, \quad i \in \mathcal{I}_t, \end{aligned} \tag{13}$$

where λ_1 and λ_2 are trade-off parameters controlling contrastive knowledge transfer and progressive tensor fusion, respectively.

4 OPTIMIZATION

We optimize PC-PID in a stage-wise alternating manner. At the first stage, $\mathbf{C}^{(1)}$ and $\mathbf{P}^{(1)}$ are alternately updated through orthogonal Procrustes optimization. At each subsequent stage, we introduce an auxiliary tensor to decouple the tensor nuclear norm and optimize the resulting augmented Lagrangian. With the remaining variables fixed, the projection matrices, diversity representations, consistency representation, auxiliary tensor, and Lagrange multipliers all admit closed-form updates. Since the CKT term is non-convex, its kernel quantities are handled using a fixed-point alternating scheme.

The complete augmented Lagrangian, closed-form updates, initialization, stopping criteria, and stage-wise algorithm are provided in Appendix B.

Complexity analysis. Let $d = \max_i d_i$ and let r_t denote the number of iterations at stage t. Since only two adjacent views are processed, the computational complexity at stage t is

$$\mathcal{O}\left(r_t\left(nmd + m^2d + nm^2 + m^3\right)\right). \tag{14}$$

For V views and $r = \max_t r_t$, the total computational complexity is

$$\mathcal{O}\left(rV\left(nmd + m^2d + nm^2 + m^3\right)\right). \tag{15}$$

Unlike static methods that jointly process all views, PC-PID uses a fixed $n \times m \times 2$ local tensor. Its per-stage cost therefore does not grow with the number of previously arrived views, preserving the efficiency of continual updating.

The storage complexity is

$$\mathcal{O}\left(Vnm + \sum_{i=1}^{V} md_i\right). \tag{16}$$

Since $m \ll n, d_i$ in practice, this is substantially smaller than retaining all historical raw views, which requires $\mathcal{O}(n\sum_{i=1}^{V} d_i)$ memory. Moreover, the local tensor requires only fixed $\mathcal{O}(nm)$ working memory, which does not increase with the continual stage.

5 EXPERIMENTS

We evaluate PC-PID by addressing six questions: **Q1**: How does PC-PID compare with state-of-the-art multi-view clustering methods? **Q2**: How much do contrastive knowledge transfer and progressive tensor fusion contribute? **Q3**: Does PC-PID improve progressively as new views arrive? **Q4**: Is PC-PID computationally efficient? **Q5**: How sensitive is it to the key parameters? **Q6**: Does its optimization converge stably?

5.1 EXPERIMENTAL SETUP

We conduct experiments on eight widely used multi-view datasets, including MSRCV1, BBCSport, Reuters-1200, MNIST-4, Scene15, Caltech101-all, STL10, and AwA. These datasets contain 210–30,475 samples and vary in their numbers of views, categories, and feature types. Detailed dataset statistics and source links are provided in the supplementary material.

We compare PC-PID with 14 representative methods, including seven static methods, i.e., DAG-FC (TFS'25) (Chen et al., 2025), RCAGL (TKDE'24) (Liu et al., 2024), SMVAGC (TIP'24) (Wang et al., 2024), TMC (TKDE'26) (Zhang et al., 2026), TLRLF4MVC (TPAMI'25) (Long et al., 2025), Orth-NTF (NeurIPS'23) (Li et al., 2023), and RAV (ICLR'26) (Peng et al., 2026), and seven continual/incremental methods, i.e., AdaptCMVC (CVPR'25) (Wang et al., 2025), ASIA-MVC (TCSVT'26) (Qu et al., 2026), CDIMVC (TKDE'25) (Feng et al., 2025a), INMKC (AAAI'25) (Feng et al., 2025b), ACMVC (IJCAI'24) (Zhang et al., 2024), CMVC (MM'22) (Wan et al., 2022), and LAIMVC (MM'24) (Feng et al., 2024). For compact presentation in Table 1, TLRLF4MVC, AdaptCMVC, and ASIA-MVC are abbreviated as TLRLF4, Adapt, and ASIA, respectively.

Following the commonly adopted CMVC protocol (Zhang et al., 2024; Wang et al., 2025; Yuan et al., 2025), views in each dataset arrive in a fixed pre-specified order. PC-PID and all continual

baselines use the same order, whereas static baselines are trained once using the complete view set and serve as full-view references.

We adopt accuracy (ACC), normalized mutual information (NMI), purity, and F-score (F). Each method is independently run 10 times under the fixed view-arrival order, and Table 1 reports the mean performance. All experiments are conducted in MATLAB R2024a on a Windows 11 workstation with an Intel Core i7-14700KF CPU, 48 GB RAM, and an NVIDIA GeForce RTX 4060 Ti GPU with 16 GB memory.

5.2 Comparison with State-of-the-Art Methods (Q1)

Table 1: Overall clustering performance comparison in terms of ACC, NMI, Purity, and F-score on eight multi-view datasets. The best and second-best results are indicated by boldface and underlining, respectively. "–" denotes unavailable results.

Type	Method	MSRCV1				BBCSport				Reuters-1200				MNIST-4			
		ACC	NMI	Purity	F	ACC	NMI	Purity	F	ACC	NMI	Purity	F	ACC	NMI	Purity	F
Static	DAG-FC	0.933	0.873	0.933	0.871	0.511	0.218	0.553	0.404	0.262	0.262	0.262	0.251	0.262	0.003	0.262	0.391
	RCAGL	0.862	0.773	0.889	0.754	0.549	0.401	0.606	0.442	0.477	0.520	0.520	0.338	0.893	0.732	0.893	0.811
	SMVAGC	0.597	0.465	0.623	0.451	0.808	0.702	0.832	0.796	0.549	0.557	0.557	0.392	0.883	0.704	0.883	0.794
	TMC	0.657	0.526	0.657	0.503	0.716	0.481	0.717	0.607	0.525	0.314	0.524	0.372	0.743	0.502	0.743	0.610
	TLRLF4	–	–	–	–	0.373	0.039	0.376	0.387	0.685	0.656	0.691	0.588	–	–	–	–
	Orth-NTF	0.957	0.919	0.957	0.918	0.917	0.828	0.917	0.871	0.758	0.712	0.758	0.663	**0.943**	**0.864**	**0.943**	**0.896**
	RAV	0.957	0.911	0.957	0.916	0.522	0.317	0.558	0.396	0.430	0.242	0.443	0.318	0.898	0.740	0.898	0.820
Continual	Adapt	0.406	0.401	0.431	0.130	0.328	0.069	0.390	0.221	0.530	0.276	0.530	0.528	0.823	0.625	0.823	0.822
	ASIA	0.524	0.456	0.557	0.417	0.774	0.579	0.774	0.635	0.497	0.497	0.497	0.332	0.733	0.494	0.733	0.608
	CDIMVC	0.905	0.827	0.905	0.821	0.379	0.074	0.428	0.309	0.331	0.143	0.380	0.276	0.912	0.792	0.912	0.843
	INMKC	0.752	0.684	0.766	0.662	0.389	0.053	0.401	0.387	0.506	0.252	0.506	0.333	0.508	0.227	0.521	0.415
	ACMVC	0.871	0.753	0.872	0.765	0.664	0.436	0.677	0.541	0.496	0.252	0.508	0.351	0.888	0.721	0.888	0.803
	CMVC	0.605	0.466	0.614	0.441	0.620	0.266	0.620	0.489	0.273	0.053	0.295	0.243	0.902	0.759	0.902	0.827
	LAIMVC	0.524	0.382	0.529	0.372	0.691	0.458	0.691	0.516	0.424	0.182	0.433	0.297	0.675	0.488	0.675	0.599
	PC-PID	**0.991**	**0.978**	**0.991**	**0.981**	**0.993**	**0.976**	**0.993**	**0.984**	**0.923**	**0.833**	**0.923**	**0.859**	0.940	**0.864**	0.940	0.891
Type	**Method**	**Scene15**				**Caltech101-all**				**STL10**				**AwA**			
		ACC	NMI	Purity	F	ACC	NMI	Purity	F	ACC	NMI	Purity	F	ACC	NMI	Purity	F
Static	DAG-FC	0.621	0.524	0.666	0.497	0.315	0.509	0.487	0.274	0.176	0.043	0.178	0.139	–	–	–	–
	RCAGL	0.550	0.569	0.567	0.447	0.373	0.479	0.561	0.256	0.947	0.135	0.947	0.898	0.096	0.121	0.236	0.062
	SMVAGC	0.494	0.512	0.507	0.388	0.190	0.396	0.195	0.152	0.192	0.144	0.195	0.118	0.068	0.067	0.081	0.039
	TMC	0.623	0.605	0.671	0.502	0.261	0.464	0.443	0.502	0.841	0.101	0.842	0.217	0.084	0.084	0.101	0.041
	TLRLF4	0.802	0.902	0.879	0.799	0.277	0.445	0.419	**0.799**	0.271	0.857	0.282	0.293	0.099	0.118	0.122	0.049
	Orth-NTF	0.807	0.826	0.807	0.712	0.504	0.784	0.715	0.712	0.463	0.647	0.485	0.351	–	–	–	–
	RAV	0.332	0.354	0.366	0.231	0.170	0.412	0.399	0.143	0.161	0.019	0.166	0.109	0.082	0.110	0.107	0.042
Continual	Adapt	0.394	0.343	0.394	0.378	0.093	0.225	0.128	0.008	0.131	0.009	0.135	0.101	0.034	0.014	0.034	0.033
	ASIA	0.271	0.203	0.301	0.168	0.271	0.203	0.129	0.026	0.056	0.147	0.114	0.122	–	–	–	–
	CDIMVC	0.629	0.562	0.640	0.468	0.629	0.562	0.442	0.217	0.236	0.466	0.599	0.401	–	–	–	–
	INMKC	0.503	0.474	0.565	0.381	0.503	0.474	0.471	0.287	0.291	0.483	0.835	0.701	0.089	0.102	0.111	0.043
	ACMVC	0.564	0.495	0.595	0.422	0.292	0.508	0.498	0.231	0.682	0.461	0.682	0.489	0.095	0.110	0.116	0.047
	CMVC	0.413	0.445	0.471	0.298	0.413	0.445	0.325	0.089	0.138	0.321	0.116	0.113	0.070	0.075	0.083	0.043
	LAIMVC	0.551	0.506	0.590	0.424	0.551	0.506	0.416	0.209	**0.978**	0.941	**0.978**	**0.956**	0.080	0.084	0.100	0.039
	PC-PID	**0.972**	**0.965**	**0.975**	**0.961**	**0.633**	**0.851**	**0.851**	0.533	**0.978**	**0.942**	**0.978**	**0.956**	**0.464**	**0.511**	**0.490**	**0.330**

Table 1 reports the results in terms of ACC, NMI, Purity, and F-score. The main observations are as follows.

- **Overall performance.** PC-PID achieves the best or competitive results on most dataset-metric cases, demonstrating its effectiveness across diverse datasets and evaluation criteria.
- **Comparison with continual methods.** On Reuters-1200, PC-PID improves ACC, NMI, Purity, and F-score over the best continual baseline by 0.393, 0.336, 0.393, and 0.331, respectively. The corresponding improvements on Scene15 are 0.343, 0.403, 0.335, and 0.493, showing its advantage in continually integrating newly arriving views.
- **Comparison with static methods.** Despite accessing views sequentially, PC-PID exceeds the best full-view static baseline on Reuters-1200 by 0.165, 0.121, 0.165, and 0.196, and on Scene15 by 0.165, 0.063, 0.096, and 0.162, respectively. This demonstrates that PC-PID maintains strong clustering performance under the continual setting.

Table 2: Ablation results averaged over eight datasets. The best and second-best results are indicated by boldface and underlining, respectively. Values in parentheses denote the absolute performance drops relative to PC-PID.

Variant	ACC	NMI	Purity	F
w/o CKT	<u>0.835</u> (↓0.027)	<u>0.827</u> (↓0.038)	<u>0.863</u> (↓0.030)	<u>0.790</u> (↓0.022)
w/o PTF	0.585 (↓0.277)	0.490 (↓0.375)	0.613 (↓0.280)	0.503 (↓0.309)
w/o CKT & PTF	0.581 (↓0.281)	0.497 (↓0.368)	0.611 (↓0.282)	0.506 (↓0.306)
Distance Transfer	0.587 (↓0.275)	0.512 (↓0.353)	0.623 (↓0.270)	0.505 (↓0.307)
PC-PID	**0.862**	**0.865**	**0.893**	**0.812**

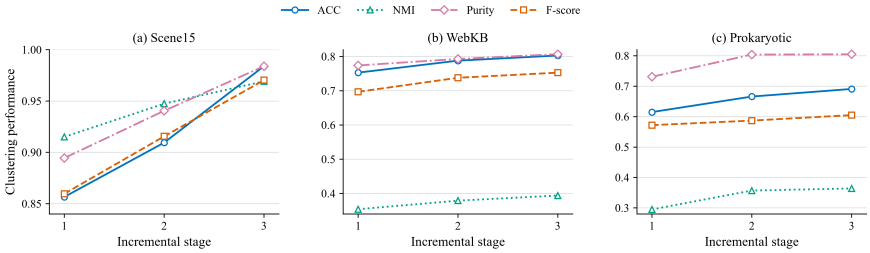

Figure 2: Progressive clustering performance of PC-PID as new views arrive on (a) Scene15, (b) WebKB, and (c) Prokaryotic. Each panel uses an independent vertical-axis range for readability.

5.3 Ablation Study (Q2)

We construct four variants: **w/o CKT**, **w/o PTF**, **w/o CKT & PTF**, and **Distance Transfer**. The last variant replaces the HSIC-based contrastive knowledge transfer with a conventional distance constraint. Table 2 reports the average results over eight datasets, with the absolute drops relative to PC-PID shown in parentheses.

The **w/o CKT** variant exhibits relatively small decreases of 0.022–0.038 and remains second best on all four metrics. In contrast, removing PTF causes substantially larger decreases of 0.277–0.375, while removing both components produces similarly severe degradation. These results show that PTF plays a central role in progressively integrating view-specific diversity, while CKT further improves the preservation and transfer of view-shared structures. Distance Transfer also causes large decreases of 0.270–0.353 and performs considerably worse than **w/o CKT**. This confirms that pointwise distance constraints provide limited semantic guidance, whereas the proposed HSIC-based formulation enables more effective knowledge transfer by preserving consistency dependence and suppressing redundant diversity dependence.

5.4 Progressive Effect Analysis (Q3)

Figure 2 shows the progressive performance of PC-PID on Scene15, WebKB, and Prokaryotic. All four clustering metrics improve as new views arrive on the three datasets, demonstrating that PC-PID progressively incorporates complementary information while preserving the shared structures learned at previous stages. Scene15 exhibits the most pronounced overall improvement, particularly in ACC, Purity, and F-score, while WebKB and Prokaryotic also show consistent gains across the incremental stages.

We further evaluate all five remaining view-arrival orders on Scene15. For every evaluated order, the final-stage results exceed their corresponding first-stage values on all four metrics. This confirms that the progressive behavior of PC-PID is not restricted to a particular view sequence. Detailed results are provided in the supplementary material.

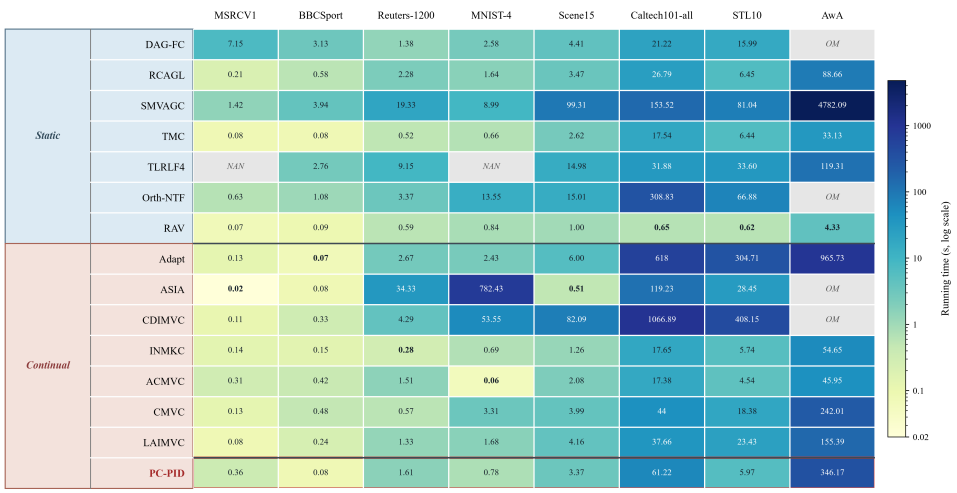

Figure 3: Running time (in seconds) of the compared methods on eight datasets. Colors are normalized using a logarithmic scale, with lighter colors indicating shorter running times. Gray blocks denote *NAN* or *OM* results, and the lowest running time on each dataset is highlighted in bold.

5.5 Time Efficiency Analysis (Q4)

Figure 3 compares the running time of all methods on eight datasets using a logarithmic color scale. PC-PID successfully completes clustering on all datasets and exhibits competitive efficiency on BBCSport, MNIST-4, Scene15, and STL10. Although its running time increases on the larger Caltech101-all and AwA datasets, it remains lower than that of several computationally expensive baselines.

Together with the clustering results in Table 1, these observations demonstrate that PC-PID achieves a favorable balance between clustering performance and computational cost. This efficiency benefits from the local two-view tensor construction, which only fuses the diversity representations of adjacent views instead of repeatedly constructing a tensor over the complete historical view set.

5.6 Additional Analysis (Q5–Q6)

PC-PID achieves stable performance under several neighboring settings of λ_1, λ_2, and m. Moreover, its representation-change loss approaches zero within approximately 15 iterations, demonstrating stable and rapid convergence. Complete parameter-sensitivity and convergence results are provided in Appendix F.

6 Conclusion

In this paper, we proposed PC-PID for continual multi-view clustering, aiming to handle continually arriving views without retraining from scratch. PC-PID decomposes multi-view representations into consistency and diversity components to capture view-shared cluster structures and view-specific complementary information, respectively. Based on this decomposition, contrastive knowledge transfer regards current and historical consistency representations as positive knowledge pairs while treating adjacent diversity representations as negative pairs, thereby preserving common knowledge and reducing redundant dependence. Progressive tensor fusion further organizes adjacent diversity representations into a local tensor and exploits their high-order correlations under a low-rank constraint. Experiments on eight benchmark datasets against 14 static and continual methods demonstrate the effectiveness and efficiency of PC-PID.

AI USE STATEMENT

In this work, we used generative AI tools (specifically ChatGPT) to assist with English translation and to improve the readability, grammar, and phrasing of the manuscript. We also used these tools to assist with manuscript organization, the presentation and interpretation of experimental results, the modification of scientific figures, and LaTeX formatting. We did not use generative AI tools to generate novel scientific ideas, design the methodology, conduct experiments, or generate experimental data or results. We have reviewed all AI-assisted work. Specifically, the authors carefully proofread and manually verified all AI-assisted content to ensure that it accurately reflects our original scientific intent, methodology, and findings without introducing inaccurate or unsupported claims. We take responsibility for the final content of this work, including all text, claims, and artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

The model formulation and optimization framework are described in Sections 3–4. The complete augmented Lagrangian, closed-form updates, initialization, stopping criteria, and stage-wise algorithm are provided in Appendix B. Dataset statistics, evaluation metrics, parameter configurations, the convergence criterion, and view-arrival-order sensitivity are reported in the supplementary material. The submitted code contains the preprocessing, training, evaluation, and plotting routines used in the experiments.

REFERENCES

Massih-Reza Amini, Nicolas Usunier, and Cyril Goutte. Learning from multiple partially observed views: An application to multilingual text categorization. In *Advances in Neural Information Processing Systems*, volume 22, pp. 28–36, 2009.

Konstantinos Bousmalis, George Trigeorgis, Nathan Silberman, Dilip Krishnan, and Dumitru Erhan. Domain separation networks. In *Advances in Neural Information Processing Systems*, volume 29, pp. 343–351, 2016.

Bing Cai, Xiaoli Wang, Gui-Fu Lu, and Zechao Li. Multi-hierarchical contrastive spectral fusion for multi-view clustering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 39617–39626, June 2026.

Xiaochun Cao, Changqing Zhang, Huazhu Fu, Si Liu, and Hua Zhang. Diversity-induced multi-view subspace clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 586–594, 2015.

Jingyi Chen, Zhibin Dong, Tiejun Li, and Yibo Han. Dc-span: A dual contrastive attention network for multi-view clustering. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(24): 20127–20135, 2026. doi: 10.1609/aaai.v40i24.39099.

Yawei Chen et al. Dual anchor graph fuzzy clustering for multi-view data. *IEEE Transactions on Fuzzy Systems*, 2025.

Adam Coates, Honglak Lee, and Andrew Y. Ng. An analysis of single-layer networks in unsupervised feature learning. In *Proceedings of the 14th International Conference on Artificial Intelligence and Statistics*, pp. 215–223, 2011.

Corinna Cortes, Mehryar Mohri, and Afshin Rostamizadeh. Algorithms for learning kernels based on centered alignment. *Journal of Machine Learning Research*, 13(28):795–828, 2012.

Matthias De Lange, Rahaf Aljundi, Marc Masana, Sarah Parisot, Xu Jia, Ales Leonardis, Gregory Slabaugh, and Tinne Tuytelaars. A continual learning survey: Defying forgetting in classification tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(7):3366–3385, 2022. doi: 10.1109/TPAMI.2021.3057446.

Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshop*, pp. 178–186, 2004.

Songhe Feng, Weixuan Liang, et al. Incremental multi-view clustering: Exploring stream-view correlations to learn consistency and diversity. *IEEE Transactions on Knowledge and Data Engineering*, 2025a.

Yu Feng, Weixuan Liang, et al. Incremental nyström-based multiple kernel clustering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 16613–16621, 2025b. doi: 10.1609/aaai.v39i16.33825.

Yu Feng et al. A lightweight anchor-based incremental framework for multi-view clustering. In *Proceedings of the 32nd ACM International Conference on Multimedia*, 2024.

Derek Greene and Pádraig Cunningham. Practical solutions to the problem of diagonal dominance in kernel document clustering. In *Proceedings of the 23rd International Conference on Machine Learning*, pp. 377–384, 2006. doi: 10.1145/1143844.1143892.

Arthur Gretton, Olivier Bousquet, Alex Smola, and Bernhard Schölkopf. Measuring statistical dependence with hilbert-schmidt norms. In *Proceedings of the 16th International Conference on Algorithmic Learning Theory*, pp. 63–77. Springer, 2005.

Ruiming Guo, Mouxing Yang, Yijie Lin, Xi Peng, and Peng Hu. Robust contrastive multi-view clustering against dual noisy correspondence. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/dbe81b08f7dc4dd8b43bc62dedfd9662-Abstract-Conference.html.

Shizhe Hu, Binyan Tian, Weibo Liu, and Yangdong Ye. Self-supervised trusted contrastive multi-view clustering with uncertainty refined. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 17305–17313, 2025. doi: 10.1609/aaai.v39i16.33902.

Jintian Ji, Songhe Feng, and Yidong Li. Tensorized unaligned multi-view clustering with multi-scale representation learning. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 1246–1256, 2024. doi: 10.1145/3637528.3671689.

Guanzhou Ke, Bo Wang, Xiaoli Wang, and Shengfeng He. Rethinking multi-view representation learning via distilled disentangling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 26774–26783, 2024.

Misha E. Kilmer and Carla D. Martin. Factorization strategies for third-order tensors. *Linear Algebra and its Applications*, 435(3):641–658, 2011. doi: 10.1016/j.laa.2010.09.020.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114(13):3521–3526, 2017. doi: 10.1073/pnas.1611835114.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97, pp. 3519–3529. PMLR, 2019.

Christoph H. Lampert, Hannes Nickisch, and Stefan Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 951–958, 2009. doi: 10.1109/CVPR.2009.5206594.

Svetlana Lazebnik, Cordelia Schmid, and Jean Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pp. 2169–2178, 2006. doi: 10.1109/CVPR.2006.68.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. doi: 10.1109/5.726791.

Jing Li, Quanxue Gao, Qianqian Wang, Ming Yang, and Wei Xia. Orthogonal non-negative tensor factorization based multi-view clustering. In *Advances in Neural Information Processing Systems*, volume 36, pp. 18186–18202, 2023.

Xingfeng Li, Hao Pan, Honglin Yuan, Yuan Sun, Xujian Zhao, Jiaqi Lin, and Zhenwen Ren. Anti-degradation lifelong multi-view clustering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8749–8759, June 2026.

Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(12):2935–2947, 2018. doi: 10.1109/TPAMI.2017.2773081.

Suyuan Liu, Qing Liao, Siwei Wang, et al. Robust and consistent anchor graph learning for multi-view clustering. *IEEE Transactions on Knowledge and Data Engineering*, 36(8), 2024. doi: 10.1109/TKDE.2024.3364663.

Zhen Long, Qiyuan Wang, Yazhou Ren, Yang Liu, and Ce Zhu. Tlrlf4mvc: Tensor low-rank and low-frequency for scalable multi-view clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 47(8):6900–6911, 2025.

Canyi Lu, Jiashi Feng, Yudong Chen, Wei Liu, Zhouchen Lin, and Shuicheng Yan. Tensor robust principal component analysis with a new tensor nuclear norm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(4):925–938, 2020. doi: 10.1109/TPAMI.2019.2891760.

Yiding Lu, Yijie Lin, Mouxing Yang, Dezhong Peng, Peng Hu, and Xi Peng. Decoupled contrastive multi-view clustering with high-order random walks. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(13):14193–14201, 2024. doi: 10.1609/aaai.v38i13.29330.

Caixuan Luo, Jie Xu, Yazhou Ren, Junbo Ma, and Xiaofeng Zhu. Simple contrastive multi-view clustering with data-level fusion. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, pp. 4697–4705, 2024. doi: 10.24963/ijcai.2024/519.

Shuangmei Peng, Zhe Chen, Tianyang Xu, and Xiaojun Wu. Relationship alignment for view-aware multi-view clustering. In *International Conference on Learning Representations*, 2026.

Qian Qu, Xinhang Wan, Jiyuan Liu, Xinwang Liu, and En Zhu. Anchor-guided sample-and-feature incremental alignment framework for multi-view clustering. *IEEE Transactions on Circuits and Systems for Video Technology*, 36(3):2882–2893, 2026.

Le Song, Alex Smola, Arthur Gretton, Justin Bedo, and Karsten M. Borgwardt. Feature selection via dependence maximization. *Journal of Machine Learning Research*, 13(47):1393–1434, 2012.

Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. In *Computer Vision–ECCV 2020*, pp. 776–794. Springer, 2020.

Xinhang Wan, Xinwang Liu, En Zhu, Jiyuan Liu, Jianping Yin, et al. Continual multi-view clustering. In *Proceedings of the 30th ACM International Conference on Multimedia*, pp. 3676–3684, 2022. doi: 10.1145/3503161.3547864.

Hua Wang, Feiping Nie, and Heng Huang. Multi-view clustering and feature learning via structured sparsity. In *Proceedings of the 30th International Conference on Machine Learning*, pp. 352–360, 2013.

Jing Wang and Songhe Feng. Contrastive and view-interaction structure learning for multi-view clustering. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, pp. 5055–5063, 2024. doi: 10.24963/ijcai.2024/559.

Jing Wang, Songhe Feng, Kristoffer Knutsen Wickstrøm, and Michael C. Kampffmeyer. Adaptcmvc: Robust adaption to incremental views in continual multi-view clustering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10285–10294, 2025.

Siwei Wang, Xinwang Liu, En Zhu, et al. Scalable and structural multi-view graph clustering with adaptive anchor fusion. *IEEE Transactions on Image Processing*, 2024.

Tingting Wu, Songhe Feng, and Jiazheng Yuan. Low-rank kernel tensor learning for incomplete multi-view clustering. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(14): 15952–15960, 2024. doi: 10.1609/aaai.v38i14.29526.

Deng Xu, Chao Zhang, Zechao Li, Chunlin Chen, and Huaxiong Li. Fast disentangled slim tensor learning for multi-view clustering. *IEEE Transactions on Multimedia*, 27:1254–1265, 2025. doi: 10.1109/TMM.2024.3521754.

Honglin Yuan, Xingfeng Li, Jian Dai, Xiaojian You, Yuan Sun, and Zhenwen Ren. Deep streaming view clustering. In *Proceedings of the 42nd International Conference on Machine Learning*, pp. 73526–73540, 2025.

Chao Zhang, Deng Xu, Xiuyi Jia, Chunlin Chen, and Huaxiong Li. Continual multi-view clustering with consistent anchor guidance. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, pp. 5434–5442, 2024. doi: 10.24963/ijcai.2024/601.

Zisen Zhang et al. Tensor multi-rank constraint guided anchor-wise adaptive multi-view clustering. *IEEE Transactions on Knowledge and Data Engineering*, 2026.

APPENDIX

A Dataset Details and Evaluation Protocol

A.1 Dataset Statistics

Table 3 summarizes the eight datasets used in the experiments, covering image, text, handwritten-digit, scene, and animal data. The adopted benchmark versions follow their established sources: MSRCV1 uses the multi-feature construction in (Wang et al., 2013), BBCSport is from (Greene & Cunningham, 2006), Reuters-1200 is sampled from the multilingual Reuters collection in (Amini et al., 2009), and MNIST-4 is a four-class subset of MNIST (LeCun et al., 1998) with the public multi-view features listed below. Scene15, Caltech101-all, STL10, and AwA are derived from (Lazebnik et al., 2006; Fei-Fei et al., 2004; Coates et al., 2011; Lampert et al., 2009), respectively.

A.2 Evaluation Protocol

Following the continual multi-view clustering protocol, each continual method receives the views in a fixed pre-specified order. The same order is used for PC-PID and all continual baselines. Static methods are trained once using the complete view set and serve as full-view references. Each method is independently run 10 times, and the mean performance is reported in the main paper.

The representation-learning procedure of PC-PID is deterministic under fixed parameters and a fixed view-arrival order. Randomness arises only from the random initialization of the final k-means clustering used for evaluation. We perform ten independent clustering evaluations, but do not assign explicit run-specific seeds such as `rng(r,'twister')`. Consequently, the reported protocol supports statistical replication of the aggregate results, whereas the exact initialization and result of each individual run may differ. The random-seed reproducibility protocol is therefore partial.

For preprocessing, each feature dimension of every view is normalized to unit ℓ_2 norm. The exact preprocessing routine is included in the submitted code appendix as `NormalizeData.m`. The code appendix also contains the complete training, evaluation, and plotting scripts, with comments that link the principal update steps to the corresponding equations in the main paper. All experiments are conducted in MATLAB R2024a on a Windows 11 workstation equipped with an Intel Core i7-14700KF CPU, 48 GB RAM, and an NVIDIA GeForce RTX 4060 Ti GPU with 16 GB memory.

Table 3: Statistics of the eight multi-view datasets.

Dataset	Samples	Classes	Views	Type
MSRCV1	210	7	6	Object image
BBCSport	544	5	2	News text
Reuters-1200	1,200	6	5	News text
MNIST-4	4,000	4	3	Handwritten digit
Scene15	4,485	15	3	Scene image
Caltech101-all	9,144	102	6	Object image
STL10	13,000	10	3	Object image
AwA	30,475	50	6	Animal image

A.3 Evaluation Metrics

Let y_i and $\hat{y}_i$ denote the ground-truth and predicted labels of the i-th sample, respectively, and let n_{ab} be the number of samples from class a assigned to cluster b. Denote the corresponding marginal counts by $n_a = \sum_b n_{ab}$ and $\hat{n}_b = \sum_a n_{ab}$. We use the following four metrics, for which larger values indicate better clustering.

Accuracy first uses the Hungarian algorithm to find the optimal permutation π between cluster and class labels:

$$\mathrm{ACC} = \frac{1}{n} \max_{\pi} \sum_{i=1}^{n} \mathbb{I}(y_i = \pi(\hat{y}_i)) \,. \tag{17}$$

Let Y and $\widehat{Y}$ be the two label random variables. Consistent with our released evaluation code, normalized mutual information is

$$\mathrm{NMI} = \frac{I(Y;\widehat{Y})}{\max\{H(Y), H(\widehat{Y})\}}, \tag{18}$$

where $I(Y;\widehat{Y}) = \sum_{a,b} \frac{n_{ab}}{n} \log \frac{n n_{ab}}{n_a \hat{n}_b}$ and $H(\cdot)$ denotes entropy. Purity is defined as

$$\mathrm{Purity} = \frac{1}{n} \sum_{b} \max_{a} n_{ab}. \tag{19}$$

For F-score, let TP be the number of sample pairs assigned to the same cluster and belonging to the same class, FP the number assigned together but belonging to different classes, and FN the number separated despite sharing a class. With $P = \mathrm{TP}/(\mathrm{TP}+\mathrm{FP})$ and $R = \mathrm{TP}/(\mathrm{TP}+\mathrm{FN})$, we compute

$$\mathrm{F} = \frac{2PR}{P+R}. \tag{20}$$

ACC measures label-assignment accuracy, NMI measures information shared by the two partitions, Purity measures within-cluster class homogeneity, and F balances pairwise precision and recall. Their complementary perspectives reduce the risk of drawing conclusions from a single clustering property.

B Complete Optimization Procedure

This section provides the complete optimization procedure for PC-PID, including initialization, closed-form updates, stopping criteria, and the stage-wise algorithm.

B.1 Initialization and Stopping Criteria

At the first stage, the consistency representation $\mathbf{C}^{(1)}$ and projection matrix $\mathbf{P}^{(1)}$ are initialized using the deterministic initialization adopted in our implementation and are then updated through orthogonal Procrustes iterations. The first-stage iterations terminate when

$$\frac{\left\| \mathbf{C}_{k+1}^{(1)} - \mathbf{C}_{k}^{(1)} \right\|_F^2}{\left\| \mathbf{C}_{k+1}^{(1)} \right\|_F^2} < 10^{-3}, \tag{21}$$

or when 20 iterations are reached.

At each subsequent stage, the local consistency, diversity, auxiliary, and multiplier variables are initialized to zero. The initial penalty parameter, growth factor, and upper bound are set to

$$\rho_0 = 10^{-3}, \qquad \eta = 2, \qquad \rho_{\max} = 10^{13}, \tag{22}$$

respectively. The local alternating iterations terminate when the LOSS value in Eq. (41) is below 10^{-3} or when 20 iterations are reached.

B.2 First-Stage Optimization

At the first stage, no historical knowledge is available. We therefore solve the reconstruction problem by alternately updating $\mathbf{P}^{(1)}$ and $\mathbf{C}^{(1)}$.

With $\mathbf{C}_k^{(1)}$ fixed, we compute the singular value decomposition

$$\mathbf{C}_k^{(1)\top}\mathbf{X}^{(1)} = \mathbf{U}_k^{(1)}\mathbf{\Sigma}_k^{(1)}\mathbf{V}_k^{(1)\top}. \tag{23}$$

The orthogonal projection matrix is then updated as

$$\mathbf{P}_{k+1}^{(1)} = \mathbf{U}_k^{(1)}\mathbf{V}_k^{(1)\top}. \tag{24}$$

With $\mathbf{P}_{k+1}^{(1)}$ fixed, the consistency representation is updated by

$$\mathbf{C}_{k+1}^{(1)} = \mathbf{X}^{(1)}\mathbf{P}_{k+1}^{(1)\top}. \tag{25}$$

Equations (24) and (25) are repeated until the stopping criterion in Eq. (21) is satisfied.

B.3 Subsequent-Stage Optimization

For stage $t \geq 2$, let $\mathcal{I}_t = \{t-1, t\}$. To decouple the tensor nuclear norm, we introduce the auxiliary tensor

$$\mathcal{G}^{(t)} = \mathrm{cat}\left(\mathbf{G}^{(t-1)}, \mathbf{G}^{(t)}\right), \tag{26}$$

together with the constraints

$$\mathbf{D}^{(i)} = \mathbf{G}^{(i)}, \qquad i \in \mathcal{I}_t. \tag{27}$$

The resulting augmented Lagrangian is

$$\begin{aligned}\mathcal{L}_\rho^{(t)} =& \mathcal{L}_{rec}^{(t)} + \lambda_1\mathcal{L}_{ckt}^{(t)} + \lambda_2\left\|\mathcal{G}^{(t)}\right\|_{\circledast} \\ &+ \sum_{i\in\mathcal{I}_t}\left[\left\langle \mathbf{W}^{(i)}, \mathbf{D}^{(i)} - \mathbf{G}^{(i)}\right\rangle + \frac{\rho}{2}\left\|\mathbf{D}^{(i)} - \mathbf{G}^{(i)}\right\|_F^2\right] \\ \text{s.t.} \quad & \mathbf{P}^{(i)}\mathbf{P}^{(i)\top} = \mathbf{I}, \quad i \in \mathcal{I}_t,\end{aligned} \tag{28}$$

where

$$\langle \mathbf{A}, \mathbf{B}\rangle = \mathrm{tr}\left(\mathbf{A}^\top\mathbf{B}\right) \tag{29}$$

denotes the Frobenius inner product, $\mathbf{W}^{(i)}$ is the Lagrange multiplier, and $\rho > 0$ is the penalty parameter.

Since the CKT term is non-convex, we employ a fixed-point alternating scheme. At the beginning of stage t, the historical consistency kernel and its associated normalization term are fixed as

$$\begin{aligned}\mathbf{K}_C &= \mathbf{H}\mathbf{C}^{(t-1)\top}\mathbf{C}^{(t-1)}\mathbf{H}, \\ p_C &= \mathrm{tr}\left(\mathbf{C}^{(t-1)\top}\mathbf{C}^{(t-1)}\mathbf{K}_C\right) + \epsilon.\end{aligned} \tag{30}$$

The variables in Eq. (28) are then updated alternately.

B.3.1 Update of the Projection Matrices

With the other variables fixed, the subproblem associated with $\mathbf{P}^{(i)}$ is an orthogonal Procrustes problem. For every $i \in \mathcal{I}_t$, we compute

$$\left(\mathbf{C}_k^{(t)} + \mathbf{D}_k^{(i)}\right)^\top \mathbf{X}^{(i)} = \mathbf{U}_k^{(i)} \mathbf{\Sigma}_k^{(i)} \mathbf{V}_k^{(i)\top}. \tag{31}$$

The projection matrix is updated as

$$\mathbf{P}_{k+1}^{(i)} = \mathbf{U}_k^{(i)} \mathbf{V}_k^{(i)\top}, \qquad i \in \mathcal{I}_t. \tag{32}$$

B.3.2 Update of the Diversity Representations

For each $i \in \mathcal{I}_t$, let $\bar{i}$ denote the other index in $\mathcal{I}_t$. The local diversity kernel is defined as

$$\mathbf{K}_{D,k}^{(i)} = \mathbf{H}\mathbf{D}_k^{(\bar{i})\top} \mathbf{D}_k^{(\bar{i})} \mathbf{H}, \qquad i \neq \bar{i}. \tag{33}$$

With the remaining variables fixed, setting the derivative of the augmented Lagrangian with respect to $\mathbf{D}^{(i)}$ to zero gives

$$\begin{aligned} \mathbf{D}_{k+1}^{(i)} = &\left(2\mathbf{X}^{(i)} \mathbf{P}_{k+1}^{(i)\top} - 2\mathbf{C}_k^{(t)} + \rho \mathbf{G}_k^{(i)} - \mathbf{W}_k^{(i)}\right) \\ &\times \left((2+\rho)\mathbf{I} + \lambda_1 \mathbf{K}_{D,k}^{(i)}\right)^\dagger, \qquad i \in \mathcal{I}_t, \end{aligned} \tag{34}$$

where $(\cdot)^\dagger$ denotes the Moore–Penrose inverse. After updating the diversity representations, the local diversity kernels are recomputed using Eq. (33).

B.3.3 Update of the Consistency Representation

With the updated projection matrices and diversity representations fixed, the consistency representation is updated as

$$\mathbf{C}_{k+1}^{(t)} = \left[\sum_{i \in \mathcal{I}_t} \left(\mathbf{X}^{(i)} \mathbf{P}_{k+1}^{(i)\top} - \mathbf{D}_{k+1}^{(i)}\right)\right] \times \left(\mathbf{I} - \frac{\lambda_1}{p_C^2} \mathbf{K}_C\right)^\dagger. \tag{35}$$

Here, $\mathbf{K}_C$ and p_C are determined by the historical consistency representation $\mathbf{C}^{(t-1)}$ and remain fixed throughout the local optimization at stage t.

B.3.4 Update of the Auxiliary Tensor

Let

$$\begin{aligned} \mathcal{W}_k^{(t)} &= \mathrm{cat}\left(\mathbf{W}_k^{(t-1)}, \mathbf{W}_k^{(t)}\right), \\ \mathcal{D}_{k+1}^{(t)} &= \mathrm{cat}\left(\mathbf{D}_{k+1}^{(t-1)}, \mathbf{D}_{k+1}^{(t)}\right). \end{aligned} \tag{36}$$

The subproblem associated with $\mathcal{G}^{(t)}$ is

$$\min_{\mathcal{G}^{(t)}} \quad \lambda_2 \left\|\mathcal{G}^{(t)}\right\|_{\circledast} + \frac{\rho}{2} \left\|\mathcal{G}^{(t)} - \left(\mathcal{D}_{k+1}^{(t)} + \frac{1}{\rho}\mathcal{W}_k^{(t)}\right)\right\|_F^2. \tag{37}$$

It is solved using tensor singular value thresholding:

$$\mathcal{G}_{k+1}^{(t)} = \mathrm{tSVT}_{\lambda_2/\rho}\left(\mathcal{D}_{k+1}^{(t)} + \frac{1}{\rho}\mathcal{W}_k^{(t)}\right). \tag{38}$$

B.3.5 Update of the Multipliers and Penalty Parameter

The Lagrange multipliers are updated by

$$\mathbf{W}_{k+1}^{(i)} = \mathbf{W}_k^{(i)} + \rho\left(\mathbf{D}_{k+1}^{(i)} - \mathbf{G}_{k+1}^{(i)}\right), \qquad i \in \mathcal{I}_t. \tag{39}$$

The penalty parameter is increased adaptively:

$$\rho \leftarrow \min\left(\eta\rho, \rho_{\max}\right), \tag{40}$$

where $\eta > 1$ is the penalty growth factor and $\rho_{\max}$ is its upper bound.

B.4 Optimization Algorithm

Algorithm 1 summarizes the complete stage-wise optimization procedure.

Algorithm 1 Stage-wise optimization of PC-PID

Require: Views $\{\mathbf{X}^{(t)}\}_{t=1}^{V}$, latent dimension m, λ_1, λ_2, ρ_0, η, and $\rho_{\max}$.
Ensure: Final clustering representation $\mathbf{F}$.

```
Initialize C^(1) and P^(1).
repeat
    Update P^(1) using Eq. (24).
    Update C^(1) using Eq. (25).
until Eq. (21) is satisfied or 20 iterations are reached
for t = 2 to V do
    Set I_t = {t − 1, t}.
    Initialize C^(t), D^(i), G^(i), and W^(i) for i ∈ I_t.
    Set ρ ← ρ_0.
    Compute K_C and p_C using Eq. (30).
    repeat
        for i ∈ I_t do
            Update P^(i) using Eq. (32).
        end for
        for i ∈ I_t do
            Compute K^(i)_{D,k} using Eq. (33).
            Update D^(i) using Eq. (34).
        end for
        Recompute the local diversity kernels K^(i)_{D,k+1}.
        Update C^(t) using Eq. (35).
        Construct D^(t)_{k+1} and W^(t)_k using Eq. (36).
        Update G^(t) using Eq. (38).
        for i ∈ I_t do
            Update W^(i) using Eq. (39).
        end for
        Update ρ using Eq. (40).
        Compute LOSS^(k+1) using Eq. (41).
    until LOSS^(k+1) < 10^−3 or 20 iterations are reached
    Retain C^(t) as the historical consistency representation for stage t + 1.
end for
F = C^(V) + (1/V) Σ_{i=1}^{V} D^(i).
return F.
```

C Parameter Configurations

C.1 Search Space and Selection Criterion

For PC-PID, we adopt a two-step parameter-search strategy. The coarse search evaluates eight latent dimensions,

$$m \in \{k, 2k, 3k, 4k, 5k, 6k, 7k, 8k\},$$

and eight candidate values for each regularization parameter,

$$\lambda_1, \lambda_2 \in \left\{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 10^{0}, 10^{1}, 10^{2}, 10^{3}\right\}.$$

We subsequently evaluate five locally refined values around the best coarse value of each parameter, while restricting the candidates to $m \in [k, 8k]$ and $\lambda_1, \lambda_2 \in [10^{-4}, 10^{3}]$.

Each candidate configuration is evaluated using the ten repeated evaluations described in Appendix A.2. The configuration with the highest mean ACC is selected. Ties are resolved first according to mean NMI and then in favor of the smaller latent dimension. Parameter selection is performed independently for each dataset, after which the selected configuration is fixed for all reported experiments on that dataset.

Table 4 lists the resulting parameter configurations used in the main experiments.

Table 4: Dataset-specific parameter configurations used by PC-PID.

Dataset	m	λ_1	λ_2
MSRCV1	40	$5e$-4	56
BBCSport	6	$4e$-3	120
Reuters-1200	30	0.1	100
MNIST-4	9	0.12	90
Scene15	75	0.01	100
Caltech101-all	408	0.001	100
STL10	60	10	0.1
AwA	400	10	100

For the Scene15 sensitivity analysis, both λ_1 and λ_2 are varied over

$$\left\{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 10^{0}, 10^{1}, 10^{2}, 10^{3}\right\},$$

while m is varied over $\{k, 2k, 3k, 4k, 5k, 6k, 7k, 8k\}$, where k denotes the number of clusters. When studying one parameter group, the remaining parameters are fixed. This sensitivity analysis uses the same coarse parameter grids defined above.

C.2 Baseline Configurations

All 14 baselines are evaluated using their official implementations and the dataset-specific settings recommended by their authors, without changing the model architecture, objective, or stopping criterion. When an official implementation provides several candidate settings, we retain its published search range and apply the same ten-run mean-ACC selection criterion described above. The exact dataset-specific hyperparameter settings are specified in the corresponding experiment scripts included in the submitted code appendix.

D Detailed Convergence Criterion

Following the representation-change criterion used in the main paper, the LOSS value measures the relative changes of the consistency and diversity representations between two successive iterations:

$$\mathrm{LOSS}^{(k+1)} = \frac{\left\|\mathbf{C}_{k+1}^{(t)} - \mathbf{C}_{k}^{(t)}\right\|_F^2}{\left\|\mathbf{C}_{k+1}^{(t)}\right\|_F^2} + \sum_{i \in \mathcal{I}_t} \frac{\left\|\mathbf{D}_{k+1}^{(i)} - \mathbf{D}_{k}^{(i)}\right\|_F^2}{\left\|\mathbf{D}_{k+1}^{(i)}\right\|_F^2}. \tag{41}$$

A smaller LOSS value indicates that both consistency and diversity representations change less between successive iterations. The convergence curve in the main paper approaches zero after approximately 15 iterations, showing that the learned representations become stable rapidly in practice.

E View-Arrival Order Sensitivity

The three Scene15 views are denoted by 1, 2, and 3, with feature dimensions 1800, 1180, and 1240, respectively. The original arrival order is $3 \to 2 \to 1$. To examine whether the progressive behavior depends on a particular order, we further evaluate all five remaining permutations while keeping the model parameters and evaluation settings unchanged. Variants 1–5 correspond to $1 \to 2 \to 3$, $1 \to 3 \to 2$, $2 \to 1 \to 3$, $2 \to 3 \to 1$, and $3 \to 1 \to 2$, respectively. Table 5 reports the clustering performance at each incremental stage.

Table 5: Representative progressive clustering results on Scene15 under all six view-arrival orders. The best result for each metric within each order is highlighted in bold.

Setting	Stage	ACC	NMI	Purity	F
Original	1	0.8566	0.9151	0.8945	0.8597
	2	0.9095	0.9474	0.9405	0.9156
	3	**0.9839**	**0.9692**	**0.9839**	**0.9703**
Variant 1	1	0.8852	0.9193	0.9001	0.8575
	2	0.8876	0.9136	0.9093	0.8526
	3	**0.8887**	**0.9198**	**0.9193**	**0.8700**
Variant 2	1	0.8192	0.8840	0.8660	0.7783
	2	0.8219	0.8817	0.8627	0.8146
	3	**0.8680**	**0.8958**	**0.8800**	**0.8273**
Variant 3	1	0.8283	0.8950	0.8774	0.8296
	2	0.9104	0.9254	0.9309	0.9025
	3	**0.9699**	**0.9419**	**0.9699**	**0.9443**
Variant 4	1	0.9128	0.9456	0.9396	0.9172
	2	0.9322	0.9279	0.9324	0.9206
	3	**0.9744**	**0.9499**	**0.9744**	**0.9528**
Variant 5	1	0.8825	0.9272	0.9019	0.8632
	2	0.9037	0.9434	0.9242	0.9107
	3	**0.9828**	**0.9666**	**0.9828**	**0.9690**

For every arrival order, all four metrics attain their highest values at the final stage, although minor intermediate fluctuations occur for some orders. This consistent final-stage improvement shows that PC-PID can progressively integrate complementary information from incoming views under different view permutations. Four orders achieve final ACC and Purity above 0.96, together with NMI and F-score above 0.94. The two orders initialized by view 1 obtain relatively lower final performance, indicating that the initial view affects the quality of the historical knowledge and hence the magnitude of subsequent improvement. Overall, PC-PID exhibits consistent progressive behavior across all tested orders, while retaining the expected sensitivity to view order.

F Parameter Sensitivity and Convergence Analyses

F.1 Parameter Sensitivity Analysis (Q5)

We investigate the sensitivity of PC-PID to the regularization parameters λ_1 and λ_2 and the latent dimension m on Scene15. Both λ_1 and λ_2 are selected from

$$\left\{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 10^{0}, 10^{1}, 10^{2}, 10^{3}\right\}.$$

As shown in Figure 4(a), several neighboring parameter combinations achieve similarly high ACC values, indicating that PC-PID does not require precise tuning within an appropriate parameter range. In contrast, excessively large or imbalanced regularization weights generally result in lower performance. This observation suggests that balancing consistency preservation and progressive diversity integration is important for obtaining reliable clustering representations.

Figure 4(b) reports the sensitivity to the latent dimension m, which varies from k to $8k$, where k denotes the number of clusters. ACC increases from 0.833 at $m = k$ to its highest value of 0.984 at $m = 5k$. Although a fluctuation occurs at $m = 3k$, all settings from $4k$ to $8k$ achieve ACC values above 0.90. These results indicate that an insufficient latent dimension may restrict the representation capacity, whereas the performance remains relatively stable once m is sufficiently large. Accordingly, the selected latent dimension for Scene15 is $m = 5k$.

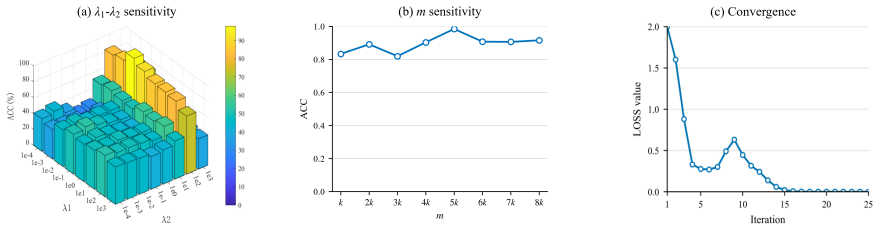

Figure 4: Parameter sensitivity and convergence analyses of PC-PID on Scene15: (a) sensitivity to λ_1 and λ_2, (b) sensitivity to the latent dimension m, and (c) convergence behavior measured by the LOSS value.

F.2 Convergence Analysis (Q6)

Following the representation-change criterion in (Xu et al., 2025), we evaluate the convergence behavior of PC-PID on Scene15 using the LOSS value defined in Eq. (41). This criterion measures the relative changes in the consistency and diversity representations between two successive iterations.

As shown in Figure 4(c), the LOSS value decreases rapidly from approximately 2.0 at the first iteration to below 0.35 within the first four iterations. A temporary increase occurs around iterations 8–10 because of the alternating block updates and adaptive penalty adjustment. The LOSS value subsequently decreases, approaches zero at approximately the 15th iteration, and remains stable thereafter. This behavior demonstrates that the learned consistency and diversity representations stabilize within a small number of iterations.