CROSS-VIEW PROTOTYPE CALIBRATION VIA RECIPROCAL CYCLIC TRANSITIONS FOR INCOMPLETE AND NOISY MULTI-VIEW CLUSTERING

Anonymous authors
Paper under double-blind review

ABSTRACT

Multi-view clustering aims to uncover a consensus cluster structure from heterogeneous views of the same objects. Prototype-based methods summarize the clusters of each view with semantic prototypes and align them across views. However, missing views and feature noise distort local neighborhood structures and shift view-specific prototypes, rendering cross-view correspondences ambiguous and prone to many-to-one matches. Establishing reliable and reciprocal prototype correspondences under such degradation therefore remains challenging. We propose ReCAP, an imputation-free framework for Reciprocal Cyclic cAlibration of cross-view Prototypes under incomplete and noisy settings. ReCAP first applies mutual k-nearest-neighbor graph contrastive learning to preserve reliable local neighborhoods and stabilize prototype estimation. It then constructs sample-mediated cyclic transitions that propagate probability mass from each source prototype through the samples and prototypes of another view and back to the source view. Maximizing the probability of returning to the source prototype couples forward and backward transitions; we prove that, in the zero-loss limit, the two directional transitions reduce to a permutation and its inverse, i.e., a reciprocal one-to-one correspondence. A visitation regularizer further prevents the returned mass from collapsing onto a few prototypes. Experiments on six benchmark datasets demonstrate competitive or superior clustering performance under incomplete, noisy, and compound degradation settings, while further analyses support the effectiveness of cyclic prototype calibration. Our code is available at `https://anonymous.4open.science/r/ReCAP-0301`.

1 INTRODUCTION

Multi-view clustering (MVC) aims to uncover a clustering structure shared across heterogeneous views of the same objects in an unsupervised manner (Chowdhury et al., 2025). Existing methods, however, commonly assume that multi-view observations are complete and reliable. In practice, sensor failures, transmission errors, and storage corruption can result in missing views and feature corruption (Wang et al., 2026c; Yu et al., 2025; Hu et al., 2023), and the two forms of degradation may co-occur. Such degradations distort intra-view structures and cross-view correspondences, hindering reliable cluster-level alignment.

These challenges have motivated extensive research on incomplete multi-view clustering (IMVC) and noise-robust multi-view clustering (NRMVC). Some studies mitigate the effects of data degradation on clustering through instance-level representation learning, information recovery, or reliability estimation (Lin et al., 2022; Yang et al., 2025; He et al., 2026). At the cluster level, prototype-based methods have attracted considerable attention for explicitly representing view-specific cluster structures through semantic centers and establishing cross-view cluster-level correspondences. For example, Li et al. (2023) learns and enhances view-specific prototypes through a dual-attention mechanism, and uses them to recover missing-view data. Jin et al. (2023) establishes cross-view prototype correspondences through permutation-based optimal matching, while Yuan et al. (2025) introduces a robust prototype contrastive objective to mitigate the adverse effects of prototype mismatch on representation learning. Despite this progress, data degradation may still perturb view-specific prototype distributions, making cross-view cluster-level correspondences ambiguous. Similarity-driven soft

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.

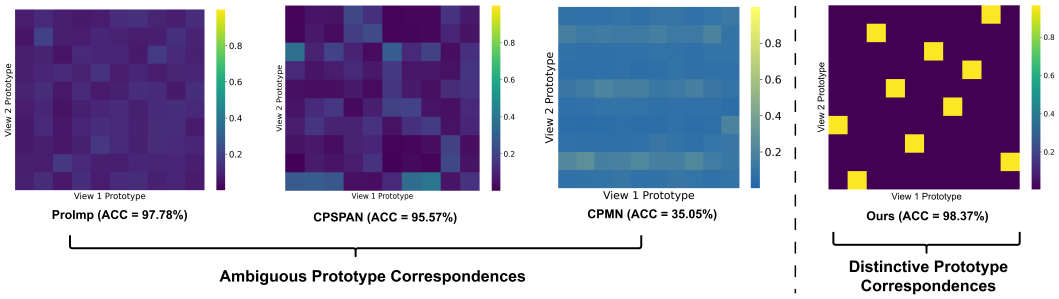


Figure 1: Cross-view prototype affinity matrices on Hdigit with a missing rate of 0.5, computed using row-wise softmax over cosine similarities with a shared temperature and color scale. The displayed baselines exhibit more diffuse or ambiguous affinity patterns, whereas our method produces more concentrated and distinctive correspondences.

alignment does not necessarily enforce reciprocal or globally one-to-one correspondences, potentially producing ambiguous or many-to-one matches. Permutation-constrained matching enforces one-to-one assignments, but its reliance on current pairwise prototype relations can still lead to erroneous correspondences under data degradation. As illustrated in Figure 1, representative methods exhibit diffuse or non-exclusive cross-view prototype correspondence patterns under data degradation. More generally, neither paradigm explicitly verifies the reciprocity of cross-view prototype correspondences through a closed-loop transition.

To address these limitations, we propose ReCAP, a reciprocal cyclic prototype calibration framework for incomplete and noisy multi-view data. ReCAP first employs mutual k-nearest-neighbor graph contrastive learning to select reliable neighborhood relations and preserve local structure, thereby providing stable representations for prototype estimation. It then treats view-specific prototypes as semantic anchors and constructs sample-mediated cross-view cyclic transitions: probability mass moves from a source prototype through target-view samples to a target prototype, and subsequently returns through source-view samples to the initial prototype. By maximizing the cyclic return probability and regularizing the marginal visitation distribution, ReCAP promotes reciprocal prototype correspondences while discouraging degenerate probability concentration. Unlike pairwise-similarity-based or single-step matching schemes, ReCAP optimizes round-trip consistency over bidirectional prototype transitions. Together, these components preserve intra-view neighborhood structures and uncover a shared clustering structure across views.

The main contributions of this work are summarized as follows:

- We investigate cross-view prototype alignment under view missingness and feature noise, identifying ambiguity in prototype correspondences and emphasizing the role of reciprocity in reliable cluster-level alignment.
- We propose ReCAP, an imputation-free multi-view clustering framework that calibrates prototypes through sample-mediated cross-view cyclic transitions. By maximizing the probability of returning to the source prototype, ReCAP encourages reciprocal correspondences, supported by visitation regularization.
- Extensive experiments on six widely used multi-view benchmark datasets demonstrate superior and robust clustering performance under incomplete and noisy settings and their combination. Further analyses validate the effectiveness of reciprocal cyclic calibration in improving cross-view prototype alignment.

2 Related Work

2.1 Incomplete and Noise-Robust Multi-view Clustering

Incomplete multi-view clustering (IMVC) aims to discover a consensus clustering structure from partially observed multi-view data (Chowdhury et al., 2025; Wang et al., 2026a). Existing methods broadly follow either imputation-based or imputation-free paradigms. The former reconstruct missing observations, as exemplified by Zhang et al. (2025), which uses limited paired samples to guide

diffusion-based view generation and jointly learns instance- and category-level interactions. The latter learns directly from available observations: Dai et al. (2025) constructs consensus prototypes in a shared semantic space, while Chen et al. (2026) combines variational information maximization, local geometry preservation, and Gramian-based alignment. Extending IMVC to compound degradation, He et al. (2026) employs global–local graph-guided contrastive learning to address both missing views and feature noise. Despite this progress, recovering missing views from scarce paired observations may propagate generation errors. Imputation-free methods avoid explicit recovery but must still estimate reliable cross-view relations from limited observations. Under compound missingness and feature corruption, reliably calibrating view-specific cluster structures without reconstructing missing views remains challenging.

2.2 Prototype Learning and Cross-View Alignment

Prototype-based methods characterize latent cluster structures using semantic centers and exploit them for view completion or cross-view alignment. Li et al. (2023) learns view-specific prototypes through dual attention and reconstructs missing views from sample–prototype relations. Yu et al. (2025) constructs view-specific multi-scale sample–prototype bipartite similarities and coordinates different views through shared consensus graphs, without explicitly matching prototypes across views. For direct prototype alignment, Jin et al. (2023) establishes cross-view correspondences through permutation-constrained optimal matching, Yuan et al. (2025) introduces robust prototypical contrastive learning to mitigate prototype mismatch, and Wang et al. (2026b) matches prototype graphs learned from correspondence-free representations. Despite their effectiveness, similarity- and contrastive-based objectives may produce ambiguous or many-to-one soft correspondences, whereas permutation-based assignments depend on potentially unreliable matching costs. More generally, these methods do not jointly constrain prototype-level cyclic returnability and the non-degeneracy of the resulting return distribution.

3 Method

3.1 Problem Definition

Given a multi-view dataset subject to view missingness and feature noise $\mathcal{X} = \{\mathbf{X}^{(v)} \in \mathbb{R}^{N\times d_v}\}_{v=1}^{V}$ with N instances and V views, $\mathbf{x}_i^{(v)} \in \mathbb{R}^{d_v}$ denotes the i-th instance in the v-th view. We introduce an observation indicator matrix $\mathbf{W} \in \{0,1\}^{N\times V}$, where $W_i^{(v)} = 1$ if $\mathbf{x}_i^{(v)}$ is observed and $W_i^{(v)} = 0$ otherwise. Each observed instance may additionally be corrupted by feature noise. We assume that every instance is observed in at least one view, i.e., $\sum_{v=1}^{V} W_i^{(v)} \geq 1$. Given the number of clusters K, our goal is to infer a unified partition $\mathbf{y} \in \{1,\ldots,K\}^N$ from the available observations without imputing the missing views. As illustrated in Figure 2, ReCAP comprises three modules: the View-Specific Reconstruction Module (VRM), the Prototype Aggregation Module (PAM), and the Prototype Cyclic Calibration Module (PCC). VRM learns view-specific representations from available observations, PAM preserves reliable local neighborhood structures for stable prototype estimation, and PCC calibrates cross-view prototype correspondences through reciprocal cyclic transitions.

3.2 View-specific Reconstruction

We employ a view-specific autoencoder to learn latent representations from the available observations:

$$\mathbf{z}_i^{(v)} = E_{\phi^{(v)}}\left(\mathbf{x}_i^{(v)}\right), \tag{1}$$

where $E_{\phi^{(v)}}$ denotes the view-specific encoder parameterized by $\phi^{(v)}$. To preserve view-specific information and avoid collapse, we minimize reconstruction error over observed samples:

$$\mathcal{L}_{REC} = \frac{1}{V}\sum_{v=1}^{V}\frac{1}{N_v}\sum_{i=1}^{N} W_i^{(v)}\left\|\mathbf{x}_i^{(v)} - D_{\theta^{(v)}}\left(\mathbf{z}_i^{(v)}\right)\right\|_2^2, \tag{2}$$

where $D_{\theta^{(v)}}$ denotes the view-specific decoder parameterized by $\theta^{(v)}$, and $N_v = \sum_{i=1}^{N} W_i^{(v)}$ is the number of observed samples in the v-th view. Thus, only observed samples contribute to the reconstruction objective.

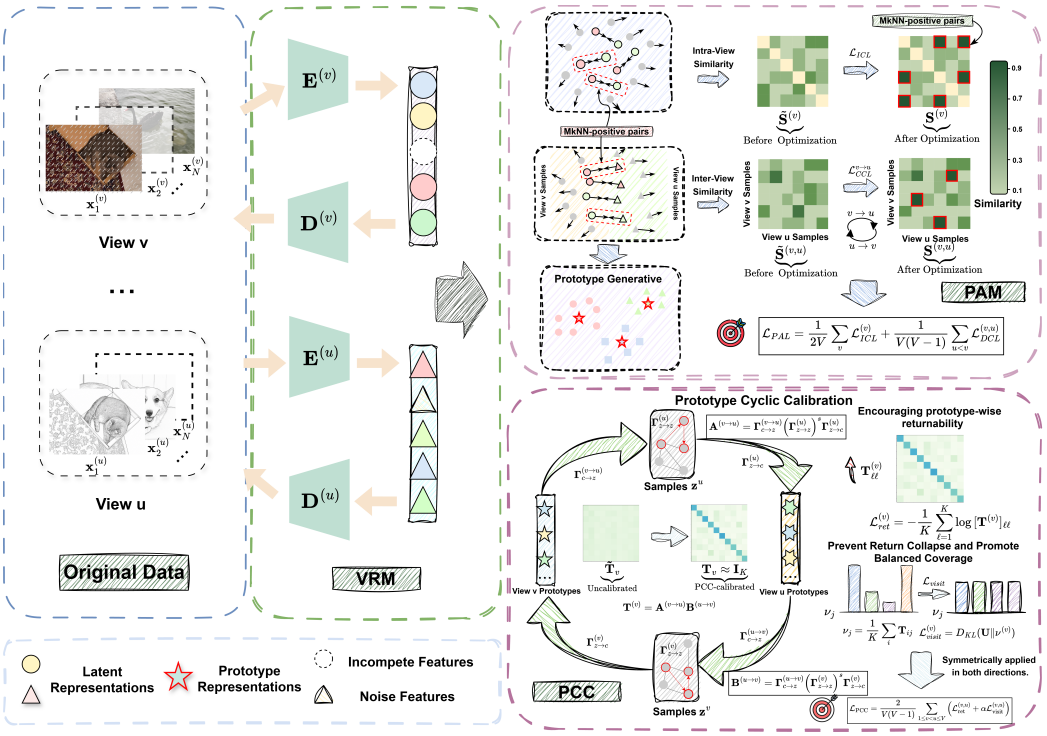


Figure 2: Overview of ReCAP, comprising the View-Specific Reconstruction Module (VRM), Prototype Aggregation Module (PAM), and Prototype Cyclic Calibration Module (PCC).

3.3 Prototype Aggregation via Graph Contrastive Learning

Many prototype-based multi-view clustering methods rely on complete or sufficiently many observed cross-view instance correspondences to learn consistent representations. Missing views reduce the number of observable instance pairs, limiting the supervision available for cross-view representation learning. Feature noise, meanwhile, distorts latent neighborhood structures and weakens the reliability of similarity-derived positive pairs. Inspired by (Wang et al., 2026b), the Prototype Aggregation Module constructs mutual k-nearest-neighbor (MkNN) graphs (Jarvis & Patrick, 1973; Gowda & Krishna, 1978) to select locally reciprocal relations as candidate semantic positives. Graph contrastive learning is then employed to preserve these neighborhood relations, thereby stabilizing early representations and prototype estimation for subsequent cross-view cyclic calibration.

Within each view, mutual k-nearest-neighbor pairs are selected as positive pairs, while the remaining distinct sample pairs are treated as negatives. The cosine similarity between two latent representations is defined as

$$S_{ij}^{(v)} = \mathrm{sim}\left(\mathbf{z}_i^{(v)}, \mathbf{z}_j^{(v)}\right) = \frac{\mathbf{z}_i^{(v)\top}\mathbf{z}_j^{(v)}}{\|\mathbf{z}_i^{(v)}\|_2 \|\mathbf{z}_j^{(v)}\|_2}. \tag{3}$$

Let $\mathcal{P}_i^{(v)}$ denote the positive set of anchor i in the v-th view. The intra-view graph contrastive objective is defined as

$$\mathcal{L}_{\mathrm{ICL}}^{(v)} = -\frac{1}{N_v}\sum_{i=1}^{N_v}\frac{1}{\left|\mathcal{P}_i^{(v)}\right|}\sum_{j\in\mathcal{P}_i^{(v)}}\log\frac{\exp\left(S_{ij}^{(v)}/\tau\right)}{\sum_{\ell\neq i}\exp\left(S_{i\ell}^{(v)}/\tau\right)}, \tag{4}$$

where τ denotes the temperature parameter. We further construct directional cross-view MkNN relations to model semantic associations between different views, where the similarity from view v to view u is defined as $S_{ij}^{(v\to u)} = \mathrm{sim}(\mathbf{z}_i^{(v)}, \mathbf{z}_j^{(u)})$. Accordingly, the directional cross-view contrastive

loss is formulated as

$$\mathcal{L}_{\mathrm{CCL}}^{(v\to u)} = -\frac{1}{N_v}\sum_{i=1}^{N_v}\frac{1}{\left|\mathcal{P}_i^{(v\to u)}\right|}\sum_{j\in\mathcal{P}_i^{(v\to u)}}\log\frac{\exp\left(S_{ij}^{(v\to u)}/\tau\right)}{\sum_{\ell=1}^{N_u}\exp\left(S_{i\ell}^{(v\to u)}/\tau\right)}. \tag{5}$$

Since cross-view MkNN relations are directional, we compute the contrastive loss in both directions to reduce directional bias. The resulting bidirectional cross-view contrastive loss is defined as

$$\mathcal{L}_{\mathrm{DCL}}^{(v,u)} = \frac{1}{2}\left(\mathcal{L}_{\mathrm{CCL}}^{(v\to u)} + \mathcal{L}_{\mathrm{CCL}}^{(u\to v)}\right). \tag{6}$$

Combining the intra-view and cross-view objectives with equal contributions, the Prototype Aggregation Loss (PAL) is formulated as

$$\mathcal{L}_{PAL} = \frac{1}{2V}\sum_{v=1}^{V}\mathcal{L}_{ICL}^{(v)} + \frac{1}{V(V-1)}\sum_{u<v}\mathcal{L}_{DCL}^{(v,u)}. \tag{7}$$

Finally, K-means is periodically applied to estimate view-specific candidate centroids, which are incorporated into the learnable prototypes through a momentum update:

$$\mathbf{C}_t^{(v)} = \eta\mathbf{C}_{t-1}^{(v)} + (1-\eta)\widehat{\mathbf{C}}_t^{(v)}, \tag{8}$$

where $\widehat{\mathbf{C}}_t^{(v)}$ denotes the newly estimated K-means centroids matched to the current prototypes using the Hungarian algorithm, and η is fixed at 0.9.

3.4 Cross-View Prototype Calibration via Reciprocal Cyclic Transitions

Existing methods align cross-view prototypes using pairwise similarities or structural relations. Under missingness and noise, prototype shifts and structural distortions can corrupt these one-step matches, which lack closed-loop verification. To address this issue, inspired by (Ayyad et al., 2021), PCC propagates probability mass from a source prototype through the sample and prototype spaces of a target view and back to the source view. It maximizes the return probability to the source prototype, promoting reciprocal cross-view prototype correspondences.

Specifically, given a source representation set $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_m]^\top \in \mathbb{R}^{m\times d}$ and a destination representation set $\mathbf{Y} = [\mathbf{y}_1, \ldots, \mathbf{y}_n]^\top \in \mathbb{R}^{n\times d}$, we define the transition matrix from $\mathbf{X}$ to $\mathbf{Y}$ as

$$[\mathbf{\Gamma}_{\mathbf{X}\to\mathbf{Y}}]_{ij} = \frac{\exp\left(\mathrm{sim}(\mathbf{x}_i, \mathbf{y}_j)/\tau\right)}{\sum_{\ell=1}^{n}\exp\left(\mathrm{sim}(\mathbf{x}_i, \mathbf{y}_\ell)/\tau\right)}, \qquad \mathbf{\Gamma}_{\mathbf{X}\to\mathbf{Y}} \in \mathbb{R}^{m\times n}, \tag{9}$$

where $\mathrm{sim}(\cdot,\cdot)$ denotes cosine similarity and $\tau = 0.2$ is shared by all transition matrices. The softmax is applied over the destination dimension, making $\mathbf{\Gamma}_{\mathbf{X}\to\mathbf{Y}}$ row-stochastic.

PCC constructs a cross-view cycle over prototype and sample spaces. For views v and u, probability mass is propagated from view-v prototypes to view-u prototypes through view-u samples, yielding the effective transition:

$$\mathbf{A}^{(v\to u)} = \mathbf{\Gamma}_{c\to z}^{(v\to u)}\left(\mathbf{\Gamma}_{z\to z}^{(u)}\right)^s\mathbf{\Gamma}_{z\to c}^{(u)}, \quad \mathbf{B}^{(u\to v)} = \mathbf{\Gamma}_{c\to z}^{(u\to v)}\left(\mathbf{\Gamma}_{z\to z}^{(v)}\right)^s\mathbf{\Gamma}_{z\to c}^{(v)}, \tag{10}$$

where s denotes the number of intra-view sample-transition steps, and $(\mathbf{\Gamma}_{z\to z})^s = \mathbf{I}$ when $s = 0$. Both $\mathbf{A}^{(v\to u)}$ and $\mathbf{B}^{(u\to v)}$ are $K \times K$ row-stochastic prototype transition matrices.

The bidirectional cyclic transition matrices are then given by

$$\mathbf{T}^{(v)} = \mathbf{A}^{(v\to u)}\mathbf{B}^{(u\to v)}, \qquad \mathbf{T}^{(u)} = \mathbf{B}^{(u\to v)}\mathbf{A}^{(v\to u)}. \tag{11}$$

To encourage each prototype to return to itself after the cross-view cycle, we minimize the bidirectional return loss:

$$\mathcal{L}_{\mathrm{ret}}^{(v,u)} = -\frac{1}{2K}\sum_{\ell=1}^{K}\left[\log\mathbf{T}_{\ell\ell}^{(v)} + \log\mathbf{T}_{\ell\ell}^{(u)}\right]. \tag{12}$$

Minimizing $\mathcal{L}_{\text{ret}}^{(v,u)}$ promotes prototype-level cyclic returnability (Proposition 1) and, ideally achieving reciprocal one-to-one transitions (Corollary 1).

Proposition 1 (Prototype-level Cyclic Returnability Guarantee). *Let $\mathbf{T}^{(v)}$ and $\mathbf{T}^{(u)}$ be the non-negative row-stochastic cyclic transition matrices defined in Eq. (11). Then,*

$$\frac{1}{4K}\left(\left\|\mathbf{T}^{(v)}-\mathbf{I}_K\right\|_{1,\text{entry}}+\left\|\mathbf{T}^{(u)}-\mathbf{I}_K\right\|_{1,\text{entry}}\right)\leq\mathcal{L}_{\text{ret}}^{(v,u)}. \tag{13}$$

Consequently, in the zero-loss limit,

$$\mathcal{L}_{\text{ret}}^{(v,u)}\to 0 \quad\Longrightarrow\quad \mathbf{T}^{(v)},\mathbf{T}^{(u)}\to\mathbf{I}_K. \tag{14}$$

Proof details are provided in Appendix B.

Corollary 1 (Reciprocity of Prototype Transitions). *Let $\mathbf{A}^{(v\to u)}$ and $\mathbf{B}^{(u\to v)}$ be the $K\times K$ non-negative row-stochastic directional effective transition matrices introduced above. Along any sequence of model states for which $\mathcal{L}_{\text{ret}}\to 0$, suppose that*

$$\mathbf{A}^{(v\to u)}\to\mathbf{A}_\star,\qquad \mathbf{B}^{(u\to v)}\to\mathbf{B}_\star. \tag{15}$$

Then there exists a permutation matrix $\mathbf{\Pi}^{(v\to u)}$ such that

$$\mathbf{A}_\star=\mathbf{\Pi}^{(v\to u)},\qquad \mathbf{B}_\star=\left(\mathbf{\Pi}^{(v\to u)}\right)^{-1}=\left(\mathbf{\Pi}^{(v\to u)}\right)^{\top}. \tag{16}$$

The proof is provided in Appendix C.

To discourage the cyclic transitions from collapsing onto a small subset of return prototypes, we regularize their marginal return distributions:

$$\mathcal{L}_{visit}^{(v,u)}=\frac{1}{2}(D_{KL}(\mathbf{U}\|\nu^{(\mathbf{v})})+D_{KL}(\mathbf{U}\|\nu^{(\mathbf{u})})), \tag{17}$$

where $\nu=\sum_i\mathbf{T}_i/\sum_{i,j}\mathbf{T}_{ij}$, $\mathbf{U}$ denotes the uniform distribution. The prototype cyclic calibration loss is defined as

$$\mathcal{L}_{\text{PCC}}=\frac{2}{V(V-1)}\sum_{1\leq v<u\leq V}\left(\mathcal{L}_{\text{ret}}^{(v,u)}+\alpha\mathcal{L}_{\text{visit}}^{(v,u)}\right), \tag{18}$$

where α is fixed at 0.05.

3.5 Overall Loss Function and Optimization

To further improve intra-cluster compactness and representation discriminability, following (Xie et al., 2016), we introduce an auxiliary self-training objective based on the view-specific prototypes. For an observed representation $\mathbf{z}_i^{(v)}$ and prototype $\mathbf{c}_k^{(v)}$, the corresponding soft assignment is computed using Student's t-distribution:

$$q_{ik}^{(v)}=\frac{(1+\|\mathbf{z}_i^{(v)}-\mathbf{c}_k^{(v)}\|_2^2)^{-1}}{\sum_{j=1}^{K}(1+\|\mathbf{z}_i^{(v)}-\mathbf{c}_j^{(v)}\|_2^2)^{-1}} \tag{19}$$

Subsequently, an auxiliary target distribution $p_{ik}^{(v)}$ is constructed to sharpen the soft assignment $q_{ik}^{(v)}$:

$$p_{ik}^{(v)}=\frac{(q_{ik}^{(v)})^2/\sum_j q_{jk}^{(v)}}{\sum_{k'}((q_{ik'}^{(v)})^2/\sum_j q_{jk'}^{(v)})} \tag{20}$$

Finally, we impose a clustering constraint by minimizing the KL divergence between $\mathbf{P}^{(v)}$ and $\mathbf{Q}^{(v)}$ for each view:

$$\mathcal{L}_{CLU}=\sum_v D_{KL}(\mathbf{P}^{(v)}\|\mathbf{Q}^{(v)}) \tag{21}$$

Overall, the total objective is formulated as:

$$\mathcal{L}_{ALL} = \mathcal{L}_{REC} + \omega_1(t)\mathcal{L}_{PAL} + \omega_2(t)\mathcal{L}_{PCC} + \omega_3(t)\mathcal{L}_{CLU} \tag{22}$$

where t denotes the training epoch, and $\omega_1(t)$, $\omega_2(t)$, and $\omega_3(t)$ are progressive loss weights defined as:

$$\omega_1(t) = \lambda_1(1 - 0.5r_t), \omega_2(t) = \lambda_2 r_t, \omega_3(t) = \lambda_3 r_t. \tag{23}$$

where λ_1, λ_2, and λ_3 are hyperparameters. The progressive coefficient r_t is defined below, where t_s and T_w denote the durations of structural warm-up and progressive weight ramp-up, respectively, and are set to 20 and 50:

$$r_t = \begin{cases} 0, & t < t_s, \\ \min\left(1, \dfrac{t - t_s + 1}{T_w}\right), & t \geq t_s, \end{cases} \tag{24}$$

4 Experiment

In this section, we conduct extensive experiments to evaluate the effectiveness of ReCAP, guided by the following research questions – **Q1 (performance & robustness):** Does ReCAP achieve superior and robust clustering performance under incomplete, noisy, and compound incomplete–noisy settings (Section 4.2)? **Q2 (cyclic prototype calibration effectiveness):** Does the cyclic calibration effectively promote semantically reliable and structurally reciprocal prototype correspondences across views (Section 4.3)? Additional experiments are provided in the Appendix.

4.1 Experimental Setup

Datasets and baselines. We conduct experiments on six widely used multi-view datasets: DHA (Lin et al., 2012), CUB (Wah et al., 2011), Hdigit (Chen et al., 2022), Caltech5V (Fei-Fei et al., 2004), Reuters (Amini et al., 2009), and NUS-WIDE (Chua et al., 2009). We compare ReCAP with nine representative methods: CPSPAN (Jin et al., 2023), ProImp (Li et al., 2023), DCP (Lin et al., 2022), SMILE (Zeng et al., 2023), CAMERA (Li et al., 2025), PMIMC (Yuan et al., 2025), CPMN (Wang et al., 2026b), GLGC (He et al., 2026), and DIMVC-HIA (Du et al., 2026). Baselines restricted to two-view data are omitted from multi-view datasets. We follow the parameter-selection protocols of the original papers. Dataset, baseline, and implementation details are provided in Appendix D to F.

Evaluation metrics. We evaluate clustering performance using clustering accuracy (ACC), adjusted Rand index (ARI), and normalized mutual information (NMI), and report the mean and standard deviation over five independent runs. To evaluate cross-view prototype alignment, we use the Semantic Reliability Score (SRS) to measure semantic consistency and prototype purity, and Structural Correspondence Reliability (SCR) to measure correspondence certainty and reciprocity. Their harmonic mean defines the Comprehensive Prototype Alignment Score (CPAS):

$$\mathrm{CPAS} = \frac{2\,\mathrm{SRS} \cdot \mathrm{SCR}}{\mathrm{SRS} + \mathrm{SCR}} \in [0, 1]. \tag{25}$$

Higher values indicate better alignment. These metrics are used only for post-hoc evaluation; their definitions are provided in Appendix G.

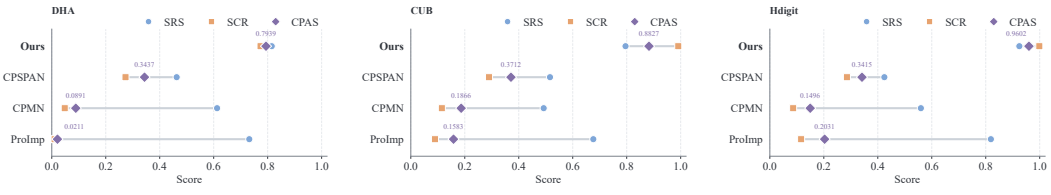


Figure 3: Prototype alignment under 50% view incompleteness. Circles, squares, and diamonds denote SRS, SCR, and CPAS, respectively; connecting lines show the SRS–SCR gap.

Table 1: Clustering performance (%) under partially incomplete multi-view settings. The best results are shown in bold, and the second-best results are underlined.

Method	DHA			CUB			Hdigit		
	ACC	ARI	NMI	ACC	ARI	NMI	ACC	ARI	NMI
CPSPAN (CVPR'23)	64.39 ± 2.67	52.34 ± 1.13	76.51 ± 0.35	68.67 ± 3.86	52.55 ± 3.64	67.99 ± 2.55	95.57 ± 0.64	80.47 ± 1.35	89.69 ± 1.17
ProImp (IJCAI'23)	66.46 ± 1.89	53.06 ± 1.48	75.29 ± 0.64	70.40 ± 4.52	53.82 ± 3.12	65.94 ± 1.73	97.78 ± 0.07	95.14 ± 0.15	93.79 ± 0.21
DCP (TPAMI'23)	34.95 ± 1.47	12.91 ± 2.23	47.40 ± 1.55	21.67 ± 3.31	3.17 ± 3.47	22.88 ± 7.29	96.15 ± 3.91	93.89 ± 3.91	93.99 ± 1.31
SMILE (TPAMI'24)	79.75 ± 1.59	69.23 ± 1.31	82.93 ± 0.70	68.48 ± 3.05	51.81 ± 2.77	63.25 ± 1.62	95.63 ± 0.19	90.58 ± 0.39	89.01 ± 0.34
CAMERA (TPAMI'25)	69.77 ± 2.98	58.70 ± 2.32	79.01 ± 0.91	73.97 ± 2.86	56.84 ± 2.74	68.48 ± 1.44	97.55 ± 0.15	94.63 ± 0.32	93.26 ± 0.39
PMIMC (TIP'25)	46.71 ± 5.04	32.30 ± 5.13	63.72 ± 4.08	64.70 ± 3.73	50.19 ± 4.30	65.57 ± 2.96	83.56 ± 4.62	70.43 ± 6.39	76.23 ± 3.78
CPMN (NeurIPS'25)	53.66 ± 2.68	34.17 ± 1.86	61.39 ± 1.36	38.40 ± 2.26	19.17 ± 2.42	36.08 ± 2.15	35.05 ± 4.51	15.06 ± 3.14	26.82 ± 3.56
GLGC (CVPR'26)	72.71 ± 1.97	58.05 ± 2.86	76.87 ± 1.51	35.50 ± 2.27	12.32 ± 1.54	31.57 ± 1.85	96.20 ± 0.18	91.74 ± 0.40	90.24 ± 0.42
DIMVC-HIA (AAAI'26)	50.35 ± 3.81	35.57 ± 3.97	61.27 ± 3.04	17.77 ± 1.73	1.54 ± 0.83	9.70 ± 2.99	15.73 ± 2.06	1.54 ± 1.23	4.55 ± 2.10
Ours	**80.08** ± 1.96	**69.24** ± 1.70	**84.22** ± 0.81	**81.17** ± 2.29	**65.06** ± 2.66	**72.62** ± 2.01	**98.37** ± 0.17	**96.41** ± 0.36	**95.36** ± 0.38
Method	Reuters			Caltech5V			NUS-WIDE		
	ACC	ARI	NMI	ACC	ARI	NMI	ACC	ARI	NMI
CPSPAN (CVPR'23)	39.27 ± 1.69	10.77 ± 1.55	16.18 ± 2.53	80.46 ± 3.26	65.45 ± 2.84	71.30 ± 1.91	13.54 ± 0.51	3.12 ± 0.14	17.69 ± 0.29
DCP (TPAMI'23)	30.02 ± 4.89	5.69 ± 3.69	15.50 ± 5.20	44.86 ± 2.64	19.51 ± 3.54	45.46 ± 2.63	12.93 ± 0.62	2.05 ± 0.31	14.92 ± 0.27
PMIMC (TIP'25)	44.48 ± 2.50	19.58 ± 1.32	25.32 ± 1.76	76.49 ± 3.22	58.35 ± 3.36	64.48 ± 2.10	12.78 ± 0.26	2.95 ± 0.18	17.61 ± 0.50
GLGC (CVPR'26)	31.72 ± 1.82	7.46 ± 1.20	12.93 ± 1.88	72.00 ± 7.52	52.98 ± 7.46	61.35 ± 5.60	16.11 ± 0.46	5.38 ± 0.34	20.01 ± 0.35
DIMVC-HIA (AAAI'26)	42.87 ± 3.88	15.70 ± 3.00	20.99 ± 3.46	71.10 ± 3.10	51.37 ± 2.91	57.56 ± 1.51	**16.99** ± 1.69	4.81 ± 0.89	16.59 ± 7.97
Ours	**53.42** ± 1.72	**24.78** ± 1.95	**29.82** ± 2.01	**88.89** ± 0.40	**78.66** ± 0.64	**80.72** ± 0.64	16.19 ± 0.38	**5.68** ± 0.14	**20.98** ± 0.40

Table 2: Clustering performance (%) under partially noisy multi-view settings. The best results are shown in bold, and the second-best results are underlined. NaN indicates a failed run with no meaningful result.

Method	DHA			CUB			Hdigit		
	ACC	ARI	NMI	ACC	ARI	NMI	ACC	ARI	NMI
CPSPAN (CVPR'23)	40.21 ± 3.34	23.69 ± 3.24	58.86 ± 3.04	30.40 ± 4.59	8.96 ± 7.78	29.38 ± 8.10	93.21 ± 0.86	85.64 ± 1.69	85.44 ± 1.08
ProImp (IJCAI'23)	42.32 ± 3.70	27.51 ± 3.62	58.10 ± 3.27	48.53 ± 4.37	32.48 ± 3.05	51.06 ± 2.48	74.23 ± 6.11	67.40 ± 7.12	80.27 ± 3.83
DCP (TPAMI'23)	11.43 ± 0.71	0.06 ± 0.15	13.25 ± 1.13	15.43 ± 1.15	0.32 ± 0.52	3.74 ± 1.24	88.14 ± 0.15	85.97 ± 0.38	91.66 ± 0.35
SMILE (TPAMI'24)	82.28 ± 1.39	73.27 ± 1.37	84.87 ± 0.48	71.02 ± 4.44	58.53 ± 2.67	68.74 ± 1.79	96.70 ± 0.14	92.83 ± 0.30	91.32 ± 0.27
CAMERA (TPAMI'25)	80.95 ± 1.99	72.05 ± 1.50	86.56 ± 0.36	80.87 ± 1.03	65.84 ± 1.64	75.21 ± 1.72	99.77 ± 0.02	99.50 ± 0.06	99.25 ± 0.07
PMIMC (TIP'25)	42.59 ± 2.43	27.16 ± 1.83	59.21 ± 1.53	49.70 ± 2.30	32.90 ± 1.68	55.32 ± 1.30	93.00 ± 1.28	86.13 ± 1.81	87.79 ± 3.09
CPMN (NeurIPS'25)	32.67 ± 2.00	14.95 ± 3.02	43.98 ± 3.11	65.10 ± 3.37	49.13 ± 2.82	63.60 ± 1.64	47.52 ± 2.04	30.96 ± 1.02	41.65 ± 1.45
GLGC (CVPR'26)	74.37 ± 2.90	61.54 ± 3.03	80.23 ± 1.36	53.17 ± 2.29	33.22 ± 2.13	47.42 ± 1.66	96.36 ± 0.28	92.11 ± 0.59	90.88 ± 0.65
DIMVC-HIA (AAAI'26)	72.26 ± 2.98	61.16 ± 3.57	78.73 ± 1.90	67.20 ± 5.51	49.12 ± 5.98	63.45 ± 3.28	NaN	NaN	NaN
Ours	**84.97** ± 1.24	**77.14** ± 1.28	**88.52** ± 0.71	**84.67** ± 1.89	**72.53** ± 1.90	**80.38** ± 1.12	**99.81** ± 0.02	**99.57** ± 0.05	**99.36** ± 0.07
Method	Reuters			Caltech5V			NUS-WIDE		
	ACC	ARI	NMI	ACC	ARI	NMI	ACC	ARI	NMI
CPSPAN (CVPR'23)	20.27 ± 0.86	0.13 ± 0.11	0.86 ± 0.20	64.11 ± 4.76	45.12 ± 5.22	51.98 ± 3.66	12.62 ± 0.88	2.04 ± 0.30	13.25 ± 0.53
DCP (TPAMI'23)	19.43 ± 0.62	0.04 ± 0.11	0.69 ± 0.15	31.51 ± 1.95	12.24 ± 3.00	18.57 ± 2.61	8.42 ± 0.61	0.01 ± 0.05	7.46 ± 0.13
PMIMC (TIP'25)	36.33 ± 1.60	10.99 ± 0.98	14.99 ± 0.98	70.86 ± 4.62	58.17 ± 3.12	50.92 ± 4.60	13.08 ± 0.53	3.34 ± 0.35	17.55 ± 0.58
GLGC (CVPR'26)	35.25 ± 5.30	9.62 ± 3.83	12.59 ± 5.37	73.74 ± 0.90	54.31 ± 0.87	62.13 ± 1.07	14.49 ± 0.39	4.19 ± 0.35	18.72 ± 0.41
DIMVC-HIA (AAAI'26)	17.02 ± 0.74	0.01 ± 0.03	0.10 ± 0.14	66.41 ± 2.31	45.63 ± 6.40	54.74 ± 4.24	15.44 ± 1.27	1.96 ± 0.77	8.62 ± 0.76
Ours	**52.87** ± 0.51	**24.45** ± 0.48	**30.31** ± 0.47	**91.31** ± 0.41	**83.34** ± 0.63	**85.44** ± 0.56	**16.73** ± 0.37	**6.05** ± 0.18	**22.35** ± 0.38

4.2 Performance Comparison

To answer **Q1**, we compare ReCAP with representative baselines under incomplete, noisy, and compound incomplete-and-noisy settings, with the corresponding degradation rates set to 0.5. As shown in Tables 1 to 3, under view incompleteness, ReCAP achieves the best ARI and NMI on all six datasets and the best ACC on five of them. In particular, it improves ACC over the strongest competitors by 7.20, 8.94, and 8.43 percentage points on CUB, Reuters, and Caltech5V, respectively. On NUS-WIDE, although its ACC is slightly lower than the best result, ReCAP achieves the highest ARI and NMI with a lower ACC variance. Under feature noise, ReCAP obtains the best results across all dataset–metric combinations; under compound degradation, it remains best in all but one case while maintaining generally low variances. Results under fully degraded settings and varying degradation rates are provided in Appendix H.

4.3 Model Discussion

Semantic and Structural Alignment. To answer **Q2**, we assess cross-view prototype correspondences using SRS, SCR, and CPAS (Equation (25)). As shown in Figure 3, ReCAP achieves the highest scores on Hdigit, DHA, and CUB. Although competing prototype-based methods maintain moderate SRS, their lower SCR reflects weaker correspondence certainty and reciprocity. Conse-

Table 3: Clustering performance (%) under partially noisy and incomplete multi-view settings. The best results are shown in bold, and the second-best results are underlined.

Method	DHA			CUB			Hdigit		
	ACC	ARI	NMI	ACC	ARI	NMI	ACC	ARI	NMI
CPSPAN (CVPR'23)	35.86 ± 1.66	20.04 ± 1.12	55.31 ± 0.82	31.30 ± 2.91	14.53 ± 1.04	33.44 ± 2.47	91.37 ± 0.89	81.96 ± 1.76	81.99 ± 1.34
ProImp (IJCAI'23)	39.00 ± 2.75	23.78 ± 1.85	51.61 ± 2.39	43.97 ± 2.56	27.06 ± 2.30	43.63 ± 2.84	57.38 ± 3.34	46.94 ± 3.37	62.52 ± 1.55
DCP (TPAMI'23)	11.97 ± 0.53	0.08 ± 0.13	13.34 ± 0.44	14.53 ± 1.06	0.23 ± 0.19	4.51 ± 0.34	78.79 ± 4.26	69.53 ± 6.33	79.53 ± 1.64
SMILE (TPAMI'24)	78.99 ± 0.54	68.22 ± 1.33	82.07 ± 0.97	65.30 ± 4.06	50.18 ± 2.23	62.77 ± 1.76	95.37 ± 0.16	90.04 ± 0.34	88.55 ± 0.37
CAMERA (TPAMI'25)	70.18 ± 3.36	57.04 ± 2.92	78.03 ± 1.35	73.97 ± 1.19	55.15 ± 1.77	**66.62** ± 1.21	96.97 ± 0.16	93.39 ± 0.34	91.88 ± 0.40
PMIMC (TIP'25)	39.71 ± 3.45	26.13 ± 3.64	58.43 ± 3.34	30.87 ± 2.19	15.38 ± 3.26	34.55 ± 2.97	51.26 ± 6.56	34.86 ± 8.44	50.34 ± 6.70
CPMN (NeurIPS'25)	52.17 ± 1.75	32.37 ± 1.34	60.01 ± 0.67	43.93 ± 2.16	23.63 ± 0.95	40.64 ± 2.02	37.76 ± 1.38	18.48 ± 1.36	30.29 ± 2.54
GLGC (CVPR'26)	61.90 ± 2.69	44.49 ± 3.65	67.98 ± 2.67	33.77 ± 2.05	10.31 ± 2.34	28.89 ± 1.77	88.29 ± 0.63	75.56 ± 1.32	75.08 ± 0.91
DIMVC-HIA (AAAI'26)	50.02 ± 6.74	35.38 ± 5.30	60.92 ± 3.53	16.27 ± 1.84	0.83 ± 0.79	5.75 ± 1.61	14.47 ± 1.19	0.79 ± 0.30	2.88 ± 0.69
Ours	**79.50** ± 1.66	**69.14** ± 1.96	**83.96** ± 1.13	**74.93** ± 1.73	**55.55** ± 1.44	66.07 ± 0.72	**98.09** ± 0.06	**95.81** ± 0.13	**94.64** ± 0.11

Method	Reuters			Caltech5V			NUS-WIDE		
	ACC	ARI	NMI	ACC	ARI	NMI	ACC	ARI	NMI
CPSPAN (CVPR'23)	19.90 ± 0.52	0.08 ± 0.14	0.77 ± 0.23	63.59 ± 5.40	43.04 ± 6.46	49.24 ± 4.55	12.41 ± 0.69	2.04 ± 0.40	12.71 ± 0.55
DCP (TPAMI'23)	18.83 ± 0.30	0.03 ± 0.04	0.84 ± 0.19	27.66 ± 2.78	6.75 ± 2.65	14.34 ± 3.06	9.42 ± 0.35	0.04 ± 0.12	6.54 ± 0.64
PMIMC (TIP'25)	20.25 ± 0.76	0.15 ± 0.16	0.86 ± 0.25	29.39 ± 1.40	8.07 ± 1.11	13.68 ± 2.06	8.82 ± 0.30	0.49 ± 0.10	9.27 ± 0.52
GLGC (CVPR'26)	29.77 ± 2.93	5.12 ± 1.26	8.70 ± 1.91	71.49 ± 7.37	52.47 ± 8.02	59.67 ± 5.99	15.26 ± 0.89	4.52 ± 0.43	17.88 ± 0.32
DIMVC-HIA (AAAI'26)	24.67 ± 1.89	1.89 ± 0.75	3.19 ± 0.83	67.23 ± 5.17	45.10 ± 3.72	49.24 ± 2.70	12.16 ± 0.37	1.93 ± 0.32	11.77 ± 0.45
Ours	**52.50** ± 1.06	**24.28** ± 0.65	**29.24** ± 0.50	**88.01** ± 0.67	**77.23** ± 1.34	**78.87** ± 1.27	**15.40** ± 0.35	**4.92** ± 0.25	**19.96** ± 0.38

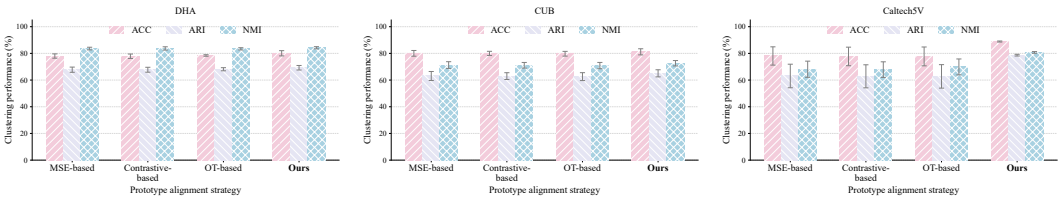


Figure 4: Comparison of prototype alignment replacement strategies in the PCC module.

quently, ReCAP attains higher CPAS by better preserving both semantic reliability and structural reciprocity.

Prototype Alignment Strategy Ablation. We replace PCC with MSE-, contrastive-, and OT-based objectives while keeping all other components fixed. As shown in Figure 4, PCC performs best across all datasets and metrics, exceeding the strongest alternative by over 10 percentage points on Caltech5V while substantially reducing variance. These results demonstrate that PCC establishes more reliable cross-view prototype correspondences, thereby improving clustering robustness.

Additional Experiments. We provide further experiments, including loss ablation, PCC objective ablation, and a warm-up strategy comparison (Appendix I); sensitivity analyses of the loss weights, MkNN neighborhood size, and intra-view transition steps, followed by a sample-mediated transition analysis (Appendix J); convergence and complexity and efficiency analyses (Appendix K); and clustering and cyclic-return visualizations (Appendix L).

5 Conclusion

In this work, we investigated cross-view prototype alignment for incomplete and noisy multi-view clustering and characterized correspondence ambiguity in terms of prototype reciprocity and coverage. To address this issue, we proposed ReCAP, a prototype calibration framework based on reciprocal cyclic transitions. Its cyclic-return objective promotes reciprocal cross-view correspondences, while visitation regularization prevents the marginal transition distribution from collapsing onto a few prototypes. Together, these objectives facilitate the discovery of a shared clustering structure without imputing missing views. Experiments on six widely used multi-view datasets demonstrate that ReCAP achieves competitive or superior clustering performance under incomplete, noisy, and compound degradation settings. These results establish reciprocal cyclic calibration as an effective alternative to conventional pairwise prototype matching.

AI USE STATEMENT

In this work, we used LLMs to assist with the translation, proofreading, and light language refinement of the manuscript. We also used LLMs to help identify potential bugs in the code and improve the clarity of mathematical derivations. All AI-assisted content was manually reviewed and verified by the authors, and we take full responsibility for the final content of this work.

REPRODUCIBILITY STATEMENT

We provide detailed descriptions of the datasets, experimental settings, implementation details, and hyperparameter configurations in the main text and Appendix. Additional ablation studies, sensitivity analyses, and theoretical derivations are also included in the Appendix. The anonymous repository contains the implementation of the main components of our method and supports end-to-end training and evaluation on a representative dataset. Upon acceptance, we will further complete and organize the codebase by releasing the full experimental pipeline, additional dataset configurations, and supporting scripts.

REFERENCES

Massih R Amini, Nicolas Usunier, and Cyril Goutte. Learning from multiple partially observed views-an application to multilingual text categorization. *Advances in neural information processing systems*, 22, 2009.

Ahmed Ayyad, Yuchen Li, Raden Muaz, Shadi Albarqouni, and Mohamed Elhoseiny. Semi-supervised few-shot learning with prototypical random walks. In *AAAI Workshop on Meta-Learning and MetaDL Challenge*, pp. 45–57. PMLR, 2021.

Man-Sheng Chen, Jia-Qi Lin, Xiang-Long Li, Bao-Yu Liu, Chang-Dong Wang, Dong Huang, and Jian-Huang Lai. Representation learning in multi-view clustering: A literature review. *Data Science and Engineering*, 7(3):225–241, 2022.

Wenlan Chen, Lu Gao, Daoyuan Wang, Fei Guo, and Cheng Liang. Geometry-aware variational information maximization for deep incomplete multi-view clustering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 20289–20297, 2026.

Anal Roy Chowdhury, Avisek Gupta, and Swagatam Das. Deep multi-view clustering: A comprehensive survey of the contemporary techniques. *Information Fusion*, 119:103012, 2025.

Tat-Seng Chua, Jinhui Tang, Richang Hong, Haojie Li, Zhiping Luo, and Yantao Zheng. Nus-wide: a real-world web image database from national university of singapore. In *Proceedings of the ACM international conference on image and video retrieval*, pp. 1–9, 2009.

Yuzhuo Dai, Jiaqi Jin, Zhibin Dong, Siwei Wang, Xinwang Liu, En Zhu, Xihong Yang, Xinbiao Gan, and Yu Feng. Imputation-free and alignment-free: Incomplete multi-view clustering driven by consensus semantic learning. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 5071–5081, 2025.

Yiming Du, Ziyu Wang, Jian Li, Rui Ning, and Lusi Li. Deep incomplete multi-view clustering via hierarchical imputation and alignment. *arXiv preprint arXiv:2601.09051*, 2026.

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 *2004 conference on computer vision and pattern recognition workshop*, pp. 178–178. IEEE, 2004.

K Chidananda Gowda and GJPR Krishna. Agglomerative clustering using the concept of mutual nearest neighbourhood. *Pattern recognition*, 10(2):105–112, 1978.

Hongqing He, Jie Xu, Wenyuan Yang, Yonghua Zhu, Guoqiu Wen, and Xiaofeng Zhu. Global-graph guided and local-graph weighted contrastive learning for unified clustering on incomplete and noise multi-view data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24813–24822, 2026.

Yu Hu, Endai Guo, Zhi Xie, Xinwang Liu, and Hongmin Cai. Robust multi-view clustering through partition integration on stiefel manifold. *IEEE Transactions on Knowledge and Data Engineering*, 35(10):10397–10410, 2023.

Raymond Austin Jarvis and Edward A Patrick. Clustering using a similarity measure based on shared near neighbors. *IEEE Transactions on computers*, 100(11):1025–1034, 1973.

Jiaqi Jin, Siwei Wang, Zhibin Dong, Xinwang Liu, and En Zhu. Deep incomplete multi-view clustering with cross-view partial sample and prototype alignment. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11600–11609, 2023.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Haobin Li, Yunfan Li, Mouxing Yang, Peng Hu, Dezhong Peng, and Xi Peng. Incomplete multi-view clustering via prototype-based imputation. *arXiv preprint arXiv:2301.11045*, 2023.

Haobin Li, Yijie Lin, Peng Hu, Mouxing Yang, and Xi Peng. Community-aware multi-view representation learning with incomplete information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

Yan-Ching Lin, Min-Chun Hu, Wen-Huang Cheng, Yung-Huan Hsieh, and Hong-Ming Chen. Human action recognition and retrieval using sole depth information. New York, NY, USA, 2012. Association for Computing Machinery. ISBN 9781450310895. doi: 10.1145/2393347.2396381. URL https://doi.org/10.1145/2393347.2396381.

Yijie Lin, Yuanbiao Gou, Xiaotian Liu, Jinfeng Bai, Jiancheng Lv, and Xi Peng. Dual contrastive prediction for incomplete multi-view representation learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(4):4447–4461, 2022.

Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016.

Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.

Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 2011.

Qianqian Wang, Xurui Liao, Wei Feng, and Quanxue Gao. Debiased and denoised representation learning for incomplete multi-view clustering. In *International Conference on Learning Representations*, volume 2026, pp. 73338–73350, 2026a.

Yiming Wang, Qun Li, Dongxia Chang, Jie Wen, Hua Dai, Fu Xiao, and Yao Zhao. Learning from disjoint views: A contrastive prototype matching network for fully incomplete multi-view clustering. *Advances in Neural Information Processing Systems*, 38:18265–18284, 2026b.

Youqing Wang, Yu Cao, Jinlu Wang, Xiang Xu, Jiapu Wang, Tengfei Liu, Junbin Gao, and Jipeng Guo. Aligning collaborative view recovery and tensorial subspace learning via latent representation for incomplete multi-view clustering. In *The Fourteenth International Conference on Learning Representations*, 2026c.

Junyuan Xie, Ross Girshick, and Ali Farhadi. Unsupervised deep embedding for clustering analysis. In *International conference on machine learning*, pp. 478–487. PMLR, 2016.

Xihong Yang, Siwei Wang, Fangdi Wang, Jiaqi Jin, Suyuan Liu, Yue Liu, En Zhu, Xinwang Liu, and Yueming Jin. Automatically identify and rectify: Robust deep contrastive multi-view clustering in noisy scenarios. *arXiv preprint arXiv:2505.21387*, 2025.

Shengju Yu, Zhibin Dong, Siwei Wang, Pei Zhang, Yi Zhang, Xinwang Liu, Naiyang Guan, Tiejun Li, and Yiu-ming Cheung. Simple yet effective incomplete multi-view clustering: Similarity-level imputation and intra-view hybrid-group prototype construction. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=KijslFbfOL.

Honglin Yuan, Yuan Sun, Fei Zhou, Jing Wen, Shihua Yuan, Xiaojian You, and Zhenwen Ren. Prototype matching learning for incomplete multi-view clustering. *IEEE Transactions on Image Processing*, 34:828–841, 2025.

Pengxin Zeng, Mouxing Yang, Yiding Lu, Changqing Zhang, Peng Hu, and Xi Peng. Semantic invariant multi-view clustering with fully incomplete information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(4):2139–2150, 2023.

Yuanyang Zhang, Yijie Lin, Weiqing Yan, Li Yao, Xinhang Wan, Guangyuan Li, Chao Zhang, Guanzhou Ke, and Jie Xu. Incomplete multi-view clustering via diffusion contrastive generation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 39, pp. 22650–22658, 2025.

Appendix

Table of Contents

A Notation and Terminology **13**

B Proof of Prototype-Level Cyclic Returnability Guarantee **14**

C Proof of Reciprocity in the Zero-Loss Limit **15**

D Detailed Information About The Datasets **15**

E Detailed information on the compared methods **16**

F Detailed information on the experimental settings **17**

G Details of the Comprehensive Prototype Alignment Score **18**

H Comparison Results **21**

I Ablation Studies **22**

J Hyperparameter Analysis **24**

K Convergence and Computational Analysis **25**

L Visualization Analysis **27**

A Notation and Terminology

Throughout this paper, we use the following notation:

N	Number of samples.
V	Number of views.
K	Number of prototypes or clusters.
C	Number of ground-truth classes.
Ω_v	Set of observed samples in view v.
$\mathbf{x}_i^{(v)}$	The i-th raw input sample in the v-th view
$\mathbf{z}_i^{(v)}$	Latent representation of sample i in view v.
$\mathbf{c}_k^{(v)}$	Prototype k in view v.
$q_{ik}^{(v)}$	Soft assignment of sample i to prototype k.
$\mathbf{h}_k^{(v)}$	Class-semantic distribution of prototype k.
$s_{kl}^{(v,u)}$	Semantic similarity between cross-view prototypes.
$\mathbf{A}^{v\to u}, \mathbf{B}^{u\to v}$	Prototype transition matrix from view v to view u, or vice versa.

B Proof of Prototype-Level Cyclic Returnability Guarantee

Proposition 1 establishes an upper bound on the deviation of the cyclic transition matrices from the identity matrix in terms of the return loss. We provide its proof below.

Proof. Consider either cyclic transition matrix $\mathbf{T} \in \mathbb{R}^{K\times K}$. Since $\mathbf{T}$ is non-negative and row-stochastic, it satisfies

$$T_{ij} \geq 0, \qquad \sum_{j=1}^{K} T_{ij} = 1. \tag{26}$$

For each row i, we have

$$\sum_{j=1}^{K} |T_{ij} - \delta_{ij}| = |T_{ii} - 1| + \sum_{j\neq i} |T_{ij}| \tag{27}$$

$$= (1 - T_{ii}) + \sum_{j\neq i} T_{ij} \tag{28}$$

$$= 2(1 - T_{ii}), \tag{29}$$

where δ_{ij} denotes the Kronecker delta. Summing over all rows yields

$$\|\mathbf{T} - \mathbf{I}_K\|_{1,\text{entry}} = 2\sum_{i=1}^{K}(1 - T_{ii}) = 2\left(K - \text{Tr}(\mathbf{T})\right). \tag{30}$$

For $0 < T_{ii} \leq 1$, the inequality $-\log T_{ii} \geq 1 - T_{ii}$ gives

$$-\frac{1}{K}\sum_{i=1}^{K} \log T_{ii} \geq \frac{1}{K}\sum_{i=1}^{K}(1 - T_{ii}) \tag{31}$$

$$= \frac{1}{2K}\|\mathbf{T} - \mathbf{I}_K\|_{1,\text{entry}}. \tag{32}$$

Applying this inequality separately to $\mathbf{T}^{(v)}$ and $\mathbf{T}^{(u)}$ gives

$$\frac{1}{2K}\left\|\mathbf{T}^{(v)} - \mathbf{I}_K\right\|_{1,\text{entry}} \leq \mathcal{L}_{\text{ret}}^{(v)}, \tag{33}$$

$$\frac{1}{2K}\left\|\mathbf{T}^{(u)} - \mathbf{I}_K\right\|_{1,\text{entry}} \leq \mathcal{L}_{\text{ret}}^{(u)}. \tag{34}$$

Adding the two inequalities and using

$$\mathcal{L}_{\text{ret}} = \frac{1}{2}\left(\mathcal{L}_{\text{ret}}^{(v)} + \mathcal{L}_{\text{ret}}^{(u)}\right) \tag{35}$$

yields

$$\frac{1}{4K}\left(\left\|\mathbf{T}^{(v)} - \mathbf{I}_K\right\|_{1,\text{entry}} + \left\|\mathbf{T}^{(u)} - \mathbf{I}_K\right\|_{1,\text{entry}}\right) \leq \mathcal{L}_{\text{ret}}. \tag{36}$$

Therefore, if $\mathcal{L}_{\text{ret}} \to 0$, both non-negative deviation terms must converge to zero. Hence,

$$\mathbf{T}^{(v)} \to \mathbf{I}_K, \qquad \mathbf{T}^{(u)} \to \mathbf{I}_K. \tag{37}$$

□

C Proof of Reciprocity in the Zero-Loss Limit

Proof. For notational simplicity, let $\mathbf{A}$ and $\mathbf{B}$ denote $\mathbf{A}^{(v\to u)}$ and $\mathbf{B}^{(u\to v)}$, respectively. By construction, both matrices are non-negative and row-stochastic.

Consider a sequence of model states satisfying $\mathcal{L}_{\text{ret}} \to 0$, along which

$$\mathbf{A} \to \mathbf{A}_\star, \qquad \mathbf{B} \to \mathbf{B}_\star. \tag{38}$$

Since the set of non-negative row-stochastic matrices is closed, $\mathbf{A}_\star$ and $\mathbf{B}_\star$ remain non-negative and row-stochastic.

By Proposition 1,

$$\mathbf{AB} \to \mathbf{I}_K, \qquad \mathbf{BA} \to \mathbf{I}_K. \tag{39}$$

By continuity of matrix multiplication, it follows that

$$\mathbf{A}_\star\mathbf{B}_\star = \mathbf{I}_K, \qquad \mathbf{B}_\star\mathbf{A}_\star = \mathbf{I}_K. \tag{40}$$

Let $\mathbf{b}_{\star,k}^\top$ denote the k-th row of $\mathbf{B}_\star$. The i-th row of $\mathbf{A}_\star\mathbf{B}_\star = \mathbf{I}_K$ satisfies

$$\mathbf{e}_i^\top = \sum_{k=1}^{K} [\mathbf{A}_\star]_{ik}\mathbf{b}_{\star,k}^\top. \tag{41}$$

Because $\mathbf{A}_\star$ is row-stochastic, the right-hand side of equation 41 is a convex combination of the rows of $\mathbf{B}_\star$, each of which is a probability vector. Since $\mathbf{e}_i$ is an extreme point of the probability simplex,

$$[\mathbf{A}_\star]_{ik} > 0 \quad \Longrightarrow \quad \mathbf{b}_{\star,k}^\top = \mathbf{e}_i^\top. \tag{42}$$

Equation 40 implies that $\mathbf{B}_\star$ is invertible and therefore cannot contain duplicate rows. Hence, for each i, there is exactly one index $\pi(i)$ for which $[\mathbf{A}_\star]_{i,\pi(i)} > 0$. Since each row of $\mathbf{A}_\star$ sums to one,

$$[\mathbf{A}_\star]_{i,\pi(i)} = 1, \qquad \mathbf{b}_{\star,\pi(i)}^\top = \mathbf{e}_i^\top. \tag{43}$$

Distinct source indices must correspond to distinct target indices; otherwise, the same row of $\mathbf{B}_\star$ would equal two different standard basis vectors. Thus, π is a bijection over $\{1, \dots, K\}$, and there exists a permutation matrix $\mathbf{\Pi}^{(v\to u)}$ such that

$$\mathbf{A}_\star = \mathbf{\Pi}^{(v\to u)}. \tag{44}$$

Finally, by equation 40,

$$\mathbf{B}_\star = \left(\mathbf{\Pi}^{(v\to u)}\right)^{-1} = \left(\mathbf{\Pi}^{(v\to u)}\right)^{\top}. \tag{45}$$

□

D Detailed Information About The Datasets

We employ six widely used open-source datasets, including DHA, CUB, Hdigit, Caltech5V, Reuters, and NUS-WIDE. The detailed information is presented as follows:

- **CUB**. CUB is a typical image–text multimodal dataset. We use a commonly adopted multi-view subset containing 600 samples from 10 categories with two views. The first view consists of 1,024-dimensional visual features that characterize the appearance, shape, and texture of bird images, while the second view contains 300-dimensional textual semantic features derived from the corresponding descriptions. The two views represent the same objects from visual and linguistic modalities, providing complementary cross-modal information.
- **DHA**. DHA (Depth-included Human Action) is a multimodal dataset for human action recognition. The version used in our experiments contains 483 samples from 23 action categories with two views. The two views are represented by 110-dimensional depth features and 6,144-dimensional RGB visual features, respectively. The depth view mainly captures spatial structure, body contours, and pose variations, whereas the RGB view describes appearance, texture, and motion information, providing complementary descriptions of human actions across modalities.

- **Hdigit**. Hdigit is a two-view handwritten digit dataset constructed from MNIST and USPS, containing 10,000 samples from 10 digit classes. The first view consists of 784-dimensional MNIST features corresponding to (28×28) grayscale handwritten digit images, while the second view contains 256-dimensional USPS features corresponding to (16×16) digit images. Although the two views share the same class semantics, they exhibit noticeable differences in image resolution, handwriting style, and data distribution, making Hdigit suitable for evaluating cross-domain multi-view representation learning.
- **Caltech5V**. Caltech5V is a multi-view image dataset derived from Caltech image data. The version used in our experiments contains 1,400 samples from 7 categories with five views. These views are represented by different visual descriptors, including 40-dimensional Wavelet Moments (WM), 254-dimensional CENTRIST, 1,984-dimensional Histogram of Oriented Gradients (HOG), 512-dimensional GIST, and 928-dimensional Local Binary Pattern (LBP) features. They characterize the same image from complementary perspectives, including texture, local structure, edges and shapes, global scene structure, and local texture patterns.
- **NUS-WIDE**. NUS-WIDE is a large-scale Web image dataset. We use a multi-view subset containing 2,000 samples from 31 categories with five views. The five views consist of 65-dimensional Color Histogram (CH), 226-dimensional Color Moments (CM), 145-dimensional Color Correlogram (CORR), 74-dimensional Edge Distribution Histogram (EDH), and 129-dimensional Wavelet Texture (WT) features. These descriptors characterize complementary visual properties, including color distribution, color statistics, spatial color correlations, edge structures, and texture information.
- **Reuters**. Reuters is a representative multilingual multi-view text dataset. The Reuters version used in our experiments contains 1,200 samples from 6 categories with five views, where each view is represented by a 2,000-dimensional text feature vector. The five views correspond to English, French, German, Italian, and Spanish documents. Although the language-specific views differ substantially in vocabulary space and linguistic expression, they convey the same underlying news semantics, making the dataset suitable for evaluating cross-view semantic consistency across heterogeneous language spaces.

E Detailed information on the compared methods

We compare our method with several state-of-the-art methods. The detailed information is presented as follows:

- **CPSPAN**. CPSPAN (Jin et al., 2023) is a deep incomplete multi-view clustering method that jointly performs sample- and prototype-level alignment. It uses pair-observed samples as proxy supervision to establish instance correspondences and employs prototype alignment to mitigate distribution shifts caused by missing observations, thereby learning consistent clustering structures from incomplete data. **The method supports multi-view datasets with more than two views.**
- **ProImp**. ProImp (Li et al., 2023) is a prototype-based imputation method for incomplete multi-view clustering. It employs a dual-stream network with attention and contrastive learning to capture view-specific prototypes and sample-to-prototype relationships. Missing information is reconstructed using prototypes from the unavailable view and relationships transferred from observed views, preserving instance commonality and view versatility. **The method is restricted to datasets with exactly two views.**
- **DCP**. DCP (Lin et al., 2022) provides an information-theoretic framework for cross-view consistency learning and missing-view recovery. It maximizes cross-view mutual information through contrastive learning while recovering missing information via dual prediction with conditional-entropy minimization, integrating representation learning and data recovery into a unified framework. **The method supports multi-view datasets with more than two views.**
- **SMILE**. SMILE (Zeng et al., 2023) is designed for fully incomplete multi-view clustering without requiring paired samples. It exploits invariant semantic distributions shared across views to reduce distribution discrepancies and learn consensus semantics, which are subsequently used to realign defective instances and obtain a unified clustering structure under

missing-instance and missing-correspondence scenarios. **Following the requirements of its original implementation, this baseline is evaluated only on two-view datasets, and remains applicable under fully incomplete settings with a missing rate of** 1.0**.**

- **CAMERA**. CAMERA (Li et al., 2025) addresses incomplete multi-view representation learning through community structures. It introduces community commonality to improve view-specific cluster compactness and community versatility to preserve view diversity. These properties are incorporated into a dual-stream network to balance sample restoration, cross-view alignment, and view-specific information preservation. **The method is restricted to datasets with exactly two views, and remains applicable under fully incomplete settings with a missing rate of** 1.0**.**
- **PMIMC**. PMIMC (Yuan et al., 2025) addresses prototype misalignment and imputation instability in incomplete multi-view clustering. It employs relational consistency learning to reduce cross-view heterogeneity and robust prototype contrastive learning to alleviate prototype misalignment. Prototype-based imputation further improves the reliability of missing-view recovery under high missing rates. **The method supports multi-view datasets with more than two views.**
- **CPMN**. CPMN (Wang et al., 2026b) targets fully incomplete multi-view clustering without cross-view correspondences. It employs correspondence-free graph contrastive learning based on mutual k-nearest neighbors to capture intrinsic structures and initialize view-specific prototypes. Cross-view prototype graph matching then resolves category-level semantic misalignment, followed by prototype-aware contrastive learning to enhance semantic consistency. **The method is restricted to datasets with exactly two views, and remains applicable under fully incomplete settings with a missing rate of** 1.0**.**
- **GLGC**. GLGC (He et al., 2026) provides a unified contrastive learning framework for incomplete and noisy multi-view clustering without explicit imputation. It constructs a global affinity graph to discover additional semantically related pairs and employs local-graph weighted contrastive learning to strengthen reliable cross-view pairs while suppressing noise-contaminated ones, improving robustness under compound degradation. **The method supports multi-view datasets with more than two views, and remains applicable under fully incomplete settings with a missing rate of** 1.0**.**
- **DIMVC-HIA**. DIMVC-HIA (Du et al., 2026) addresses incomplete multi-view clustering through hierarchical imputation and semantic alignment. It estimates missing cluster assignments using cross-view contrastive similarities and reconstructs missing latent features from intra-view, intra-cluster statistics. Energy-based semantic alignment and contrastive assignment alignment further enhance cluster compactness and cross-view consistency. **The method supports multi-view datasets with more than two views.**

For all methods, hyperparameters follow the tuning strategies specified in their original papers. We use identical view-missing and feature-noise generation mechanisms across methods. Each method is independently run five times with different random seeds, and the mean and standard deviation are reported.

F Detailed information on the experimental settings

This section provides the details of the experimental setup, as follows:

- **Network Architecture**. ReCAP employs an MLP-based autoencoder architecture, with the feature layer dimensions set to: $d_v \to 1024 \to 1024 \to 1024 \to d$, $d = 128$. Meanwhile, the model maintains a learnable prototype matrix of size $K \times d$ for each of the V views, which is used to store and update the view-specific cluster prototypes.
- **Training Strategy.** We first pretrain the MLP for warm-up and then proceed to the formal training stage. The learning rate is updated using a cosine annealing schedule implemented by `CosineAnnealingLR` (Loshchilov & Hutter, 2016) and the initial learning rate is set to 1×10^{-4}. Select Adam as the optimizer (Kingma & Ba, 2014). For DHA, we instead use a fixed learning rate of 1×10^{-4} throughout training. Before prototype alignment, the model first undergoes a feature aggregation warm-up stage to obtain stable latent representations.

Table 4: Implementation settings. P/T denotes pretraining/training epochs.

Dataset	Batch Size	Epochs (P/T)	s	$\boldsymbol{\lambda}$
DHA	483	250/200	0	$(1, 0.1, 10)$
CUB	600	150/300	0	$(1, 1, 1)$
Hdigit	1024	100/300	2	$(1, 1, 1)$
Caltech5V	1024	100/300	2	$(10, 10, 0.1)$
NUS-WIDE	1024	100/300	2	$(100, 0.1, 10)$
Reuters	512	100/100	2	$(1, 1, 1)$

We first warm up representation learning for 20 epochs, followed by a 50-epoch curriculum phase during which the loss weights are progressively adjusted.

- **Data degradation protocol**. We evaluate ReCAP under incomplete (I), noisy (N), and compound incomplete–noisy (IN) settings. Before degradation, the features of each view are normalized independently. Given a degradation rate $r \in [0, 1]$, we randomly select $\lfloor rN \rfloor$ instances without replacement for each active degradation.

  For the incomplete setting, each selected instance is assigned a nontrivial binary observation mask, such that at least one view is observed and at least one view is missing. Formally, $1 \leq \sum_{v=1}^{V} W_i^{(v)} \leq V - 1$ for every selected incomplete instance. Missing observations are zero-filled, while the observation mask is provided to the model to exclude unavailable views from training.

  For the noisy setting, one view is randomly selected for each noisy instance and corrupted by additive Gaussian noise:

$$\widetilde{\mathbf{x}}_i^{(v)} = \mathbf{x}_i^{(v)} + \boldsymbol{\epsilon}_i^{(v)}, \qquad \boldsymbol{\epsilon}_i^{(v)} \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I}),$$

  where $\sigma = 0.4$. The remaining views are unchanged, and the noise locations are not provided to the model.

  For the compound IN setting, the incomplete and noisy subsets are sampled independently, each at rate r, and their overlap is allowed. Missingness is applied first, after which Gaussian noise is injected only into observed views. Thus, r denotes the sample-level rate of each active degradation rather than the proportion of samples simultaneously affected by both degradations.

- **Experiment Setting**. We implement ReCAP using PyTorch and conduct all experiments on an NVIDIA GeForce RTX 3080Ti GPU. Each view is processed by an independent autoencoder, whose output dimension is fixed at 128. The number of clusters is set to the ground-truth number of categories following the standard multi-view clustering protocol.The dataset-specific training configurations are summarized in Table 4.

  For prototype aggregation, the number of mutual nearest neighbors is set to $k = 6$ for all datasets. In PCC, we use direct sample-mediated transitions ($s = 0$) on DHA and CUB, and shallow two-step propagation ($s = 2$) on Hdigit, Caltech5V, Reuters, and NUS-WIDE. The sensitivity to the propagation depth is reported in Appendix J.

  We first pretrain the view-specific autoencoders and then jointly optimize all objectives for the number of epochs listed in Table 4. During joint optimization, the first 20 epochs are used to warm up representation learning, followed by a 50-epoch curriculum phase during which the loss weights are progressively adjusted. The visitation regularization coefficient and prototype momentum are fixed to $\alpha = 0.05$ and $\eta = 0.9$, respectively.

G Details of the Comprehensive Prototype Alignment Score

This section details the Comprehensive Prototype Alignment Score (CPAS) defined in Equation (25), including its Semantic Reliability Score (SRS) and Structural Correspondence Reliability (SCR).

Semantic Reliability Score. SRS combines Prototype Semantic Correspondence (PSC) and Hard Prototype Purity (H-Purity), which measure cross-view semantic consistency and prototype-level semantic purity, respectively. For PSC, we first define the semantic distribution of each prototype as

follows:

$$q_{ik}^{(v)} = \frac{\left(1 + \|\mathbf{z}_i^{(v)} - \mathbf{c}_k^{(v)}\|_2^2\right)^{-1}}{\sum_{r=1}^{K}\left(1 + \|\mathbf{z}_i^{(v)} - \mathbf{c}_r^{(v)}\|_2^2\right)^{-1}}, \tag{46}$$

where $\mathbf{z}_i^{(v)}$ and $\mathbf{c}_k^{(v)}$ denote the representation of sample i and prototype k in view v, respectively, and $q_{ik}^{(v)}$ is their soft assignment computed using Student's t-distribution (Xie et al., 2016). Given the ground-truth labels y_i, the semantic distribution of prototype k over class c is defined as:

$$h_{kc}^{(v)} = \frac{\sum_{i\in\Omega_v} q_{ik}^{(v)} \mathbb{I}(y_i = c)}{\sum_{i\in\Omega_v} q_{ik}^{(v)}}, \tag{47}$$

where Ω_v denotes the set of observed samples in view v. The resulting semantic distribution is

$$\mathbf{h}_k^{(v)} = [h_{k1}^{(v)}, \ldots, h_{kC}^{(v)}] \tag{48}$$

characterizes the class-semantic distribution captured by prototype k. Since ground-truth labels are used, SRS and CPAS serve only as post-training evaluation metrics and are not involved in unsupervised training. We further compute the semantic similarity between cross-view prototypes. For prototype k in view v and prototype l in view u, we compute the Jensen–Shannon divergence between their semantic distributions as

$$\mathbf{m}_{kl}^{(v,u)} = \frac{1}{2}\left(\mathbf{h}_k^{(v)} + \mathbf{h}_l^{(u)}\right), \tag{49}$$

$$\mathrm{JSD}\left(\mathbf{h}_k^{(v)}, \mathbf{h}_l^{(u)}\right) = \frac{1}{2} D_{\mathrm{KL}}\left(\mathbf{h}_k^{(v)} \middle\| \mathbf{m}_{kl}^{(v,u)}\right) + \frac{1}{2} D_{\mathrm{KL}}\left(\mathbf{h}_l^{(u)} \middle\| \mathbf{m}_{kl}^{(v,u)}\right). \tag{50}$$

The divergence is normalized and converted into a semantic similarity:

$$s_{kl}^{(v,u)} = 1 - \frac{\mathrm{JSD}\left(\mathbf{h}_k^{(v)}, \mathbf{h}_l^{(u)}\right)}{\log 2}. \tag{51}$$

When the two prototypes have identical class-semantic distributions, $s_{kl}^{(v,u)} = 1$. Finally, based on Eq. 10, we obtain the prototype transition matrix $\mathbf{A}^{v\to u}$, where $A_{kl}^{v\to u}$ denotes the probability of transitioning from prototype k in view v to prototype l in view u. The directional PSC is defined as

$$\mathrm{PSC}^{v\to u} = \frac{1}{K}\sum_{k=1}^{K}\sum_{l=1}^{K} A_{kl}^{v\to u} s_{kl}^{(v,u)}. \tag{52}$$

The bidirectional PSC is then given by

$$\mathrm{PSC}^{(v,u)} = \frac{1}{2}\left(\mathrm{PSC}^{v\to u} + \mathrm{PSC}^{u\to v}\right). \tag{53}$$

A higher PSC indicates that the prototype transition probabilities are more concentrated on semantically similar target prototypes. However, PSC has an inherent limitation: when different prototypes exhibit similarly ambiguous class distributions, their JSD can remain small. We therefore complement PSC with a prototype purity measure.

Each sample is assigned to its most probable prototype:

$$\hat{k}_i^{(v)} = \arg\max_k q_{ik}^{(v)}. \tag{54}$$

The resulting class distribution of prototype k is

$$\bar{h}_{kc}^{(v)} = \frac{\sum_{i\in\Omega_v} \mathbb{I}(\hat{k}_i^{(v)} = k)\mathbb{I}(y_i = c)}{\sum_{i\in\Omega_v} \mathbb{I}(\hat{k}_i^{(v)} = k)}. \tag{55}$$

Its entropy-based purity is defined as

$$\text{H-Purity}_k^{(v)} = 1 - \frac{-\sum_{c=1}^{C} \bar{h}_{kc}^{(v)} \log \bar{h}_{kc}^{(v)}}{\log C}. \tag{56}$$

We average it across prototypes and views:

$$\text{H-Purity}^{(v)} = \frac{1}{K}\sum_{k=1}^{K}\text{H-Purity}_k^{(v)}, \qquad \text{H-Purity}^{(v,u)} = \frac{\text{H-Purity}^{(v)} + \text{H-Purity}^{(u)}}{2}. \tag{57}$$

Finally, the Semantic Reliability Score (SRS) is defined as the harmonic mean of PSC and H-Purity:

$$\boxed{\text{SRS}^{(v,u)} = \frac{2\,\text{PSC}^{(v,u)}\,\text{H-Purity}^{(v,u)}}{\text{PSC}^{(v,u)} + \text{H-Purity}^{(v,u)}}}. \tag{58}$$

Structural Correspondence Reliability. SCR does not rely on ground-truth labels. It evaluates the certainty of prototype mappings through MTM and the reciprocity of cross-view prototype correspondences through RMR. To quantitatively evaluate the certainty of cross-view prototype matching, we introduce the Matching Trustworthiness Measure (MTM), which is defined as follows:

$$MTM^{(v,u)} = 1 - \frac{1}{2}\left[\mathcal{H}_\tau(\mathbf{S}^{v,u}) + \mathcal{H}_\tau(\mathbf{S}^{u,v})\right], \qquad \mathcal{MTM}^{v,u} \in [0,1], \tag{59}$$

where $\mathbf{S}_{ij}^{v,u} = \frac{(\mathbf{c}_i^{(v)})^T\mathbf{c}_j^{(u)}}{\|\mathbf{c}_i^{(v)}\|_2\|\mathbf{c}_j^{(u)}\|_2}$, $\mathbf{S}_{ij}^{v,u} = \mathbf{S}_{ji}^{u,v}$. $\mathcal{H}_\tau(\cdot)$ denotes the normalized entropy of the row-wise softmax correspondence distribution, which is defined as follows:

$$\mathcal{H}_\tau(\mathbf{S}) = -\frac{1}{K\log K}\sum_{i,j} P_{ij}\log P_{ij}, \qquad \mathbf{P} = \text{Softmax}_{\text{row}}(\mathbf{S}/\tau), \tag{60}$$

where τ denotes the temperature coefficient and is fixed at 0.2. $\mathcal{H}_\tau(\mathbf{S}^{v,u})$ and $\mathcal{H}_\tau(\mathbf{S}^{u,v})$ denote the normalized entropies of the cross-view prototype correspondences in the two directions, respectively. A larger $MTM^{(v,u)}$ indicates more certain cross-view prototype matching, whereas a smaller $MTM^{(v,u)}$ suggests more ambiguous and random correspondences.

However, MTM may remain high when multiple prototypes are confidently mapped to the same target prototype. We therefore introduce the Reciprocal Matching Rate (RMR) to assess one-to-one correspondence reciprocity. Let the directional hard prototype mappings be

$$f(k) = \arg\max_l A_{kl}^{v\to u}, \qquad g(l) = \arg\max_k B_{lk}^{u\to v}. \tag{61}$$

If prototype k satisfies

$$g(f(k)) = k, \tag{62}$$

its forward match can return to itself through the reverse match. The Reciprocal Matching Rate (RMR) is defined as

$$\text{RMR}^{(v,u)} = \frac{1}{K}\sum_{k=1}^{K}\mathbb{I}\big[g(f(k)) = k\big]. \tag{63}$$

When the two views contain the same number of prototypes, $\text{RMR} = 1$ indicates that the bidirectional hard mappings form mutually inverse one-to-one correspondences. A many-to-one mapping prevents at least some source prototypes from returning to themselves, thereby reducing RMR.

Finally, the Structural Correspondence Reliability (SCR) is defined as the harmonic mean of MTM and RMR:

$$\boxed{\text{SCR}^{(v,u)} = \frac{2\,\text{MTM}^{(v,u)}\text{RMR}^{(v,u)}}{\text{MTM}^{(v,u)} + \text{RMR}^{(v,u)}}}. \tag{64}$$

Finally, CPAS is defined as the harmonic mean of SRS and SCR:

$$\boxed{\text{CPAS}^{(v,u)} = \frac{2\,\text{SRS}^{(v,u)}\text{SCR}^{(v,u)}}{\text{SRS}^{(v,u)} + \text{SCR}^{(v,u)}}}. \tag{65}$$

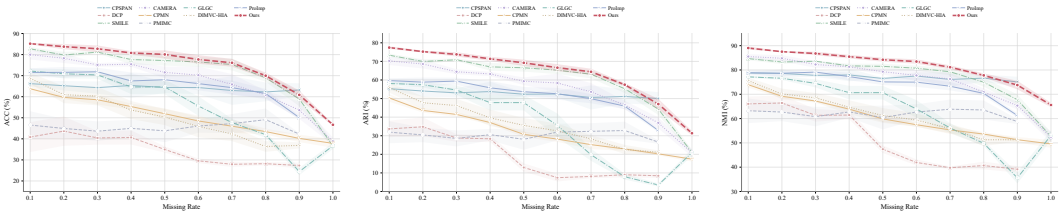
H COMPARISON RESULTS

Results under a degradation rate of 1.0. Tables 5 to 7 report the results at a degradation rate of 1.0. Under full incompleteness, ReCAP achieves the best results across all datasets and metrics, except for ARI on CUB, where it ranks third, and ARI and NMI on Hdigit, where it ranks second. When the noise rate is set to 1.0, ReCAP ranks first across all metrics on five datasets, while ranking third on Hdigit. When both the missing and noise rates are set to 1.0, ReCAP achieves the best reported results across all datasets and metrics. These results indicate that ReCAP remains competitive under severe individual and compound degradation.

Robustness to Varying Degradation Rates. As shown in Figures 5 to 7, we use DHA as a representative dataset to evaluate ReCAP under increasing rates of view incompleteness, feature noise, and their combination. Under view incompleteness, the performance of most methods declines as the missing rate increases, whereas ReCAP remains leading or competitive across the evaluated rates and degrades more gradually. Under feature noise, ReCAP shows relatively small performance variations, while several baselines exhibit non-monotonic trends as the noise rate changes. Such fluctuations may reflect differences in the sampled corruption patterns and stochastic optimization rather than consistent benefits from stronger noise. Under compound degradation, ReCAP again maintains leading or competitive performance across most degradation rates. Overall, these trends indicate that ReCAP is less sensitive to increasing individual and compound data degradation.

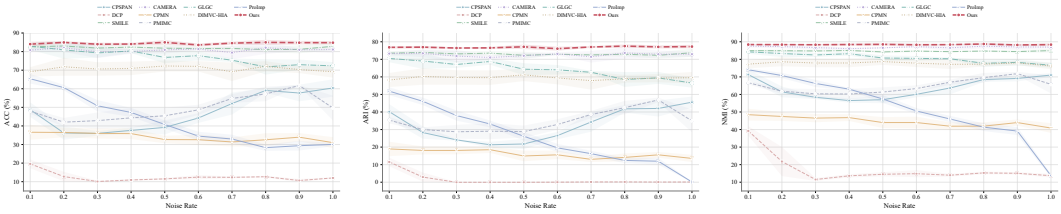
Table 5: Clustering performance (%) under fully incomplete multi-view settings.

Method	DHA			CUB			Hdigit		
	ACC	ARI	NMI	ACC	ARI	NMI	ACC	ARI	NMI
SMILE (TPAMI'24)	36.52 ± 1.40	20.65 ± 0.96	52.18 ± 0.83	38.43 ± 4.27	18.88 ± 3.54	33.81 ± 3.74	32.40 ± 2.59	14.24 ± 2.22	24.40 ± 2.61
CAMERA (TPAMI'25)	38.84 ± 1.75	20.76 ± 2.12	52.35 ± 2.08	42.87 ± 1.82	22.34 ± 1.16	37.66 ± 1.20	27.35 ± 1.09	8.88 ± 0.86	15.84 ± 1.64
CPMN (NeurIPS'25)	36.23 ± 0.74	15.49 ± 2.05	46.39 ± 1.26	42.73 ± 3.21	23.75 ± 3.42	40.75 ± 3.98	30.25 ± 2.05	12.91 ± 1.86	23.43 ± 2.13
GLGC (CVPR'26)	37.68 ± 2.96	22.46 ± 2.12	54.15 ± 2.27	42.23 ± 2.22	**24.07** ± 2.16	42.72 ± 3.10	32.60 ± 2.31	**17.25** ± 2.37	**32.06** ± 2.61
Ours	**46.67** ± 0.55	**31.34** ± 1.26	**65.59** ± 0.92	**44.80** ± 2.43	22.74 ± 2.95	**43.37** ± 2.50	**33.97** ± 1.73	14.68 ± 1.32	27.99 ± 1.61
Method	Reuters			Caltech5V			NUS-WIDE		
	ACC	ARI	NMI	ACC	ARI	NMI	ACC	ARI	NMI
GLGC (CVPR'26)	30.87 ± 2.54	6.48 ± 0.84	10.97 ± 1.00	58.71 ± 4.74	37.66 ± 4.76	48.05 ± 4.50	13.68 ± 0.37	3.63 ± 0.35	16.92 ± 0.37
Ours	**52.45** ± 1.35	**24.53** ± 1.67	**29.94** ± 1.73	**84.44** ± 1.02	**70.38** ± 1.96	**72.53** ± 1.62	**14.92** ± 0.44	**4.29** ± 0.10	**18.46** ± 0.38



(a) ACC (b) ARI (c) NMI

Figure 5: Clustering performance comparison under different missing rates.



(a) ACC (b) ARI (c) NMI

Figure 6: Clustering performance comparison under different noise rates.

Table 6: Clustering performance (%) under fully noisy multi-view settings. NaN indicates a failed run with no meaningful result.

Method	DHA			CUB			Hdigit		
	ACC	ARI	NMI	ACC	ARI	NMI	ACC	ARI	NMI
CPSPAN (CVPR'23)	60.42 ± 4.67	45.58 ± 3.21	71.04 ± 1.10	51.30 ± 3.04	33.29 ± 2.27	47.26 ± 2.09	92.11 ± 0.65	83.33 ± 1.28	82.78 ± 1.00
ProImp (IJCAI'23)	30.06 ± 0.98	13.19 ± 1.35	40.38 ± 1.52	44.73 ± 4.37	25.20 ± 2.98	41.57 ± 3.70	97.99 ± 0.10	95.59 ± 0.21	94.53 ± 0.22
DCP (TPAMI'23)	12.01 ± 0.37	0.11 ± 0.16	13.70 ± 1.01	13.00 ± 0.69	0.09 ± 0.23	2.99 ± 0.42	82.97 ± 4.51	77.29 ± 6.96	86.64 ± 2.01
SMILE (TPAMI'24)	82.22 ± 1.33	72.69 ± 1.17	84.73 ± 0.39	72.60 ± 5.37	59.32 ± 2.66	68.83 ± 1.56	96.50 ± 0.13	92.39 ± 0.28	90.88 ± 0.29
CAMERA (TPAMI'25)	81.12 ± 1.87	72.95 ± 1.54	86.98 ± 0.61	81.26 ± 4.21	67.31 ± 5.54	76.74 ± 3.65	**99.76** ± 0.03	**99.47** ± 0.08	**99.22** ± 0.10
PMIMC (TIP'25)	49.65 ± 6.65	34.93 ± 6.47	65.76 ± 5.20	63.77 ± 3.38	53.82 ± 4.70	70.08 ± 2.68	99.71 ± 0.03	99.35 ± 0.07	99.01 ± 0.09
CPMN (NeurIPS'25)	31.30 ± 3.57	13.56 ± 3.03	40.87 ± 3.20	67.53 ± 6.14	52.32 ± 6.45	65.99 ± 4.50	37.68 ± 2.16	21.89 ± 3.88	31.63 ± 4.96
GLGC (CVPR'26)	71.68 ± 3.05	56.83 ± 3.99	76.73 ± 2.25	42.27 ± 3.36	25.62 ± 2.78	44.54 ± 2.82	90.48 ± 1.19	80.18 ± 2.34	79.66 ± 2.15
DIMVC-HIA (AAAI'26)	69.15 ± 3.15	59.54 ± 3.26	76.19 ± 1.87	54.17 ± 6.04	35.82 ± 5.01	51.48 ± 4.32	NaN	NaN	NaN
Ours	**84.80** ± 0.89	**77.27** ± 0.98	**88.38** ± 0.72	**83.03** ± 2.22	**70.24** ± 1.86	**79.13** ± 0.97	98.91 ± 0.36	97.60 ± 0.78	96.96 ± 0.85
Method	Reuters			Caltech5V			NUS-WIDE		
	ACC	ARI	NMI	ACC	ARI	NMI	ACC	ARI	NMI
CPSPAN (CVPR'23)	20.15 ± 0.64	0.25 ± 0.17	0.94 ± 0.25	67.51 ± 2.32	48.27 ± 1.70	54.01 ± 1.76	11.84 ± 0.46	2.26 ± 0.26	13.54 ± 0.70
DCP (TPAMI'23)	18.02 ± 0.33	0.00 ± 0.02	1.14 ± 0.37	30.04 ± 1.98	10.98 ± 1.32	17.01 ± 1.29	9.82 ± 1.48	0.02 ± 0.06	6.31 ± 0.28
PMIMC (TIP'25)	52.22 ± 1.66	23.55 ± 1.55	29.80 ± 1.58	82.66 ± 0.77	68.03 ± 1.08	73.11 ± 0.62	14.21 ± 0.45	4.11 ± 0.31	19.15 ± 0.37
GLGC (CVPR'26)	29.32 ± 2.54	4.63 ± 1.04	6.53 ± 0.85	72.74 ± 6.88	55.49 ± 8.55	63.40 ± 5.65	15.25 ± 0.34	4.63 ± 0.41	18.53 ± 0.57
DIMVC-HIA (AAAI'26)	23.40 ± 5.99	5.65 ± 5.55	12.09 ± 9.46	23.12 ± 1.40	1.34 ± 0.31	2.80 ± 0.54	14.42 ± 1.02	0.75 ± 0.78	5.30 ± 1.86
Ours	**52.48** ± 0.69	**24.24** ± 0.56	**29.83** ± 0.79	**91.13** ± 0.37	**82.94** ± 0.78	**84.91** ± 0.89	**16.40** ± 0.38	**5.83** ± 0.24	**22.07** ± 0.37

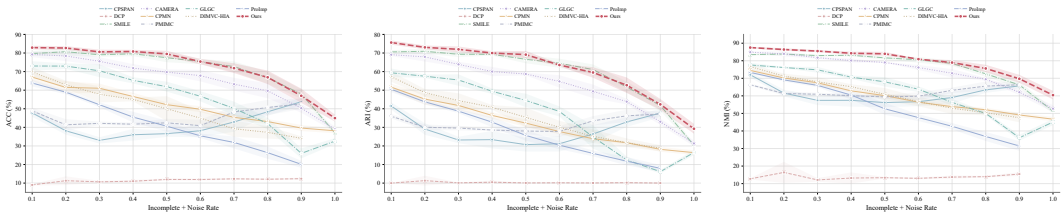
Table 7: Clustering performance (%) under fully noisy and incomplete multi-view settings.

Method	DHA			CUB			Hdigit		
	ACC	ARI	NMI	ACC	ARI	NMI	ACC	ARI	NMI
SMILE (TPAMI'24)	38.51 ± 2.51	20.83 ± 1.47	51.35 ± 1.36	37.30 ± 2.54	16.98 ± 2.50	31.19 ± 3.20	32.61 ± 1.68	13.71 ± 1.47	23.61 ± 1.65
CAMERA (TPAMI'25)	38.47 ± 1.12	20.49 ± 1.66	51.68 ± 1.63	36.67 ± 1.41	17.77 ± 1.06	33.31 ± 1.24	28.21 ± 1.23	9.46 ± 1.08	16.83 ± 1.58
CPMN (NeurIPS'25)	38.05 ± 1.45	17.84 ± 1.62	47.61 ± 1.11	40.53 ± 3.72	20.71 ± 2.07	37.61 ± 2.57	33.08 ± 3.15	13.58 ± 2.45	23.96 ± 3.08
GLGC (CVPR'26)	32.59 ± 1.57	16.44 ± 1.41	45.07 ± 1.77	30.07 ± 2.08	11.47 ± 0.97	24.55 ± 1.78	33.17 ± 1.39	16.08 ± 1.05	28.68 ± 1.69
Ours	**44.97** ± 1.87	**29.24** ± 2.66	**60.41** ± 2.44	**41.20** ± 3.08	**21.23** ± 2.47	**40.27** ± 3.59	**37.37** ± 4.22	**16.24** ± 2.10	**30.22** ± 3.95
Method	Reuters			Caltech5V			NUS-WIDE		
	ACC	ARI	NMI	ACC	ARI	NMI	ACC	ARI	NMI
GLGC (CVPR'26)	21.45 ± 0.48	0.63 ± 0.18	1.62 ± 0.31	51.04 ± 5.02	24.75 ± 5.23	37.29 ± 4.80	11.27 ± 0.43	1.35 ± 0.27	12.34 ± 0.42
Ours	**48.13** ± 0.74	**19.75** ± 0.77	**24.15** ± 0.94	**80.70** ± 0.72	**64.53** ± 1.16	**67.00** ± 1.02	**13.58** ± 0.24	**3.74** ± 0.20	**17.36** ± 0.30

I Ablation Studies

Loss Component Ablation. We conduct ablation studies on the dual-view CUB dataset and the multi-view Caltech5V dataset under incomplete multi-view settings with varying missing rates, and the clustering results are reported in Table 8. Specifically, we employ the autoencoder trained solely with the reconstruction loss as the baseline and activate the remaining objectives individually and in combination associated with the remaining modules to evaluate the effectiveness of individual components and their combinations. As can be observed, introducing $\mathcal{L}_{PCC}$ alone leads to a degradation in clustering performance. This result suggests that without sufficient representation learning and prototype warm-up, the model is unable to establish reliable prototypes with meaningful semantic structures at the early training stage. Performing cross-view prototype alignment under such conditions may reinforce inaccurate prototype correspondences and further amplify the representation discrepancy across views, thereby deteriorating the underlying clustering structure. Moreover, CUB and Caltech5V exhibit different dependencies on the proposed components. On CUB, the model relies more heavily on $\mathcal{L}_{CLU}$. Sharpening the cluster assignments first produces more compact and discriminative cluster structures, which subsequently provide a more reliable semantic basis for cross-view prototype alignment and facilitate further performance improvements. In contrast, on Caltech5V, $\mathcal{L}_{PAL}$ alone is sufficient to construct a relatively high-quality and structurally stable prototype space. Consequently, incorporating cross-view prototype alignment on top of such prototypes yields a more pronounced improvement in clustering performance.

Overall, ReCAP achieves the best performance on both datasets when all components are jointly incorporated. Notably, the complete model not only provides consistent improvements in clustering accuracy (ACC), but also achieves more substantial gains in ARI and NMI. These results demonstrate the complementary effects among the proposed modules: their integration improves not only the correctness of sample assignments but also the overall quality and structural consistency of the resulting clusters.



(a) ACC (b) ARI (c) NMI

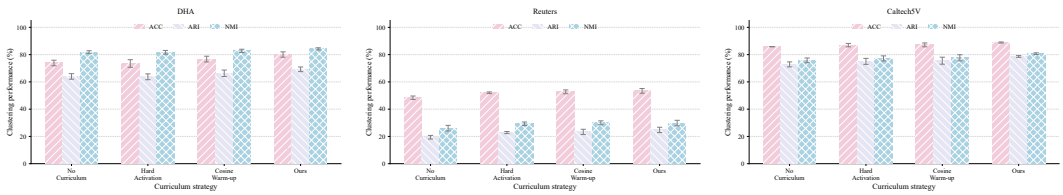
Figure 7: Clustering performance comparison under different incomplete & noise rates.

Table 8: Ablation study of different loss components on CUB and Caltech5V. Results are reported as mean±standard deviation (%). The best results are highlighted in bold.

$\mathcal{L}_{REC}$	$\mathcal{L}_{PAL}$	$\mathcal{L}_{PCC}$	$\mathcal{L}_{CLU}$	CUB						Caltech5V					
				Rate = 0.3			Rate = 0.5			Rate = 0.3			Rate = 0.5		
				ACC↑	ARI↑	NMI↑	ACC↑	ARI↑	NMI↑	ACC↑	ARI↑	NMI↑	ACC↑	ARI↑	NMI↑
✓				44.10 ± 1.20	33.85 ± 1.28	50.65 ± 1.62	37.43 ± 3.93	21.44 ± 3.87	42.08 ± 3.44	42.20 ± 4.95	29.53 ± 7.03	37.95 ± 5.74	29.23 ± 4.72	15.38 ± 4.54	21.37 ± 4.84
✓	✓			76.10 ± 3.57	58.73 ± 4.72	68.84 ± 3.32	76.17 ± 1.85	56.65 ± 2.92	66.51 ± 2.20	77.53 ± 4.83	60.77 ± 6.04	67.04 ± 3.84	78.27 ± 4.59	61.08 ± 6.01	66.53 ± 3.78
✓		✓		35.40 ± 3.26	23.07 ± 2.51	40.74 ± 1.91	30.77 ± 1.85	16.07 ± 1.29	35.07 ± 2.73	34.94 ± 3.35	11.51 ± 3.21	17.33 ± 4.20	26.66 ± 1.85	11.45 ± 1.54	16.33 ± 1.78
✓			✓	44.03 ± 3.81	33.86 ± 3.82	52.51 ± 2.86	36.10 ± 3.35	23.56 ± 3.64	44.92 ± 3.79	41.36 ± 1.88	22.70 ± 2.73	31.02 ± 3.22	33.74 ± 3.86	16.30 ± 0.84	24.07 ± 0.95
✓	✓	✓		68.97 ± 2.54	57.47 ± 2.75	68.68 ± 2.33	63.73 ± 3.64	48.87 ± 2.65	62.00 ± 2.55	89.11 ± 1.22	78.69 ± 2.30	80.94 ± 1.75	83.81 ± 2.35	69.60 ± 3.61	73.75 ± 2.31
✓	✓		✓	82.90 ± 1.74	68.14 ± 2.60	75.09 ± 1.83	80.33 ± 1.48	63.40 ± 2.37	71.34 ± 1.98	77.66 ± 7.55	64.67 ± 8.59	69.61 ± 6.04	77.87 ± 7.13	63.01 ± 8.84	68.02 ± 5.96
✓		✓	✓	37.77 ± 2.35	25.44 ± 2.55	43.75 ± 2.40	35.43 ± 1.40	21.95 ± 2.50	41.77 ± 2.32	35.46 ± 1.78	15.63 ± 2.09	21.73 ± 2.70	32.87 ± 1.15	14.10 ± 1.54	20.17 ± 1.94
✓	✓	✓	✓	**83.37 ± 2.43**	**69.10 ± 3.40**	**76.36 ± 2.30**	**81.17 ± 2.29**	**65.06 ± 2.66**	**72.62 ± 2.01**	**89.96 ± 0.94**	**80.73 ± 1.58**	**82.81 ± 1.51**	**88.89 ± 0.40**	**78.66 ± 0.64**	**80.72 ± 0.64**

PCC Objective Ablation. Table 9 isolates the contributions of the cyclic-return objective $\mathcal{L}_{\text{ret}}$ and visitation regularizer $\mathcal{L}_{\text{visit}}$. Using $\mathcal{L}_{\text{ret}}$ alone already yields strong performance on both datasets, whereas $\mathcal{L}_{\text{visit}}$ alone is markedly less effective, particularly on Caltech5V. This difference reflects their distinct roles: $\mathcal{L}_{\text{ret}}$ provides a prototype-specific cyclic correspondence signal, while $\mathcal{L}_{\text{visit}}$ only regulates the marginal coverage of the returned probability mass. Adding $\mathcal{L}_{\text{visit}}$ to the return objective improves ACC and ARI on CUB while maintaining comparable NMI. On Caltech5V, the combined objective improves ACC, ARI, and NMI over the return-only variant by 0.83, 1.52, and 1.36 percentage points, respectively. It also reduces the standard deviation of every metric relative to the return-only variant. These results indicate that $\mathcal{L}_{\text{ret}}$ provides the primary calibration signal, while $\mathcal{L}_{\text{visit}}$ discourages degenerate marginal visitation and improves run-to-run stability.

Effect of Warm-up Scheduling Strategies. We further examine how different objective scheduling strategies affect performance under the incomplete setting with a missing rate of 0.5. We compare four variants: (i) *No Curriculum*, where all objectives are optimized with their full weights from the start; (ii) *Hard Activation*, where the prototype-related objectives are switched to their full weights at the scheduled epoch; (iii) *Cosine Warm-up*, which gradually increases their weights according to a cosine schedule; and (iv) *Linear Warm-up*, adopted as the default in ReCAP, where the corresponding weights increase linearly during the warm-up stage. As shown in Figure 8, gradually introducing the prototype-related objectives is generally more effective than applying them at full strength either from the beginning or through abrupt activation. Cosine warm-up already yields strong results, while linear warm-up performs best in eight of the nine dataset–metric combinations. The difference is especially clear on Caltech5V, where the linear schedule achieves the highest ACC, ARI, and NMI and also shows noticeably smaller standard deviations across runs. Overall, these results favor a smooth introduction of the prototype-related objectives when training with incomplete views, with the linear schedule providing the most consistent performance among the tested strategies.



(a) DHA (b) Reuters (c) Caltech5V

Figure 8: Effect of different warm-up scheduling strategies on clustering performance.

Table 9: Ablation study of $\mathcal{L}_{\text{ret}}$ and $\mathcal{L}_{\text{visit}}$. The best and second-best results are highlighted in bold and underlined, respectively.

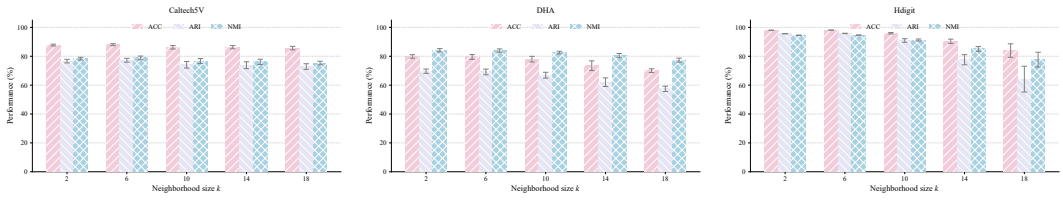
$\mathcal{L}_{\text{ret}}$	$\mathcal{L}_{\text{visit}}$	CUB			Caltech5V		
		ACC↑	ARI↑	NMI↑	ACC↑	ARI↑	NMI↑
✓		<u>81.00</u> ± 3.24	<u>64.82</u> ± 4.29	**72.69** ± 3.16	<u>88.06</u> ± 1.34	<u>77.14</u> ± 2.53	<u>79.36</u> ± 2.16
	✓	80.07 ± 0.95	62.98 ± 1.25	70.90 ± 1.23	77.87 ± 7.13	62.86 ± 8.84	67.79 ± 5.98
✓	✓	**81.17** ± 2.29	**65.06** ± 2.66	<u>72.62</u> ± 2.01	**88.89** ± 0.40	**78.66** ± 0.64	**80.72** ± 0.64

J HYPERPARAMETER ANALYSIS

This section further investigates the sensitivity of ReCAP to its key hyperparameters.

Sensitivity Analysis of Loss Weights. Specifically, we analyze the weighting coefficients associated with different loss terms on the CUB and DHA datasets under incomplete multi-view settings with missing rates of 0.3 and 0.5, respectively. The corresponding results are illustrated in Figure 11 and Figure 12. When evaluating the interaction between any two hyperparameters, the remaining one is fixed at its empirically optimal value, allowing the influence of different parameter combinations to be examined more clearly. For DHA with a missing rate of 0.5, when fixing $\lambda_1 = 1$ and varying λ_2 and λ_3, the clustering performance is mainly affected by λ_2. When λ_2 takes relatively small values, ReCAP maintains favorable clustering performance over a relatively broad range of λ_3, particularly when $\lambda_3 \leq 10$, indicating a certain degree of robustness to λ_3. Further observations from the (λ_1, λ_2) and (λ_1, λ_3) settings show that the clustering performance is more sensitive to λ_1. As λ_1 decreases, ACC, ARI, and NMI exhibit varying degrees of degradation, suggesting that an appropriate weight for the corresponding loss term is important for preserving a stable clustering structure. A broadly similar trend can be observed on CUB with a missing rate of 0.3, where the model also exhibits noticeable sensitivity to λ_1. In particular, when jointly varying λ_1 and λ_3, the clustering performance improves considerably as both hyperparameters increase, indicating a stronger complementary effect between the corresponding objectives on CUB. Overall, ReCAP maintains relatively stable clustering performance over a reasonably broad range of hyperparameter settings, while the optimal balance among different loss terms remains moderately dataset-dependent.

Sensitivity Analysis of k in MkNN. To investigate the influence of the neighborhood size k in the MkNN algorithm of the PAM, we conduct a sensitivity analysis on Hdigit, DHA, and Caltech5V under a composite incomplete-and-noisy setting with the ratio set to 0.5. The results are shown in Figure 9. Caltech5V remains relatively stable across different values of k. As k increases, the average performance decreases slightly while the standard deviation increases noticeably, with $k = 6$ showing the best stability. In contrast, the clustering performance of DHA and Hdigit decreases as k increases, accompanied by larger fluctuations. Overall, a small or moderate k is more suitable for preserving reliable local neighborhood structures, whereas an excessively large k may introduce noisy relations and degrade clustering performance.

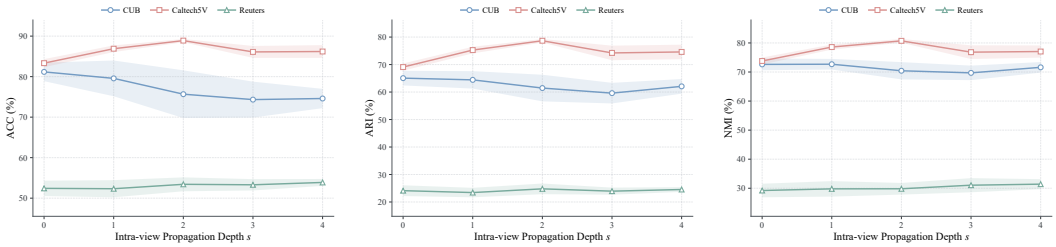


(a) Caltech5V (b) DHA (c) Hdigit

Figure 9: Sensitivity analysis of k on different datasets.

Sensitivity to the Number of Intra-View Transition Steps. We investigate the influence of the intra-view propagation depth $s \in \{0, 1, 2, 3, 4\}$. As shown in Figure 10, CUB achieves its best performance at $s = 0$, indicating that direct sample-mediated prototype transitions are sufficient, whereas additional diffusion may blur its local correspondence structure. In contrast, Caltech5V benefits from shallow propagation and achieves the best performance with the lowest variance at

$s = 2$. Further increasing s degrades its performance, suggesting that excessive probability diffusion may lead to over-smoothing. Reuters remains relatively stable across different depths: although $s = 4$ yields slightly higher ACC and NMI, its results overlap with those of $s = 2$ within the standard deviations. Overall, direct or shallow transitions ($s \leq 2$) are generally sufficient to achieve strong and stable performance, while deeper propagation provides no consistent benefit.



(a) ACC sensitivity. (b) ARI sensitivity. (c) NMI sensitivity.

Figure 10: Sensitivity of ReCAP to the intra-view propagation depth s in PCC:(a) ACC, (b) ARI, and (c) NMI. Solid lines report the mean performance over five independent runs, while the shaded regions denote one standard deviation. Setting $s = 0$ removes intra-view sample diffusion while retaining the sample-mediated cyclic prototype transition.

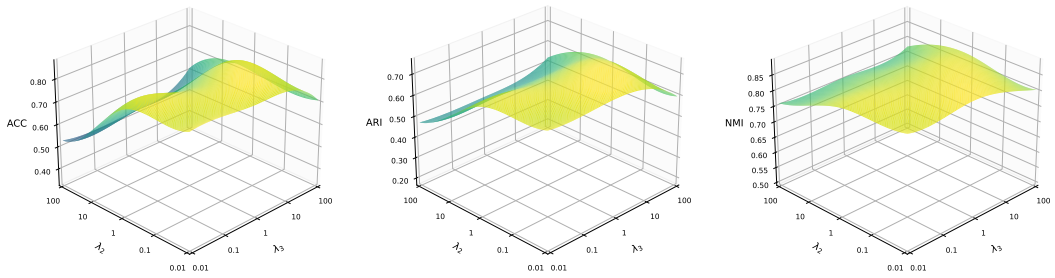
Effect of Sample-Mediated Cyclic Transitions. To examine whether sample representations provide a useful semantic bridge for cyclic prototype calibration, we compare the proposed sample-mediated cycle with a direct prototype cycle. As shown in Table 10, the sample-mediated cycle consistently improves ACC, ARI, NMI, SRS, SCR, and CPAS across all three datasets. The improvements are particularly pronounced on Caltech5V, where ACC, ARI, and NMI increase by 10.99, 15.64, and 12.77 percentage points, respectively, accompanied by substantially lower variance. On Reuters, CPAS increases from 0.5329 to 0.7259, indicating that the sample-mediated path improves both semantic reliability and structural reciprocity. Consistent gains are also observed on CUB, although the improvement in clustering performance is comparatively moderate. These results suggest that aggregating sample-level evidence provides a more reliable transition between view-specific prototype spaces than directly constructing the cycle from pairwise prototype relations.

Table 10: Comparison between the direct prototype cycle and the proposed sample-mediated cycle.

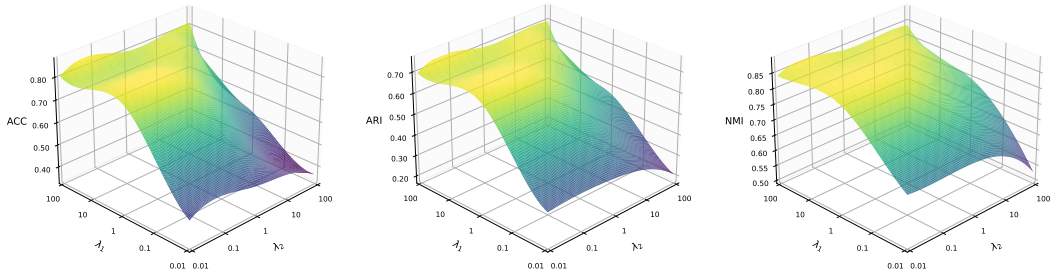
Dataset	Transition Path	ACC (%)	ARI (%)	NMI (%)	SRS	SCR	CPAS
Caltech5V	Direct prototype cycle	77.90 ± 7.10	63.02 ± 8.80	67.95 ± 5.94	0.7090	0.8479	0.7722
	Sample-mediated cycle	$\mathbf{88.89} \pm 0.40$	$\mathbf{78.66} \pm 0.64$	$\mathbf{80.72} \pm 0.64$	**0.8010**	**0.8945**	**0.8452**
Reuters	Direct prototype cycle	48.10 ± 2.56	20.09 ± 2.04	24.59 ± 2.53	0.3711	0.9448	0.5329
	Sample-mediated cycle	$\mathbf{53.42} \pm 1.72$	$\mathbf{24.78} \pm 1.95$	$\mathbf{29.82} \pm 2.01$	**0.5700**	**0.9991**	**0.7259**
CUB	Direct prototype cycle	79.37 ± 1.39	61.90 ± 1.90	70.31 ± 1.71	0.7752	0.8876	0.8276
	Sample-mediated cycle	$\mathbf{81.17} \pm 2.29$	$\mathbf{65.06} \pm 2.66$	$\mathbf{72.62} \pm 2.01$	**0.7957**	**0.9911**	**0.8827**

K Convergence and Computational Analysis

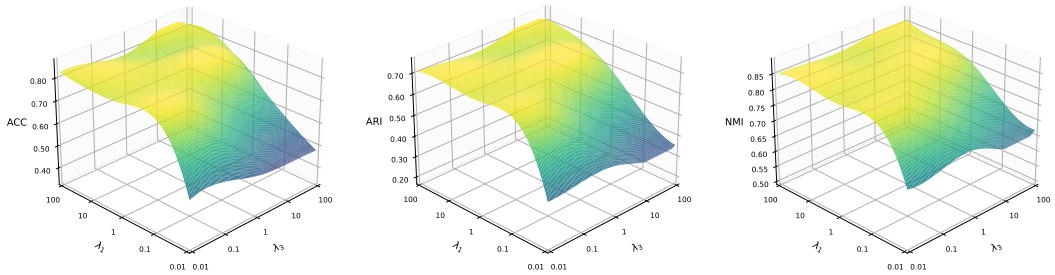
Convergence Analysis. This section analyzes the training behavior of ReCAP on all datasets under partial incompleteness with a missing rate of 0.5. As shown in Figure 13, the model exhibits a consistent convergence trend on nearly all datasets. The loss decreases rapidly in the early stage as intra- and cross-view feature aggregation is learned, decreases more gradually in the middle stage during cross-view prototype alignment, and finally stabilizes as the clustering structure converges. On NUS-WIDE, whose feature structure is more complex, the total loss increases slightly at the early stage of prototype alignment because its underlying distribution is more difficult to capture, but subsequently decreases and converges. Overall, ReCAP exhibits stable training behavior and favorable convergence across all datasets.



(a) ACC w.r.t. λ_2 and λ_3 (b) ARI w.r.t. λ_2 and λ_3 (c) NMI w.r.t. λ_2 and λ_3



(d) ACC w.r.t. λ_1 and λ_2 (e) ARI w.r.t. λ_1 and λ_2 (f) NMI w.r.t. λ_1 and λ_2



(g) ACC w.r.t. λ_1 and λ_3 (h) ARI w.r.t. λ_1 and λ_3 (i) NMI w.r.t. λ_1 and λ_3

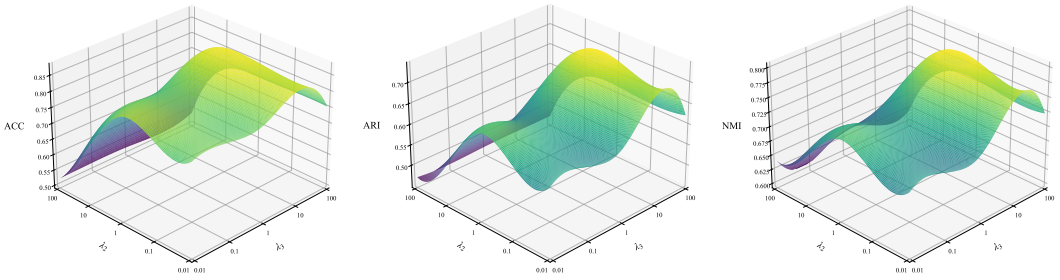
Figure 11: Hyperparameter sensitivity analysis of λ_1, λ_2, and λ_3 on the DHA dataset in terms of ACC, ARI, and NMI.

Complexity analysis. Let N, V, and d denote the number of samples, views, and the latent dimensionality, respectively, and let $\mathcal{A}$ denote the per-sample cost of the view-specific autoencoders. Autoencoder pre-training and joint representation learning require $\mathcal{O}((E_{\text{pre}} + E)N\mathcal{A})$ time. Constructing the intra- and cross-view MkNN similarity matrices requires $\mathcal{O}(EV^2N^2d)$ time. Therefore, when $s = 0$, the overall complexity is

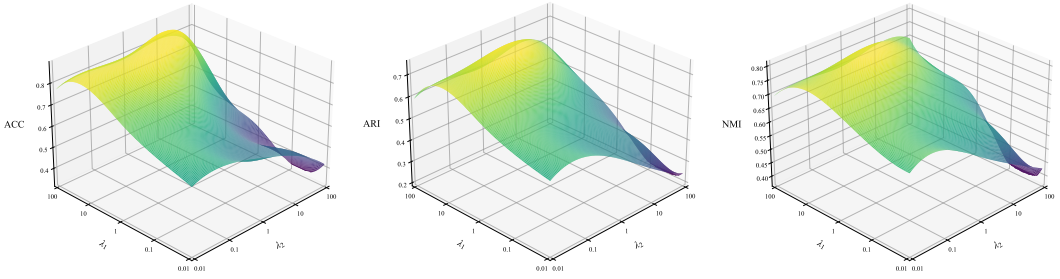
$$\mathcal{O}\big((E_{\text{pre}} + E)N\mathcal{A} + EV^2N^2d\big)\,. \tag{66}$$

When $s = 2$, dense intra-view transition-matrix multiplication introduces an additional $\mathcal{O}(E_{\text{PCC}}V^2N^3)$ cost. The space complexity is $\mathcal{O}(P + NH + V^2N^2)$.

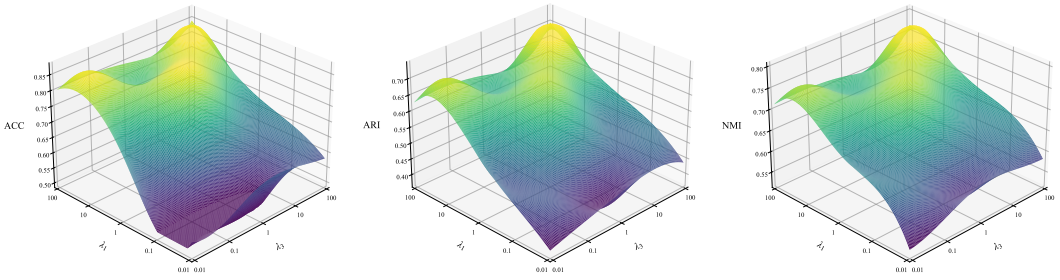
Computational cost and runtime. Table 11 reports the computational profile of ReCAP on the six datasets. *Runtime* denotes the wall-clock time required for autoencoder pre-training and joint optimization. *Params* denotes the number of trainable model parameters, measured in millions. *Peak GPU Mem.* denotes the maximum CUDA memory allocated by PyTorch during training. Metric computation, prototype-quality evaluation, and visualization are excluded from the runtime measurement.



(a) ACC w.r.t. λ_2 and λ_3 (b) ARI w.r.t. λ_2 and λ_3 (c) NMI w.r.t. λ_2 and λ_3

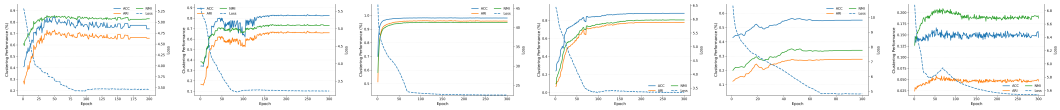


(d) ACC w.r.t. λ_1 and λ_2 (e) ARI w.r.t. λ_1 and λ_2 (f) NMI w.r.t. λ_1 and λ_2



(g) ACC w.r.t. λ_1 and λ_3 (h) ARI w.r.t. λ_1 and λ_3 (i) NMI w.r.t. λ_1 and λ_3

Figure 12: Hyperparameter sensitivity analysis of λ_1, λ_2, and λ_3 on the CUB dataset in terms of ACC, ARI, and NMI.



(a) DHA (b) CUB (c) Hdigit (d) Caltech5V (e) Reuters (f) NUS-WIDE

Figure 13: Training behavior on all datasets under partial incompleteness.

L VISUALIZATION ANALYSIS

Visualization of Learned Clustering Structures. This section further evaluates the ability of ReCAP to capture underlying cluster structures through clustering visualization. Specifically, we conduct experiments on the multi-view Caltech5V dataset and the dual-view Hdigit dataset under an incomplete setting with a missing rate of 0.5, and employ t-SNE (Maaten & Hinton, 2008) to visualize the feature representations before and after learning, as shown in Figure 14. The initial representations of both datasets exhibit relatively scattered distributions with substantial overlap among different classes. After learning with ReCAP, the samples form clearer and better-separated cluster

Table 11: Computational cost of ReCAP on six multi-view datasets. Runtime includes autoencoder pre-training and joint optimization, while evaluation and visualization are excluded.

Dataset	Runtime (s)	Params (M)	Peak GPU Mem. (GB)
DHA	57.40	12.31	0.304
CUB	50.93	11.65	0.273
Hdigit	599.70	11.07	0.327
Caltech5V	226.85	29.96	1.104
NUS-WIDE	287.41	23.65	0.980
Reuters	301.82	42.83	0.960

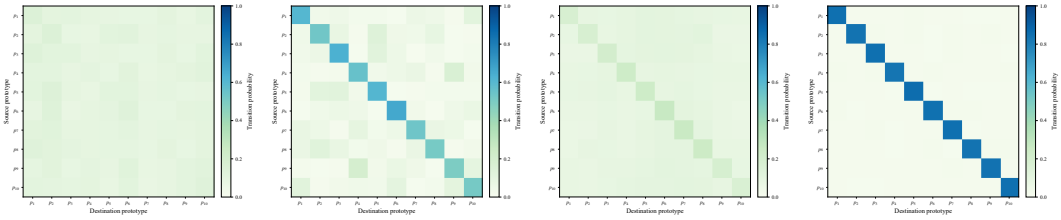
structures. These results demonstrate that ReCAP can effectively capture the underlying clustering structure and learn discriminative representations even when partial views are missing.



(a) Initial feature t-SNE on Caltech5V (b) t-SNE after clustering on Caltech5V (c) Initial feature t-SNE on Hdigit (d) t-SNE after clustering on Hdigit

Figure 14: t-SNE visualization of the learned representations on Caltech5V and Hdigit before and after clustering with a missing rate of 0.5.

Visualization of Cyclic Returnability. To qualitatively assess whether the cyclic-return objective encourages probability mass to return to its originating prototype after cross-view propagation, we visualize the cyclic transition matrices $\mathbf{T}$ with and without PCC. For the variant without PCC, $\mathbf{T}$ is computed post hoc from the learned representations using the same transition definition. As shown in Figure 15, under the compound incomplete-and-noisy setting with a degradation rate of 0.5, the matrices without PCC exhibit more diffuse probability distributions and only weak diagonal structures on both CUB and Hdigit. In contrast, incorporating PCC produces substantially stronger diagonal concentrations while suppressing off-diagonal transitions. These results provide qualitative evidence that PCC improves prototype-level cyclic returnability and reduces ambiguity in cross-view prototype transitions.



(a) CUB w/o PCC (b) CUB with PCC (c) Hdigit w/o PCC (d) Hdigit with PCC

Figure 15: Visualization of the cyclic transition matrices $\mathbf{T}$ on CUB and Hdigit under the compound setting of partial view incompleteness and feature noise. Panels (a) and (c) show the matrices computed post hoc without PCC, whereas panels (b) and (d) show the corresponding matrices obtained with PCC. The clearer diagonal structures produced by PCC indicate stronger prototype-level cyclic returnability.