HIERARCHICAL RESIDUAL GUIDANCE FOR MULTI-VIEW CLUSTERING

Anonymous authors
Paper under double-blind review

ABSTRACT

Deep multi-view clustering (MVC) commonly relies on stacked view-specific encoders to learn increasingly abstract representations. However, deeper encoders do not necessarily produce better clustering features, because repeated nonlinear transformations can progressively weaken useful structures established at earlier stages. We propose Hierarchical Residual Guidance for Multi-View Clustering (HRG-MVC), which organizes deep multi-view representation learning around a hierarchical residual encoder. At each stage, the projected input is retained through an additive shortcut and the nonlinear branch learns only an incremental update, providing an explicit information path across encoder depth. To complement this view-specific residual hierarchy, a Cross-View Transformer estimates level-wise consensus from the representations of the same sample across views. The consensus is projected and selectively introduced into the subsequent residual stage through a feature-wise gate, allowing complementary cross-view information to refine deeper features without replacing their local basis. Reconstruction, inter-view feature contrast, and inter-level assignment contrast jointly optimize the hierarchy. Experiments on six benchmark datasets show competitive clustering performance, while depth and stress-test studies demonstrate the improved stability of residual encoders as the network becomes deeper. Additional ablations characterize the role of Transformer-based consensus and selective cross-view feedback.

1 INTRODUCTION

Multi-view clustering (MVC) groups unlabeled samples from multiple heterogeneous descriptions of the same objects (Fang et al., 2023; Chao et al., 2021). Deep MVC has substantially improved representation quality through autoencoding and collaborative training (Abavisani & Patel, 2018; Li et al., 2019; Xu et al., 2021; 2023), contrastive learning (Trosten et al., 2021; Xu et al., 2022; Fu et al., 2024; Chen et al., 2023; Yang et al., 2023; Lu et al., 2024), feature aggregation (Yan et al., 2023; Dong et al., 2025; Xiao et al., 2025), and shared–specific representation learning (Chen et al., 2025c;d; Li et al., 2026b; Zou et al., 2026). Recent studies further exploit multi-scale structures, sparse cross-view relations, and relationship-aware alignment (Chen et al., 2025a; Liu et al., 2025; Cui et al., 2026; Peng et al., 2026). Most of these advances focus on how features from different views should be aligned, fused, or regularized.

A less studied issue is how view-specific information is propagated when the encoder becomes deeper. Stacking nonlinear transformations increases representation capacity but can also make previously learned structures harder to preserve, so additional depth does not necessarily translate into better clustering features. This behavior is closely related to the degradation problem that motivated residual learning (He et al., 2016b;a). Multi-level MVC methods have shown that intermediate representations remain useful. MFLVC and 3MC impose feature- or cluster-level constraints at several encoder depths (Xu et al., 2022; Chen et al., 2025b), MGCMVC and MCSF exploit multi-granularity or hierarchical information (Ju et al., 2025; Cai et al., 2026), and CLSFMVC integrates layer-wise structural graphs (Li et al., 2026a). These approaches make better use of intermediate features, but they do not directly address the stability of the view-specific feature path itself as nonlinear stages are stacked.

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.

Residual connections provide a natural way to retain an explicit path across successive transformations. AMVC-GWFF has incorporated residual blocks into multi-view autoencoders and then performs global and fine-grained cross-view fusion (Huang et al., 2026), showing that residual encoding is relevant to MVC. In this work, we place residual propagation at the center of hierarchical representation learning. Each encoder stage preserves its projected input through an additive shortcut and learns only a residual update. This design gives deeper layers direct access to the representation entering the current stage instead of rebuilding every feature solely through a new nonlinear mapping.

Residual propagation by itself, however, remains confined to each individual view. We therefore use a Cross-View Transformer as an auxiliary consensus estimator. After a residual level is formed, the Transformer summarizes the representations of the same sample across views. The resulting consensus is projected to the next stage and introduced through a learnable feature-wise gate. In this way, the residual hierarchy provides the main view-specific information path, while cross-view consensus supplies an adaptive correction that complements the local representation.

The main contributions are summarized as follows.

- We develop a hierarchical residual encoder for deep MVC to alleviate representation degradation across encoder depth. Each stage retains the projected input through an additive shortcut, providing a continuous view-specific information path through successive nonlinear transformations.
- We augment the residual hierarchy with gated cross-view guidance. A level-wise Cross-View Transformer estimates sample-dependent consensus, which is projected and selectively incorporated into the subsequent residual stage without replacing the local carrier.
- We analyze the local stability of the residual stage and derive a sufficient condition under which a gated cross-view correction can reduce representation error. Experiments on six benchmarks, including depth comparisons and a seven-block stress test, provide direct evidence for the stabilizing role of residual propagation and further examine the contribution of cross-view guidance.

2 Related Work

2.1 Deep and Contrastive Multi-View Clustering

Autoencoder-based MVC learns view-specific latent representations while preserving input information through reconstruction (Hinton & Salakhutdinov, 2006; Xie et al., 2016; Abavisani & Patel, 2018; Li et al., 2019; Xu et al., 2021; 2023). Contrastive learning has become another major line of research, using paired samples, cluster assignments, neighborhood relations, or structural cues to promote cross-view agreement (Chen et al., 2020; Trosten et al., 2021; Xu et al., 2022; Fu et al., 2024; Chen et al., 2023; Yang et al., 2023; Lu et al., 2024). Recent methods also explore global aggregation and progressive fusion (Yan et al., 2023; Dong et al., 2025; Xiao et al., 2025), shared–specific representation learning (Chen et al., 2025c;d; Li et al., 2026b; Zou et al., 2026), and structure-aware or sparse cross-view modeling (Chen et al., 2025a; Liu et al., 2025; Cui et al., 2026; Peng et al., 2026). A recent survey provides a broader review of this rapidly developing area (Chowdhury et al., 2025).

2.2 Hierarchical and Residual Multi-View Learning

Multi-level MVC methods exploit intermediate representations at different semantic depths. MFLVC performs feature- and cluster-level contrast across multiple levels (Xu et al., 2022); 3MC combines inter-view feature learning with inter-layer cluster-assignment contrast (Chen et al., 2025b); MGCMVC and MCSF further exploit multi-granularity or hierarchical information (Ju et al., 2025; Cai et al., 2026). CLSFMVC constructs a similarity graph from the fused representation at each encoder layer and then integrates these graphs for spectral clustering (Li et al., 2026a). These studies demonstrate that intermediate features contain useful clustering information.

Residual learning provides a direct path for propagating representations through stacked nonlinear transformations (He et al., 2016b;a). AMVC-GWFF incorporates residual blocks into view-

specific autoencoders and subsequently applies global and sample-adaptive fusion (Huang et al., 2026). HRG-MVC builds the view-specific hierarchy itself from residual stages and examines how this residual path behaves as encoder depth increases. Cross-view interaction is then attached to the residual hierarchy through level-wise consensus guidance, so the consensus estimated at one level can participate in the construction of deeper representations.

2.3 Transformer-Based Cross-View Modeling

Transformers model adaptive dependencies through multi-head self-attention (Vaswani et al., 2017). MVCformer applies Transformer-based interaction to multi-view clustering (Zhao et al., 2023), and TTAE combines Tensor Transformer representations with self-expression and reconstruction for affinity learning (Wang et al., 2025). In HRG-MVC, the Transformer is used only as a level-wise consensus estimator for the residual hierarchy. Its global-token representation is projected back to the view-specific streams and serves as an auxiliary correction for the subsequent residual stage.

3 Methodology

3.1 Problem Formulation and Overview

Let $\mathcal{X} = \{X^v\}_{v=1}^V$ denote a multi-view dataset with N instances and V views, where $X^v \in \mathbb{R}^{N \times d_v}$. The goal is to partition the instances into K clusters without label supervision. HRG-MVC contains L hierarchical levels; the main experiments use $L = 3$ unless otherwise stated in the depth study.

Figure 1 illustrates the overall architecture. The view-specific residual hierarchy forms the backbone of HRG-MVC. For view v at level l, the incoming representation is projected to U_l^v, and a residual MLP produces the local representation H_l^v. The projected shortcut provides the main information path across depth. Starting from the second level, the consensus G_{l-1} obtained from the preceding level supplies an additional cross-view correction through a feature-wise gate. The complete update for $l \geq 2$ is

$$Z_l^v = \underbrace{U_l^v}_{\text{local shortcut}} + \underbrace{F_l^v(U_l^v)}_{\text{local residual update}} + \underbrace{A_l^v \odot R_l^v(G_{l-1})}_{\text{gated cross-view residual}}, \tag{1}$$

with $Z_1^v = U_1^v + F_1^v(U_1^v)$. The first two terms define the hierarchical residual encoder, whereas the third term provides auxiliary cross-view guidance. The following subsections describe the residual backbone first and then the consensus-guidance module. Implementation details of the ResMLP and Cross-View Transformer blocks are given in Appendix D.

3.2 Hierarchical Residual Encoding

The hierarchical residual encoder is the main representation-learning component of HRG-MVC. Each view is processed by an independent residual stream. Let $Z_0^v = X^v$. At level l, the incoming representation is projected to the stage dimension and combined with a nonlinear residual update.

$$\begin{aligned} U_l^v &= P_l^v(Z_{l-1}^v), \\ H_l^v &= U_l^v + F_l^v(U_l^v), \end{aligned} \tag{2}$$

where P_l^v is a dimension-matching projection and F_l^v denotes the nonlinear residual branch implemented with normalization, linear layers, GELU activation (Hendrycks & Gimpel, 2016), and dropout. The projection reduces to the identity when the dimensions already match. After dimension matching, the shortcut carries U_l^v directly to the stage output, while F_l^v learns an additional nonlinear update. Repeating this structure across levels creates an explicit carrier path through the encoder and reduces the dependence of deeper representations on a sequence of transform-only mappings.

The final guided representation of each view is decoded independently,

$$\hat{X}^v = D^v(Z_L^v), \tag{3}$$

encouraging the learned hierarchy to remain informative about its source view.

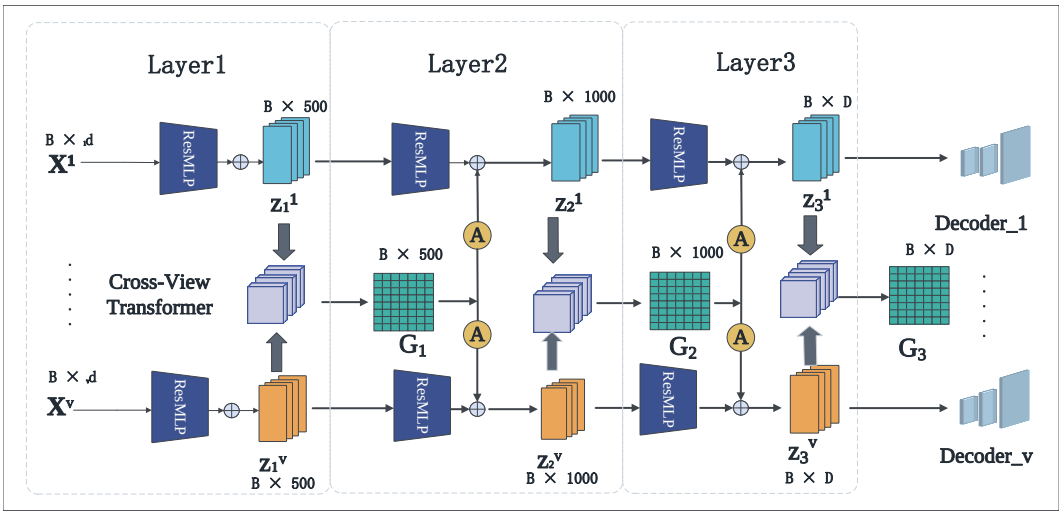

Figure 1: Forward information flow of HRG-MVC. View-specific ResMLP stages form the hierarchical residual backbone and preserve the projected input through additive shortcuts. A Cross-View Transformer estimates consensus G_l after each level and supplies it to the next residual stage as a gated auxiliary correction. Final level-wise scores are aggregated for clustering, while reconstruction and contrastive objectives regularize the hierarchy during training.

3.3 Gated Cross-View Residual Guidance

The residual hierarchy preserves view-specific information across depth, but its streams remain independent unless cross-view evidence is introduced. HRG-MVC therefore attaches an auxiliary consensus module to each level. The module estimates a sample-dependent summary from the current view representations. For each instance, the V view representations are treated as tokens, augmented with learnable view embeddings, and preceded by a learnable global token (Dosovitskiy et al., 2021). A single Transformer encoder layer performs self-attention only across the V views of that instance (Vaswani et al., 2017). We use the normalized global-token output as the level-wise consensus,

$$G_l = \mathrm{LN}\big(\mathcal{T}_l\big[C_l, Z_l^1 + E_l^1, \ldots, Z_l^V + E_l^V\big]_{\mathrm{cls}}\big), \tag{4}$$

where C_l is the learnable global token and E_l^v is the embedding identifying view v. Because the token sequence contains only the views of the same instance, G_l summarizes cross-view evidence without mixing different samples.

The consensus complements the residual backbone rather than replacing it. It is used as guidance for the next stage. For $l \geq 2$, a view-specific projection maps the preceding consensus to the current feature dimension,

$$\widetilde{G}_{l-1}^v = R_l^v(G_{l-1}). \tag{5}$$

A feature-wise gate then estimates how much of this global correction should be accepted by the current local representation,

$$A_l^v = \sigma\Big(g_l^v([H_l^v \| \widetilde{G}_{l-1}^v])\Big), \qquad Z_l^v = H_l^v + A_l^v \odot \widetilde{G}_{l-1}^v. \tag{6}$$

Here $H_l^v = U_l^v + F_l^v(U_l^v)$, g_l^v is a learned affine map, σ is the sigmoid function, and $\odot$ denotes element-wise multiplication. At the first level, $Z_1^v = H_1^v$. The forward computation follows $Z_1^v \to G_1 \to Z_2^v \to G_2 \to \cdots \to Z_L^v \to G_L$. Since G_{l-1} enters Eq. 6, cross-view information changes the representation propagated to the next stage rather than acting only through the training loss. The gate accommodates differences in the usefulness of the consensus across features and samples.

3.4 Multi-Level Objective and Inference

We optimize the hierarchical representations using reconstruction together with feature- and cluster-level contrastive constraints. The complete objective is

$$\mathcal{L} = \mathcal{L}_{rec} + \lambda_1 \mathcal{L}_{fcl} + \lambda_2 \mathcal{L}_{lcl}. \tag{7}$$

For a mini-batch of size B, the reconstruction term is

$$\mathcal{L}_{rec} = \sum_{v=1}^{V} \frac{\|X_b^v - \hat{X}_b^v\|_F^2}{Bd_v}. \tag{8}$$

At each level, the representations are normalized row-wise before feature contrast. We use cosine similarity

$$\mathrm{sim}(a, b) = \frac{a^\top b}{\|a\|_2 \|b\|_2}. \tag{9}$$

For a view pair (v, w), representations of the same instance form positive pairs, while the remaining instances in the paired mini-batches serve as negatives. The directional feature contrast is

$$\ell_f^{l,v\to w} = -\frac{1}{B}\sum_{i=1}^{B} \log \frac{\exp(\mathrm{sim}(Z_{l,i}^v, Z_{l,i}^w)/\tau_f)}{\sum_{u\in\{v,w\}}\sum_{j=1}^{B} \exp(\mathrm{sim}(Z_{l,i}^v, Z_{l,j}^u)/\tau_f) - \exp(1/\tau_f)}. \tag{10}$$

The subtracted term removes the anchor's self-similarity. Averaging the two directions and summing over all levels and view pairs gives

$$\mathcal{L}_{fcl} = \frac{1}{2}\sum_{l=1}^{L}\sum_{v<w}\left(\ell_f^{l,v\to w} + \ell_f^{l,w\to v}\right). \tag{11}$$

Each level-wise consensus is mapped to cluster scores and soft assignments,

$$S_l = \psi_l(G_l), \qquad \Pi_l = \mathrm{softmax}(S_l). \tag{12}$$

The response of cluster k at level l is $r_{l,k} = \Pi_l(:, k) \in \mathbb{R}^B$. For two levels l and m, response vectors with the same cluster index form positive pairs. Their directional contrast is

$$\ell_c^{l\to m} = -\frac{1}{K}\sum_{k=1}^{K} \log \frac{\exp(\mathrm{sim}(r_{l,k}, r_{m,k})/\tau_c)}{\sum_{n\in\{l,m\}}\sum_{j=1}^{K} \exp(\mathrm{sim}(r_{l,k}, r_{n,j})/\tau_c) - \exp(1/\tau_c)}. \tag{13}$$

The inter-level cluster loss averages the two directions and retains the balance term contributed by every level pair.

$$\mathcal{L}_{lcl} = \frac{1}{2}\sum_{l<m}\left(\ell_c^{l\to m} + \ell_c^{m\to l}\right) + \sum_{l<m}(\mathcal{B}_l + \mathcal{B}_m). \tag{14}$$

Here, τ_c is the cluster-level temperature. The balance term is

$$\begin{aligned} \bar{p}_{l,k} &= \frac{1}{B}\sum_{i=1}^{B} \Pi_l(i, k), \\ \mathcal{B}_l &= \log K + \sum_{k=1}^{K} \bar{p}_{l,k} \log(\bar{p}_{l,k} + \epsilon), \end{aligned} \tag{15}$$

which discourages batch-level assignments from concentrating on a small subset of clusters.

We first pretrain the view-specific autoencoders using reconstruction with cross-view interaction disabled. We then jointly optimize all modules with hierarchical guidance enabled. During inference, the unnormalized level-wise cluster scores are averaged.

$$Q = \frac{1}{L}\sum_{l=1}^{L} S_l, \qquad Y_i = \arg\max_k Q(i, k), \tag{16}$$

where Y_i is the predicted cluster of instance i. The soft assignments Π_l are used by the training objective, whereas inference averages the raw scores S_l. Algorithm B.1 summarizes the complete optimization procedure.

The complete optimization procedure is summarized in Algorithm A.1 in Appendix B.

3.5 Computational Complexity

For a mini-batch of B samples, the local residual encoder has cost dominated by the view-specific linear projections and MLP blocks. The cross-view Transformer operates on only $V + 1$ tokens for each sample, giving a quadratic attention term in the number of views rather than in the number of samples. The feature contrast loss introduces the main B^2 training term through pairwise similarities within a mini-batch, while the inter-level assignment contrast is smaller for the settings considered here. For fixed V, L, and network widths, inference is linear in the number of samples. Detailed expressions for the encoder, Transformer, guidance, and contrastive components are provided in Appendix C.

4 Experiments

We evaluate HRG-MVC on six benchmark datasets and compare it with representative deep MVC methods. The mechanism studies focus first on residual preservation and depth stability, and then examine how Transformer-based consensus and feature-wise gating complement the residual hierarchy.

4.1 Experimental Settings

Datasets and metrics. We evaluate HRG-MVC on six publicly available multi-view datasets, namely Caltech-5V (Fei-Fei et al., 2004), CIFAR-10 (Krizhevsky, 2009), Handwritten, NoisyMNIST, UCI-Digit (Duin, 1998), and ALOI-100 (Geusebroek et al., 2005). These benchmarks vary substantially in sample size, number of views, and clustering granularity. Table 1 summarizes the statistics required to interpret the experiments. Clustering performance is measured using clustering accuracy (ACC), normalized mutual information (NMI), and adjusted Rand index (ARI). Higher values indicate better performance for all three metrics.

Table 1: Statistics of the six multi-view benchmark datasets.

Dataset	Samples	Views	Clusters
Caltech-5V	1,400	5	7
CIFAR-10	50,000	3	10
Handwritten	2,000	6	10
NoisyMNIST	50,000	2	10
UCI-Digit	2,000	3	10
ALOI-100	10,800	4	100

Compared methods. We compare HRG-MVC with representative deep MVC baselines spanning contrastive alignment, aggregation, hierarchical learning, and shared–specific modeling, including CoMVC (Trosten et al., 2021), GCFAggMVC (Yan et al., 2023), DealMVC (Yang et al., 2023), 3MC (Chen et al., 2025b), MGCMVC (Ju et al., 2025), DMVCS (Li et al., 2026b), LDRFMVC (Zou et al., 2026), MCSF (Cai et al., 2026), and CLSFMVC (Li et al., 2026a). Table 2 focuses on methods whose results can be evaluated under matched dataset protocols or reproduced from released implementations. CoMVC and DealMVC values in the current table follow the DMVCS comparison on matching datasets. For CLSFMVC, the best λ_{fc} values are 0.01, 0.01, 0.001, 10, 0.01, and 0.001 in the dataset order of Table 2; the other listed baselines are reproduced from released implementations.

Implementation details. All HRG-MVC experiments use seed 5 unless a multi-seed study is stated explicitly. Each complete view is independently min–max normalized to $[0, 1]$. The model is trained for 200 reconstruction-pretraining epochs followed by 400 joint-training epochs, and evaluation is strictly performed on the final model at epoch 600. Intermediate evaluations are used only for monitoring and never for checkpoint selection. Training uses shuffled mini-batches with incomplete final mini-batches dropped; evaluation uses all samples.

Experiments are conducted on one NVIDIA GeForce RTX 4090 D GPU under Ubuntu 22.04.5 using Python 3.10.20 and PyTorch 2.2.2. Adam (Kingma & Ba, 2015) is used with a learning rate of 3×10^{-4} and zero weight decay. The module dropout rate is 0.2, the first encoder block applies

feature dropout at a rate of 0.15, and every Cross-View Transformer uses four attention heads. The feature- and cluster-level temperatures are set to 0.5 and 1.0, respectively.

We use the following dataset-specific $(B, \lambda_1, \lambda_2)$ settings. $(128, 0.01, 0.1)$ for Caltech-5V, $(512, 10^{-4}, 10^{-4})$ for CIFAR-10, $(512, 10^{-4}, 0.1)$ for Handwritten, $(512, 0.1, 0.001)$ for NoisyMNIST, $(128, 0.01, 0.001)$ for UCI-Digit, and $(512, 0.01, 0.01)$ for ALOI-100. For the controlled architecture studies, each ablated variant is independently retuned over the same predefined grid of loss weights and under the same search budget. The full HRG-MVC retains the dataset-specific configuration used in the main comparison, while the seed, training schedule, optimizer, and final-epoch evaluation protocol remain unchanged.

4.2 Comparison with Representative Methods

Table 2: Clustering results on six multi-view benchmarks. Method labels report publication year and venue. Best results are shown in bold and second-best results are underlined.

	Caltech-5V			CIFAR-10			Handwritten		
Method	ACC	NMI	ARI	ACC	NMI	ARI	ACC	NMI	ARI
CoMVC (2021 CVPR)	0.5324	0.3937	0.3166	0.8546	0.7517	0.7276	0.8367	0.8227	0.7599
DealMVC (2023 ACM)	0.6423	0.5373	0.4504	0.3400	0.3161	0.2147	0.4780	0.4545	0.3036
GCFAggMVC (2023 CVPR)	0.3287	0.1579	0.0953	0.4795	0.5746	0.3913	0.7159	0.7915	0.6395
3MC (2025 IF)	0.9136	0.8380	0.8238	0.9930	0.9811	0.9847	0.8875	0.8695	0.8209
MGCMVC (2025 TNNLS)	0.8080	0.7350	0.6720	0.9930	0.9800	0.9840	0.8220	0.8020	0.7270
DMVCS (2026 TKDE)	0.8429	0.8070	0.7551	0.9405	0.8739	0.8751	<u>0.9783</u>	<u>0.9499</u>	<u>0.9523</u>
LDRFMVC (2026 TPAMI)	<u>0.9186</u>	<u>0.8467</u>	<u>0.8334</u>	0.9892	0.9736	0.9760	0.9105	0.8489	0.8163
MCSF (2026 CVPR)	0.9100	0.8252	0.8157	<u>0.9934</u>	<u>0.9816</u>	<u>0.9856</u>	0.9615	0.9171	0.9164
CLSFMVC (2026 ESWA)	0.8771	0.8070	0.7581	0.9918	0.9776	0.9819	0.9685	0.9273	0.9311
HRG-MVC (Ours)	**0.9207**	**0.8470**	**0.8353**	**0.9944**	**0.9843**	**0.9876**	**0.9845**	**0.9629**	**0.9660**

	NoisyMNIST			UCI-Digit			ALOI-100		
Method	ACC	NMI	ARI	ACC	NMI	ARI	ACC	NMI	ARI
CoMVC (2021 CVPR)	0.4776	0.4413	0.3256	0.7692	0.7693	0.6717	0.1260	0.4093	0.0757
DealMVC (2023 ACM)	0.8861	0.9392	0.8746	0.5301	0.5612	0.3838	0.0250	0.1042	0.0030
GCFAggMVC (2023 CVPR)	0.9830	0.9528	0.9632	0.5898	0.6327	0.4340	0.4248	0.8462	0.5001
3MC (2025 IF)	0.9882	0.9663	0.9741	0.9000	0.8288	0.7951	0.7893	0.9158	0.7571
MGCMVC (2025 TNNLS)	<u>0.9908</u>	<u>0.9722</u>	<u>0.9800</u>	0.6840	0.7000	0.5710	0.6240	0.8180	0.5530
DMVCS (2026 TKDE)	0.6971	0.6805	0.5506	0.9717	0.9359	0.9382	**0.9310**	**0.9599**	**0.8988**
LDRFMVC (2026 TPAMI)	0.9002	0.8038	0.7960	0.9550	0.9032	0.9023	O/M	O/M	O/M
MCSF (2026 CVPR)	0.9354	0.9128	0.8899	<u>0.9735</u>	<u>0.9390</u>	<u>0.9421</u>	0.8834	0.9336	<u>0.8429</u>
CLSFMVC (2026 ESWA)	0.9714	0.9268	0.9391	0.9610	0.9166	0.9152	0.8121	0.8956	0.6049
HRG-MVC (Ours)	**0.9923**	**0.9762**	**0.9833**	**0.9740**	**0.9395**	**0.9432**	<u>0.8873</u>	<u>0.9348</u>	0.8402

Overall comparison. Table 2 shows that HRG-MVC achieves consistently strong performance across datasets with substantially different sample sizes, numbers of views, and clustering granularities. It obtains the best reported ACC, NMI, and ARI on five of the six benchmarks. The results on Caltech-5V, Handwritten, and NoisyMNIST show that the residual hierarchy effectively integrates complementary information from heterogeneous views. On CIFAR-10 and UCI-Digit, HRG-MVC maintains leading performance in regimes where recent methods already produce highly accurate partitions.

CLSFMVC provides a directly relevant hierarchical comparison because it constructs similarity graphs from intermediate representations and fuses them across layers (Li et al., 2026a). HRG-MVC performs consistently better under the reported protocols. The two methods exploit hierarchy in different ways: CLSFMVC aggregates structures extracted from completed layers, whereas HRG-MVC builds each view through a residual information path and allows cross-view consensus to refine subsequent stages. The controlled studies below examine the residual backbone and the guidance module separately.

ALOI-100 contains 100 categories and therefore emphasizes considerably finer-grained distinctions than the other benchmarks. DMVCS performs better on this benchmark by explicitly separating shared and view-dependent information. Even in this setting, HRG-MVC achieves 88.73% ACC

and 93.48% NMI, ranking second on both metrics while retaining strong ARI. The result on ALOI-100 also shows that a stable residual hierarchy does not eliminate the difficulty of fine-grained cross-view fusion, motivating more conservative use of consensus when discriminative cues are strongly view dependent.

4.3 Mechanism Analysis

Residual Backbone and Cross-View Consensus

We first examine the residual backbone and then the effect of replacing fixed mean aggregation with Transformer-based consensus. The complete model uses the configuration reported in Table 2, and the ablated architectures follow the common retuning protocol defined in Section 4.1.

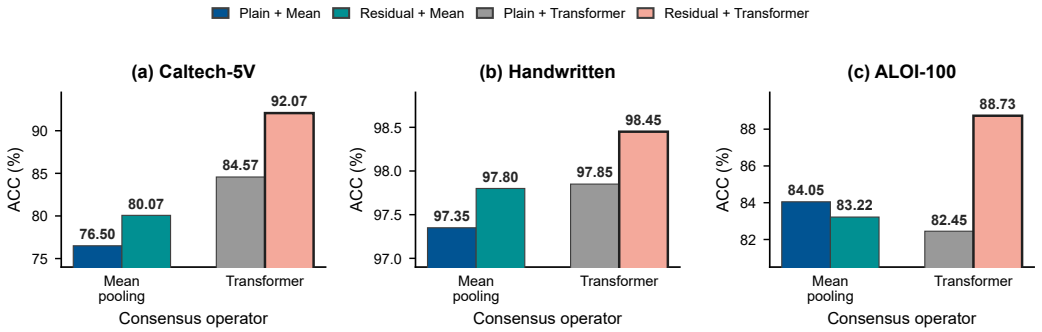

Figure 2: Residual–consensus interaction on Caltech-5V, Handwritten, and ALOI-100 in terms of ACC (%). Bars compare plain and residual view-specific encoders under mean-pooling and Transformer consensus operators.

Figure 2 shows that the effect of the consensus operator depends on the quality of the view-specific representation supplied to it. On Caltech-5V, replacing the plain encoder with the residual encoder improves both mean pooling and Transformer aggregation. Handwritten is comparatively insensitive because all four variants already form strong partitions. On ALOI-100, neither component is uniformly advantageous in isolation, while the residual-plus-Transformer configuration gives the strongest result among the four variants. The comparison supports using residual encoding as the primary feature path and treating adaptive consensus as a complementary module.

Residual Shortcuts across Encoder Depths

Before comparing depth, we scale $\mathcal{L}_{fcl}$ by $3/L$ and $\mathcal{L}_{lcl}$ by $3/\binom{L}{2}$. This normalization prevents deeper models from receiving a larger aggregate contrastive contribution merely because they expose more supervised levels and level pairs.

Table 3: Encoder-depth comparison with depth-normalized contrastive losses and a fixed decoder (%).

Depth	Residual	Caltech-5V			Handwritten			ALOI-100		
		ACC	NMI	ARI	ACC	NMI	ARI	ACC	NMI	ARI
3	No	88.64	80.30	78.02	97.95	95.06	95.53	82.45	89.31	74.84
	Yes	**92.07**	**84.70**	**83.53**	**98.45**	**96.29**	**96.60**	**88.73**	**93.48**	**84.02**
5	No	61.36	50.16	40.63	97.30	93.90	94.14	50.36	72.40	39.92
	Yes	**86.79**	**80.18**	**76.50**	**97.80**	**94.72**	**95.16**	**82.28**	**90.59**	**76.50**
7	No	28.50	34.26	20.03	92.60	89.53	86.54	66.99	82.77	57.67
	Yes	**85.86**	**76.82**	**72.99**	**96.85**	**92.73**	**93.15**	**84.74**	**90.34**	**77.72**

Table 3 shows a clear difference between plain and residual encoders as depth increases. The plain encoder deteriorates sharply on Caltech-5V and ALOI-100, whereas the corresponding residual models retain substantially stronger clustering performance. Handwritten remains comparatively

stable across depths, with residual variants consistently preserving stronger performance as the encoder deepens. These results support the use of additive shortcuts when hierarchical encoders contain more nonlinear stages.

Cross-View Guidance and Feature-Wise Gating

After establishing the role of the residual backbone, we isolate the effect of returning cross-view consensus to that hierarchy. Three variants are compared. The *no-guidance* variant retains the level-wise Cross-View Transformers and multi-level objectives but removes the feedback term, so $Z_l^v = H_l^v$. The *ungated-guidance* variant restores the feedback path and directly adds the projected consensus,

$$Z_l^v = H_l^v + R_l^v(G_{l-1}), \qquad l \geq 2. \tag{17}$$

The full HRG-MVC model uses the feature-wise gate in Eq. 6. Thus, the comparison separates the effect of cross-view feedback from the effect of selectively controlling its contribution.

Table 4 shows that the effect of cross-view feedback depends on the dataset. On Caltech-5V, returning the consensus to the residual hierarchy improves the clustering result, and feature-wise gating gives the strongest performance among the three variants. Handwritten is already well separated without feedback; direct ungated injection lowers all three metrics, whereas the gated model largely recovers the performance of the no-guidance variant. These observations do not imply that cross-view feedback is uniformly beneficial. Rather, they indicate that its contribution should be controlled instead of being added with a fixed strength.

Table 4: Ablation of cross-view guidance and feature-wise gating (%). The no-guidance variant removes the return path from G_{l-1} to the subsequent residual stage, while the ungated variant uses a unit gate. The full model retains its main-result configuration. Bold denotes the best result for each dataset and metric.

Dataset	Variant	Guidance	Gate	ACC	NMI	ARI
Caltech-5V	No guidance	×	–	84.29	77.51	72.09
	Ungated guidance	✓	×	88.86	82.63	79.63
	Gated guidance (HRG-MVC)	✓	✓	**92.07**	**84.70**	**83.53**
Handwritten	No guidance	×	–	**98.50**	**96.37**	**96.70**
	Ungated guidance	✓	×	97.70	94.53	94.97
	Gated guidance (HRG-MVC)	✓	✓	98.45	96.29	96.60

5 Conclusion

We proposed HRG-MVC with hierarchical residual encoding as its main representation-learning mechanism. Each view is processed through residual MLP stages that preserve the projected stage input and learn incremental nonlinear updates, providing an explicit information path across encoder depth. A Cross-View Transformer complements this backbone by estimating level-wise consensus, which is projected and selectively returned to the subsequent residual stage through a feature-wise gate. The theoretical analysis shows that the residual stage retains a non-vanishing local differential path under a bounded-Jacobian condition and characterizes when a gated cross-view correction can reduce representation error. Experiments on six benchmarks demonstrate strong clustering performance across diverse multi-view settings. More importantly, the depth comparison and seven-block stress test show that residual shortcuts substantially improve the stability of deeper encoders. The guidance ablation indicates that cross-view feedback can further refine the residual hierarchy, while its contribution depends on the dataset and benefits from selective control. Future work will investigate more reliable consensus estimation for fine-grained and strongly view-dependent settings.

STATEMENTS

AI USE STATEMENT

Generative AI tools were used for language editing, LaTeX assistance, and code review. No experimental result was generated or altered by a generative AI system.

REPRODUCIBILITY STATEMENT

Section 3 specifies the forward architecture and objectives, Section 4.1 reports the main optimization protocol, and Section 4.3 isolates the proposed mechanisms.

REFERENCES

M. Abavisani and V. M. Patel. Deep multimodal subspace clustering networks. *IEEE Journal of Selected Topics in Signal Processing*, 12(6):1601–1614, 2018.

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

G. Chao, S. Sun, and J. Bi. A survey on multiview clustering. *IEEE Transactions on Artificial Intelligence*, 2(2):146–168, 2021. doi: 10.1109/TAI.2021.3065894.

J. Chen, H. Mao, W. L. Woo, and X. Peng. Deep multi-view clustering by contrasting cluster assignments. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 16752–16761, 2023.

Liang Chen, Zhe Xue, Yawen Li, Meiyu Liang, Yan Wang, Anton van den Hengel, and Yuankai Qi. Medusa: A multi-scale high-order contrastive dual-diffusion approach for multi-view clustering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10295–10304, 2025a.

T. Chen, S. Kornblith, M. Norouzi, and G. Hinton. A simple framework for contrastive learning of visual representations. In *Proceedings of the International Conference on Machine Learning*, pp. 1597–1607, 2020.

Z. Chen, X.-J. Wu, T. Xu, H. Li, and J. Kittler. Multi-layer multi-level comprehensive learning for deep multi-view clustering. *Information Fusion*, 116:102785, 2025b. doi: 10.1016/j.inffus.2024.102785.

Zhe Chen, Xiao-Jun Wu, Tianyang Xu, and Josef Kittler. DFL-Net: Disentangled feature learning network for multi-view clustering. *IEEE Transactions on Knowledge and Data Engineering*, 37 (8):4537–4547, 2025c. doi: 10.1109/TKDE.2025.3574150.

Zhe Chen, Xiao-Jun Wu, Tianyang Xu, Hui Li, and Josef Kittler. Deep discriminative multi-view clustering. *IEEE Transactions on Circuits and Systems for Video Technology*, 35(7):6974–6978, 2025d. doi: 10.1109/TCSVT.2025.3541928.

Anal Roy Chowdhury, Avisek Gupta, and Swagatam Das. Deep multi-view clustering: A comprehensive survey of the contemporary techniques. *Information Fusion*, 119:103012, 2025. doi: 10.1016/j.inffus.2025.103012.

Jinrong Cui, Xiaohuang Wu, Haitao Zhang, Chongjie Dong, and Jie Wen. Structure-guided deep multi-view clustering. *Information Fusion*, 125:103461, 2026. doi: 10.1016/j.inffus.2025.103461.

Zhibin Dong, Meng Liu, Siwei Wang, Ke Liang, Yi Zhang, Suyuan Liu, Jiaqi Jin, Xinwang Liu, and En Zhu. Enhanced then progressive fusion with view graph for multi-view clustering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15518–15527, 2025.

A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *Proceedings of the International Conference on Learning Representations*, 2021.

Robert Duin. Multiple features. UCI Machine Learning Repository, 1998. URL `https://archive.ics.uci.edu/dataset/72/multiple+features`.

Uno Fang, Man Li, Jianxin Li, Longxiang Gao, Tao Jia, and Yanchun Zhang. A comprehensive survey on multi-view clustering. *IEEE Transactions on Knowledge and Data Engineering*, 35 (12):12350–12368, 2023. doi: 10.1109/TKDE.2023.3270311.

Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental Bayesian approach tested on 101 object categories. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, 2004. doi: 10.1109/CVPR.2004.383.

Lele Fu, Sheng Huang, Lei Zhang, Jinghua Yang, Zibin Zheng, Chuanfu Zhang, and Chuan Chen. Subspace-contrastive multi-view clustering. *ACM Transactions on Knowledge Discovery from Data*, 18(9):211:1–211:35, 2024. doi: 10.1145/3674839.

Jan-Mark Geusebroek, Gertjan J. Burghouts, and Arnold W. M. Smeulders. The amsterdam library of object images. *International Journal of Computer Vision*, 61(1):103–112, 2005. doi: 10.1023/B:VISI.0000042993.50813.60.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *Proceedings of the European Conference on Computer Vision*, pp. 630–645, 2016a. doi: 10.1007/978-3-319-46493-0_38.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016b. doi: 10.1109/CVPR.2016.90.

D. Hendrycks and K. Gimpel. Gaussian error linear units (GELUs). *arXiv preprint arXiv:1606.08415*, 2016.

G. E. Hinton and R. R. Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313(5786):504–507, 2006.

Xin Huang, Yan Xu, Bing Gao, Guanqiu Qi, Yuanyuan Li, and Zhiqin Zhu. Adaptive multi-view clustering with global weighting and fine-grained feature fusion. *Knowledge-Based Systems*, 349: 116428, 2026. doi: 10.1016/j.knosys.2026.116428.

Hengrong Ju, Yang Lu, Weiping Ding, Wei Zhang, and Xibei Yang. Multigranularity information fused contrastive learning with multiview clustering. *IEEE Transactions on Neural Networks and Learning Systems*, 36(10):18915–18929, 2025. doi: 10.1109/TNNLS.2025.3574885.

D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In *Proceedings of the International Conference on Learning Representations*, 2015.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009. URL `https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf`.

Jun Li, Guojun Li, Haojie Du, Kai Xu, and Baozhu Chen. Cross-layer structure information fusion for deep multi-view clustering. *Expert Systems with Applications*, 301:130448, 2026a. doi: 10.1016/j.eswa.2025.130448.

P. Li, D. Chang, Y. Wang, Z. Kong, L. Kong, and Y. Zhao. Disentangled contrastive multi-view clustering via semantic relevance invariance. *IEEE Transactions on Knowledge and Data Engineering*, 38(3):1622–1635, 2026b. doi: 10.1109/TKDE.2026.3656269.

Z. Li, Q. Wang, Z. Tao, Q. Gao, and Z. Yang. Deep adversarial multi-view clustering network. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pp. 2952–2958, 2019.

Ruimeng Liu, Xin Zou, Chang Tang, Xiao Zheng, Xingchen Hu, Kun Sun, and Xinwang Liu. SparseMVC: Probing cross-view sparsity variations for multi-view clustering. In *Advances in Neural Information Processing Systems*, volume 38, 2025.

Y. Lu, Y. Lin, M. Yang, D. Peng, P. Hu, and X. Peng. Decoupled contrastive multi-view clustering with high-order random walks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 14193–14201, 2024.

Shuangmei Peng, Zhe Chen, Tianyang Xu, and Xiao-Jun Wu. Relationship alignment for view-aware multi-view clustering. In *Proceedings of the International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=uRA9cT4MK6.

D. J. Trosten, S. Lokse, R. Jenssen, and M. Kampffmeyer. Reconsidering representation alignment for multi-view clustering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1255–1265, 2021.

Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9(86):2579–2605, 2008.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Qianqian Wang, Zihao Zhang, Wei Feng, Zhiqiang Tao, and Quanxue Gao. Contrastive multi-view subspace clustering via tensor transformers autoencoder. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39(20):21207–21215, 2025. doi: 10.1609/aaai.v39i20.35419.

Baili Xiao, Zhibin Dong, Ke Liang, Suyuan Liu, Siwei Wang, Tianrui Liu, Xingchen Hu, En Zhu, and Xinwang Liu. EASEMVC: Efficient dual selection mechanism for deep multi-view clustering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20716–20726, 2025.

J. Xie, R. Girshick, and A. Farhadi. Unsupervised deep embedding for clustering analysis. In *Proceedings of the International Conference on Machine Learning*, pp. 478–487, 2016.

J. Xu, Y. Ren, G. Li, L. Pan, C. Zhu, and Z. Xu. Deep embedded multi-view clustering with collaborative training. *Information Sciences*, 573:279–290, 2021. doi: 10.1016/j.ins.2020.12.073.

J. Xu, H. Tang, Y. Ren, L. Peng, X. Zhu, and L. He. Multi-level feature learning for contrastive multi-view clustering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16051–16060, 2022.

Jie Xu, Yazhou Ren, Huayi Tang, Zhimeng Yang, Lili Pan, Yang Yang, Xiaorong Pu, Philip S. Yu, and Lifang He. Self-supervised discriminative feature learning for deep multi-view clustering. *IEEE Transactions on Knowledge and Data Engineering*, 35(7):7470–7482, 2023. doi: 10.1109/TKDE.2022.3193569.

W. Yan, Y. Zhang, C. Lv, C. Tang, G. Yue, L. Liao, and W. Lin. GCFAgg: Global and cross-view feature aggregation for multi-view clustering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19863–19872, 2023.

Xihong Yang, Jiaqi Jin, Siwei Wang, Ke Liang, Yue Liu, Yi Wen, Suyuan Liu, Sihang Zhou, Xinwang Liu, and En Zhu. DealMVC: Dual contrastive calibration for multi-view clustering. In *Proceedings of the 31st ACM International Conference on Multimedia*, pp. 337–346, 2023. doi: 10.1145/3581783.3611951.

Mingyu Zhao, Weidong Yang, and Feiping Nie. MVCformer: A transformer-based multi-view clustering method. *Information Sciences*, 649:119622, 2023. doi: 10.1016/j.ins.2023.119622.

Xin Zou, Ruimeng Liu, Chang Tang, Zhenglai Li, Xinwang Liu, Kunlun He, and Wanqing Li. Learning disentangled representations for generalized multi-view clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2026. doi: 10.1109/TPAMI.2026.3687339. Early access.

APPENDIX

A THEORETICAL ANALYSIS

The formal analysis supporting the residual information path and auxiliary cross-view guidance is given below.

Proposition A.1 (Local stability of the residual carrier)**.** *Condition on a fixed realization of the stochastic dropout operations (or equivalently consider inference mode), and let $H(U) = U + F(U)$ denote one local residual stage with respect to the projected state entering the residual addition. Assume that F is differentiable and that $\|J_F(U)\|_2 \le \rho < 1$ in a neighborhood of interest. Then*

$$J_H(U) = I + J_F(U), \qquad 1 - \rho \le \sigma_{\min}(J_H(U)) \le \sigma_{\max}(J_H(U)) \le 1 + \rho. \tag{A.1}$$

Thus, with respect to the projected input U, no infinitesimal direction is annihilated by the residual stage under the stated local condition.

Proof. For any unit vector x, the reverse triangle inequality gives

$$\|(I + J_F)x\|_2 \ge \|x\|_2 - \|J_F x\|_2 \ge 1 - \rho,$$

while the triangle inequality gives

$$\|(I + J_F)x\|_2 \le 1 + \rho.$$

Taking the infimum and supremum over all unit vectors yields the stated singular-value bounds. □

Proposition A.1 is stated with respect to the projected state U_l^v. Under the local bound on the residual branch, the Jacobian of the complete stage remains bounded away from singularity. This provides a local explanation for the improved depth stability observed in Section 4.3.

At the self-attention sublayer of one head, the update associated with the global token is a data-dependent weighted sum in the learned value space. Writing the input tokens as $T_0 = C_l$ and $T_v = Z_l^v + E_l^v$ for $v = 1, \dots, V$, and denoting the pre-normalized tokens by $\bar{T}_j$, the attention contribution can be written as

$$O_l = \sum_{j=0}^{V} \alpha_{l,j} W_{\text{val}} \bar{T}_j, \qquad \alpha_{l,j} \ge 0, \qquad \sum_{j=0}^{V} \alpha_{l,j} = 1. \tag{A.2}$$

The complete Transformer additionally contains multiple heads, output projection, residual updates, a feed-forward block, and final normalization. The expression above shows that the global token receives a sample-dependent combination of the current view tokens in the learned value space.

Proposition A.2 (When gated cross-view guidance is beneficial)**.** *Let s denote a target semantic representation, let the unguided local state be $H = s + e$, and let $C = R(G)$ be a projected cross-view guidance vector. Consider the scalar-gated update*

$$Z(a) = H + aC, \qquad a \in [0, 1]. \tag{A.3}$$

Define

$$E = \mathbb{E}\|e\|_2^2, \qquad Q = \mathbb{E}\|C\|_2^2 > 0, \qquad c = \mathbb{E}[e^\top C].$$

Then

$$\mathcal{R}(a) = \mathbb{E}\|Z(a) - s\|_2^2 = E + 2ac + a^2 Q. \tag{A.4}$$

If $c < 0$, the minimizer over $[0, 1]$ is

$$a^* = \min\left\{1, -\frac{c}{Q}\right\}, \tag{A.5}$$

and $\mathcal{R}(a^*) < \mathcal{R}(0) = E$. If $c \ge 0$, the minimizer is $a^* = 0$, so suppressing the guidance recovers the unguided local state.

Proof. Expanding the squared norm gives the stated quadratic form. Since $Q > 0$, $\mathcal{R}(a)$ is convex with unconstrained minimizer $-c/Q$. When $c < 0$, projection onto $[0, 1]$ gives the stated solution and yields $\mathcal{R}(a^*) < E$. When $c \ge 0$, $\mathcal{R}'(0) \ge 0$ and convexity makes $a^* = 0$ optimal on $[0, 1]$. □

Algorithm B.1 Training and inference of HRG-MVC

Require: Multi-view data $\mathcal{X} = \{X^v\}_{v=1}^V$, number of clusters K, number of levels L, loss weights λ_1, λ_2, pretraining epochs T_{pre}, joint-training epochs T_{joint}
Ensure: Cluster labels Y

```
Initialize view-specific encoders and decoders, Cross-View Transformers, guidance projections, gates, and cluster heads
for t = 1 to T_pre do
    Sample a mini-batch {X_b^v}_{v=1}^V
    for l = 1 to L do
        Compute U_l^v = P_l^v(Z_{l-1}^v) and Z_l^v = U_l^v + F_l^v(U_l^v) for every view v
    end for
    Reconstruct X̂_b^v = D^v(Z_L^v) and update the autoencoders using L_rec
end for
for t = 1 to T_joint do
    Sample a mini-batch {X_b^v}_{v=1}^V
    Compute Z_1^v = P_1^v(X_b^v) + F_1^v(P_1^v(X_b^v)) for all views and obtain G_1 using Eq. 4
    for l = 2 to L do
        for v = 1 to V do
            Compute U_l^v and H_l^v using Eq. 2
            Project G_{l-1}, compute A_l^v, and obtain Z_l^v using Eqs. 5–6
        end for
        Obtain the level-wise consensus G_l using Eq. 4
    end for
    Decode {Z_L^v}_{v=1}^V and compute L_rec, L_fcl, and L_lcl
    Update all trainable parameters by minimizing Eq. 7
end for
Compute {S_l}_{l=1}^L on the complete dataset and assign labels using Eq. 16
return Y
```

The sign of c determines whether the guidance is corrective on average. The learned feature-wise sigmoid gate in HRG-MVC generalizes this scalar model by assigning a separate data-dependent weight to each feature. The proposition characterizes when such a correction can be useful; it does not claim that optimization always recovers the risk-minimizing gate.

B Training Algorithm

Algorithm B.1 summarizes reconstruction pretraining, joint optimization, and final inference.

C Detailed Computational Complexity

Let B be the mini-batch size, d_l the representation dimension at level l, $d_0^v = d_v$ the input dimension of view v, and f_l the Transformer feed-forward width at level l. The local residual encoder has forward cost

$$\mathcal{O}\left(B \sum_{l=1}^{L} \sum_{v=1}^{V} \left(d_{l-1}^v d_l + d_l^2\right)\right), \tag{C.6}$$

ignoring element-wise operations. The decoder has the same order as a standard view-specific autoencoder with reversed widths.

Because self-attention is applied only to the V views of the same sample and one global token, the Transformer cost is

$$\mathcal{O}\left(B \sum_{l=1}^{L} \left[(V+1)d_l^2 + (V+1)^2 d_l + (V+1) d_l f_l\right]\right). \tag{C.7}$$

Thus, the attention term is quadratic in V rather than N, and no sample-wise N^2 affinity matrix is introduced.

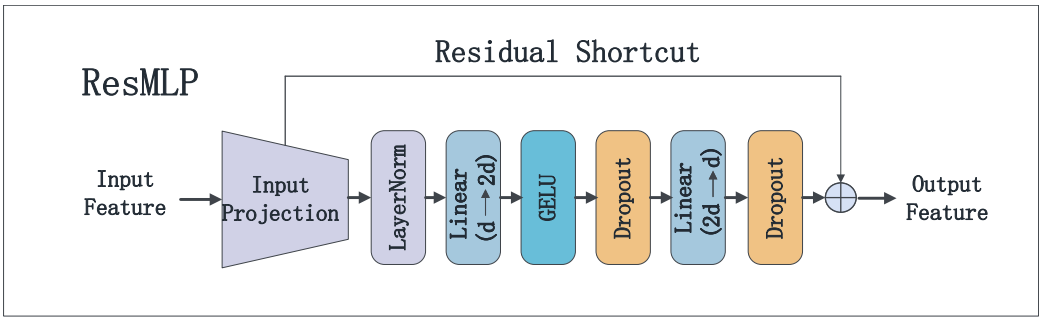

Figure D.1: Internal structure of a ResMLP stage. The projected stage input forms the primary local carrier, while the nonlinear MLP branch learns an incremental within-view update. For $l \geq 2$, the preceding consensus is projected and admitted through a feature-wise gate before being added as a second residual correction.

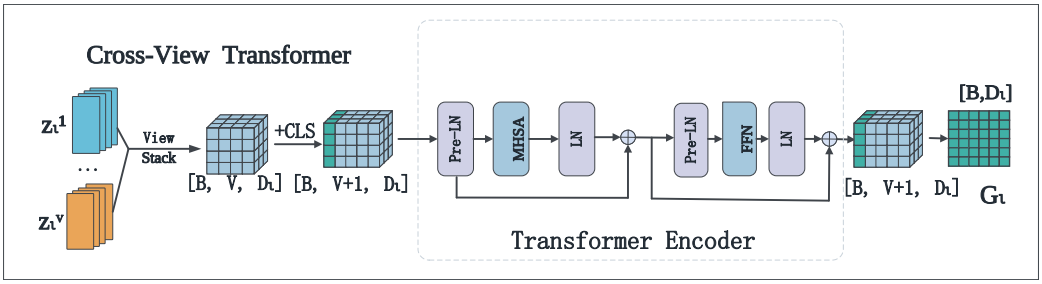

Figure D.2: Cross-View Transformer used as the level-wise consensus estimator. For one sample, the representations from all views are treated as tokens with learnable view embeddings. A learnable global token collects cross-view evidence through a single pre-norm Transformer encoder layer, and its normalized output defines G_l.

For $l \geq 2$, the guidance projection and feature-wise gate require

$$\mathcal{O}\left(B \sum_{l=2}^{L} \sum_{v=1}^{V} \left(g_{l-1} d_l + d_l^2\right)\right), \tag{C.8}$$

where g_{l-1} is the dimension of G_{l-1}. The feature contrast objective contributes

$$\mathcal{O}\left(B^2 \sum_{l=1}^{L} \binom{V}{2} d_l\right), \tag{C.9}$$

while inter-level assignment contrast costs

$$\mathcal{O}\left(BK^2 \binom{L}{2}\right). \tag{C.10}$$

These contrastive terms affect training only. For fixed V, L, and network widths, inference remains linear in the number of samples.

D Additional Architecture Details

Figure D.1 gives the exact ResMLP stage used by the implementation. Figure D.2 gives the corresponding Cross-View Transformer.

E Additional Experimental Analysis

The following results provide additional sensitivity and visualization evidence.

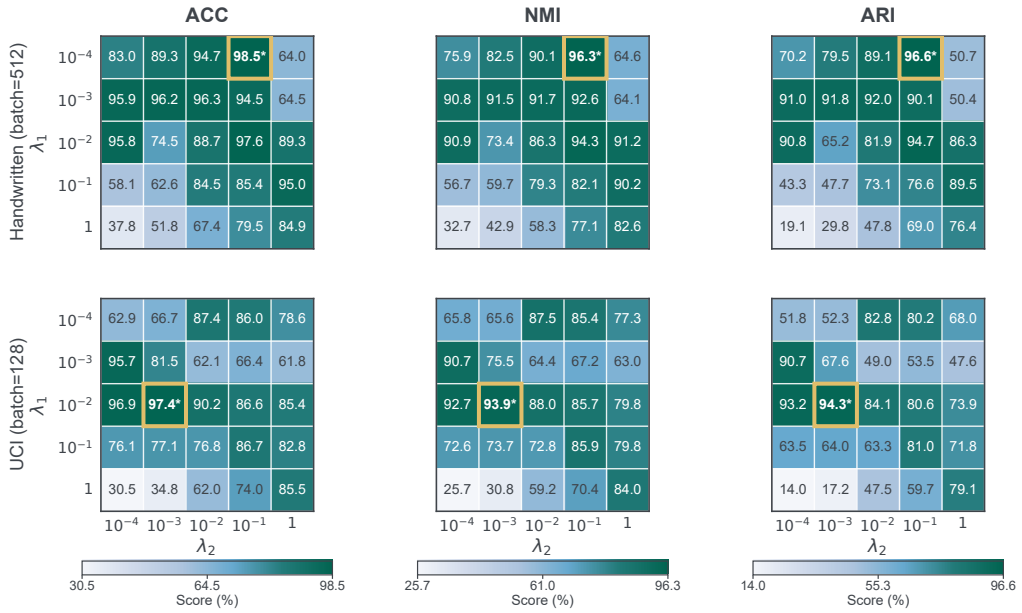

Figure E.3: Sensitivity to λ_1 and λ_2 on Handwritten and UCI-Digit. Columns show final-epoch ACC, NMI, and ARI (%). Boxes mark the selected settings.

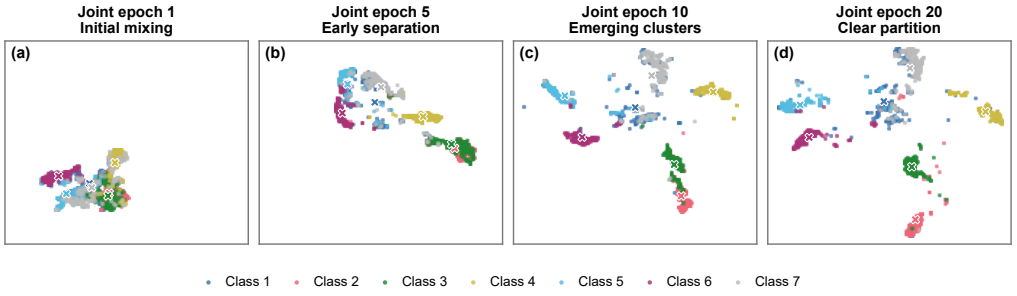

Figure E.4: Caltech-5V feature and clustering-score evolution from reconstruction pretraining to the end of joint training. Ground-truth labels are used only for coloring.

E.1 Parameter Sensitivity

We examine the effects of the two contrastive-loss weights, λ_1 and λ_2, on Handwritten and UCI-Digit. Figure E.3 shows that both datasets favor relatively small λ_1 values, while the preferred λ_2 differs between them. The selected settings lie within the tested grid and give strong results across ACC, NMI, and ARI.

E.2 Representation Visualization

Figure E.4 provides a qualitative t-SNE (van der Maaten & Hinton, 2008) view of representation evolution during joint optimization. The class structure becomes progressively clearer in the local feature space and is more pronounced in the clustering-score space toward the end of training.