Rethinking Multi-view Representation Learning via Distilled Disentangling – (ID: 6899) Supplementary Materials –

A. Observation Experiments

We utilized the Mutual Information Neural Estimator (MINE)[1] [3] as a mutual information estimator to independently assess the mutual information between view-consistent representations and view-specific representations proposed by CONAN[2] [11], DVIB[3] [2], Multi-VAE[4] [25], and our approach. To ensure a fair comparison, we standardized the representation dimensions of all comparative methods to 10. For constructing the MINE estimator, we employed fully connected layers with Rectified Linear Unit (ReLU) activation, specifying the network architecture as 20-100-100-100-1. We use Adam with the learning rate of 1×10^{-4} and the batch size of 128 to train the model for 500 epochs. To mitigate randomness, we executed the MINE procedure 10 times and recorded the average results.

B. Related Work

Multi-view Representation Learning. The goal of MvRL is to extract both shared and view-specific information from multiple data sources, integrating them into a cohesive representation that is advantageous for predictive tasks [5, 13, 16]. Existing approaches in this field generally fall into three categories: statistic-based, deep learning-based, and hybrid methods.

Statistic-based methods, employing techniques like canonical correlation analysis [6, 15], non-negative matrix factorization [14, 23], and subspace methods [4, 22], excel in deriving interpretable models. However, they struggle with datasets that are high-dimensional or large-scale. In contrast, deep learning-based methods have gained prominence, especially in unsupervised settings, where generative models such as autoencoders [1, 21, 27] and generative adversarial networks [29] are used to learn latent representations. Although effective, these methods face the challenge of redundancy when concatenating representations from all views, leading to suboptimal results for downstream tasks. Researchers have attempted to address this by exploring fusion methods for multi-view representations [11, 19, 24]. Nevertheless, deep learning-based methods often lack interpretability, being perceived as "black-box" approaches. Hybrid methods, such as those found in [10, 17, 28], combine statistical and deep learning approaches. They use deep learning for feature extraction and statistical learning for modeling interpretable representations. These methods effectively balance the strengths of both approaches but require substantial computational resources for post-processing.

Our approach is categorized under deep learning-based methods. We distinguish our work by utilizing deep learning's capacity to handle large datasets effectively. Moreover, we address the interpretability challenges in representations by incorporating disentanglement techniques.

C. Pseudo-code of MRDD

See Algorithm 1.

Algorithm 1 The pseudo-code of the proposed method.

Input: $\mathcal{X} = \{x^{(1)}, x^{(2)}, \cdots, x^{(v)} | x^{(i)} \in \mathbb{R}^{n \times d_v}\}$, the consistent encoder E_c, view-specific encoders and decoders $\{E_s^{(i)}\}_{i=1}^v$, $\{D_s^{(i)}\}_{i=1}^v$

Output: the view-consistent representation c, and view-specific representations $\{s^{(i)}\}_{i=1}^v$

masked inputs $\{x^{(i)}\}_{i=1}^v \rightarrow \{\hat{x}^{(i)}\}_{i=1}^v$.

$c \leftarrow$ concatenating all of $E_c(\hat{x})$'s outputs.

computing the consistent loss $\mathcal{L}_c$ using Eq.(3) .

fixed the the consistent encoder E_c.

repeat

  $\{s^{(i)}\}_{i=1}^V \leftarrow E_s^{(i)}(\{x^{(i)}\}_{i=1}^v)$, and $c \leftarrow E_c(\{x^{(i)}\}_{i=1}^v)$

  computing the disentangling loss $\mathcal{L}_d^i$ using Eq. (5).

  computing the reconstruction loss $\mathcal{L}_r^i$ using Eq. (6).

until $\mathcal{L}_s$ convergence.

[1] Code is accessible at: https://github.com/gtegner/mine-pytorch/

[2] Code is accessible at: https://github.com/Guanzhou-Ke/conan

[3] Code is accessible at: https://github.com/feng-bao-ucsf/DVIB

[4] Code is accessible at: https://github.com/SubmissionsIn/Multi-VAE

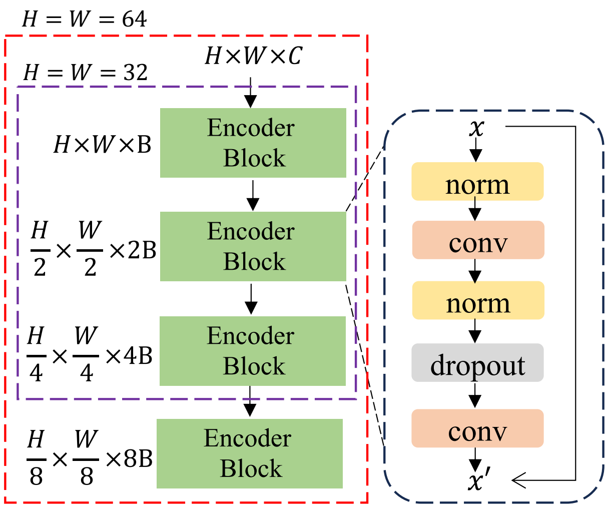


Figure 1. Illustration of encoder, where H, W, and C denote the height, width, and channels of an image, respectively. B denotes the number of output channels.

D. Network Structures

We employed convolutional neural networks to construct both the encoder and decoder components in our approach, ensuring a symmetric structure for both. As depicted in Fig. 1, an encoder block comprises two convolutional layers, two batch normalization layers, and a dropout module. These encoder blocks are then stacked to form the complete encoder. In the decoder architecture, the `Conv` module is substituted with the `ConvTranspose2d` module in PyTorch.

The output channel base, denoted as B, is set at 16 by default. To maintain consistent latent representations, we devised two distinct architectures tailored to different data dimensions. For data with a dimension of 32, only the first three layers from Figure 1 are utilized in both the encoder and decoder structures. Conversely, for data with a dimension of 64, we incorporate four blocks to constitute the encoder and decoder structures, ensuring the output dimension is normalized to 8×8. This approach is crucial for maintaining coherence across varying data dimensions.

E. Evaluation Metrics

To evaluate the performance of clustering, we apply three well-known metrics to the comparative experiments, including clustering accuracy (ACC$_{clu}$) and normalized mutual information (NMI). Given sample $x_j \in \mathbf{x}^i$ for any $j \in \{1, 2, \cdots, n\}$, the predicated clustering label and the real label are indicated as y_j and c_j, respectively. The ACC$_{clu}$ is defined as:

$$ACC_{clu} = \frac{\sum_{i=1}^{N} \delta(y_j, map(c_j))}{N} \tag{1}$$

where $y_j \in \mathbf{Y}$ represents ground-truth labels and $c_j \in \mathbf{C}$ denotes predicted clustering labels which generated by kmeans; $\delta(a, b)$ is the indicator function, i.e., $\delta(a, b) = 1$ if $a = b$, and $\delta(a, b) = 0$ otherwise; $map(\cdot)$ is the mapping function corresponding to the best one-to-one assignment of clusters to labels implemented by the Hungarian algorithm [12]; Then NMI is computed by:

$$NMI = \frac{I(\mathbf{Y}; \mathbf{C})}{\frac{1}{2}(H(\mathbf{Y}) + H(\mathbf{C}))} \tag{2}$$

$I(\cdot; \cdot)$ and $H(\cdot)$ represent mutual information and entropy functionals, respectively.

As for the classification task, we compute classification accuracy (ACC$_{cls}$) and F-score to report classification results, as shown below.

$$Fscore = \frac{2 \times P \times R}{P + R} \tag{3}$$

where $P = \frac{TP}{TP+FP}$; TP and FP are the number of true positives and the number of false positives, respectively; $R = \frac{TP}{TP+FN}$, where FN is the number of false negatives. Higher values of all of the aforementioned metrics indicate better performance.

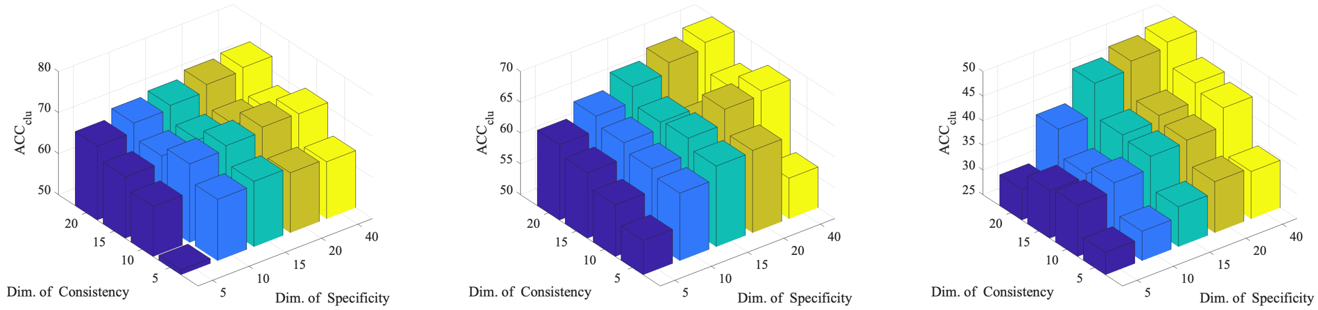
F. Classification Results

We evaluated the performance of all baseline models through classification tasks on the E-FMNIST and COIL-20 datasets, as summarized in Table 1. The results illuminate that, within the same experimental framework, the representations extracted by our method significantly enhance classification performance. Notably, in comparison to the second-best method, UNITER, our MRDD-cs approach demonstrated improvements of 4.59 and 4.58 in terms of Accuracy (ACC$_{cls}$) and F-score on the E-FMNIST dataset, respectively. These outcomes underscore that minimizing redundancy between view-consistent and view-specific representations proves advantageous in augmenting the effectiveness of downstream tasks.

G. Ablation Study

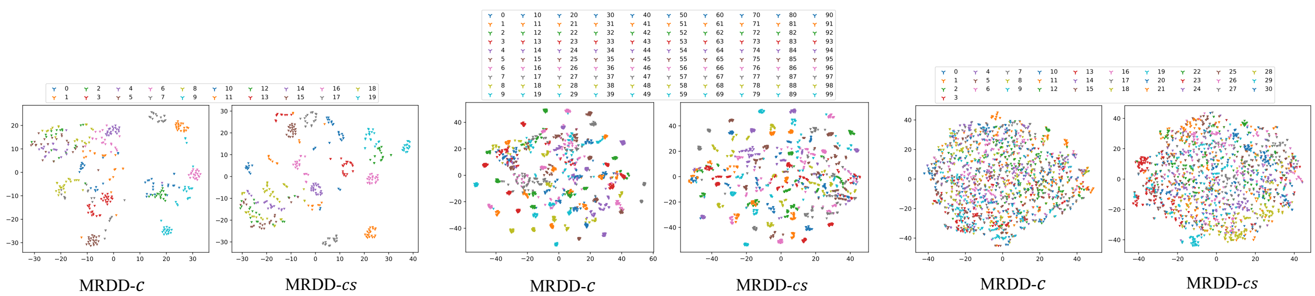
G.1. The dimension of consistency and specificity

We investigate the impact of view-consistent and view-specific representations extracted by our method across various dimensions. The view-consistent representation dimensions are set within the range $5, 10, 15, 20$, while the view-specific representation dimension spans $5, 10, 15, 20, 40$. As illustrated in Fig. 2, the results show a positive correlation with the view-specific representation dimension when the view-consistent representation dimension is held constant. Specifically, when the dimensions of view-consistent representations are fixed at 20, a noticeable



(a) COIL-20 (b) COIL-100 (c) Office-31

Figure 2. The clustering results (%) of the different dimensions of consistency and specificity on the COIL-20, COIL-100, and Office-31 datasets. The x-axis represents the consistency dimension, the y-axis represents the specificity dimension, and the z-axis represents the clustering accuracy.



(a) COIL-20 (b) COIL-100 (c) Office-31

Figure 3. Visualization of the representations of MRDD-**c** and MRDD-**cs** using t-SNE [20] on the COIL-20, COIL-100, and Office-31.

	E-FMNIST		COIL-20	
Method	ACC_{cls}	F-Score	ACC_{cls}	F-Score
Random	9.99±0.13	9.99±0.13	4.60±0.67	3.11±0.46
Joint-VAE[7]	56.50±0.23	56.39±0.21	87.76 ±2.00	84.24 ±3.16
β-VAE [9]	56.04±0.42	55.99±0.41	51.21±1.69	49.81±1.41
CONAN† [11]	58.13±0.21	55.74±0.15	67.53±2.72	61.54±2.82
CMC† [18]	67.43±0.13	64.85±0.17	89.16±0.01	89.15±0.01
Multi-VAE [25]	81.54±0.38	79.43±0.24	90.39±1.12	89.32±1.53
MIB [8]	75.33±0.05	73.80±0.05	59.72±2.29	53.99±2.03
DVIB [2]	72.18±0.29	72.91±0.22	44.31±3.30	42.17±3.02
UNITER [26]	<u>84.19±0.11</u>	<u>84.10±0.11</u>	<u>91.27±0.94</u>	<u>90.58±1.01</u>
MRDD-c (Ours)	82.51 ±0.30	82.28±0.29	88.18±0.96	87.57±0.82
MRDD-cs (Ours)	**88.78±0.22**	**88.68±0.18**	**95.97±0.56**	**96.15±0.88**
Δ SOTA	+4.59	+4.58	+4.7	5.57

Table 1. **Classification results (%) on E-FMNIST and COIL-20 datasets. Bold** denotes the best results and <u>underline</u> denotes the second-best. † denotes we set the dimensionality of latent representations as 10. All results are reproduced using the official released code.

incremental relationship is observed between the dimensions of view-specific representations and clustering performance.

In contrast, when the dimensions of view-specific representations are fixed at 40, consistent representations do not exhibit a clear pattern of variation. We posit that the overall performance of our method is primarily influenced by the expressive capacity of view-consistent representations. Additionally, a marginal improvement in overall performance is noted when the dimensions of view-specific representations surpass those of view-consistent representations. This observation suggests a nuanced interplay between the dimensions of these representations and their impact on the performance of downstream tasks.

H. Visualization

We visualize the representations of MRDD-**c** and MRDD-**cs** on the COIL-20, COIL-100, and Office-31 dataset. Fig. 3 indicates that view-consistent representations can distinguish different samples at a coarse level. However, after incorporating view-specific representations, the discriminative ability of the representations is enhanced, especially evident in the COIL-20 and COIL-100 dataset.

On the other hand, we demonstrate the reconstruction sampling of the COIL-20 and Office-31 datasets. As depicted in Fig. 4 and 5, reconstructing using only consistent representations results in the outline information of objects, indicating that the model has learned shared information among views. Furthermore, when incorporating view-

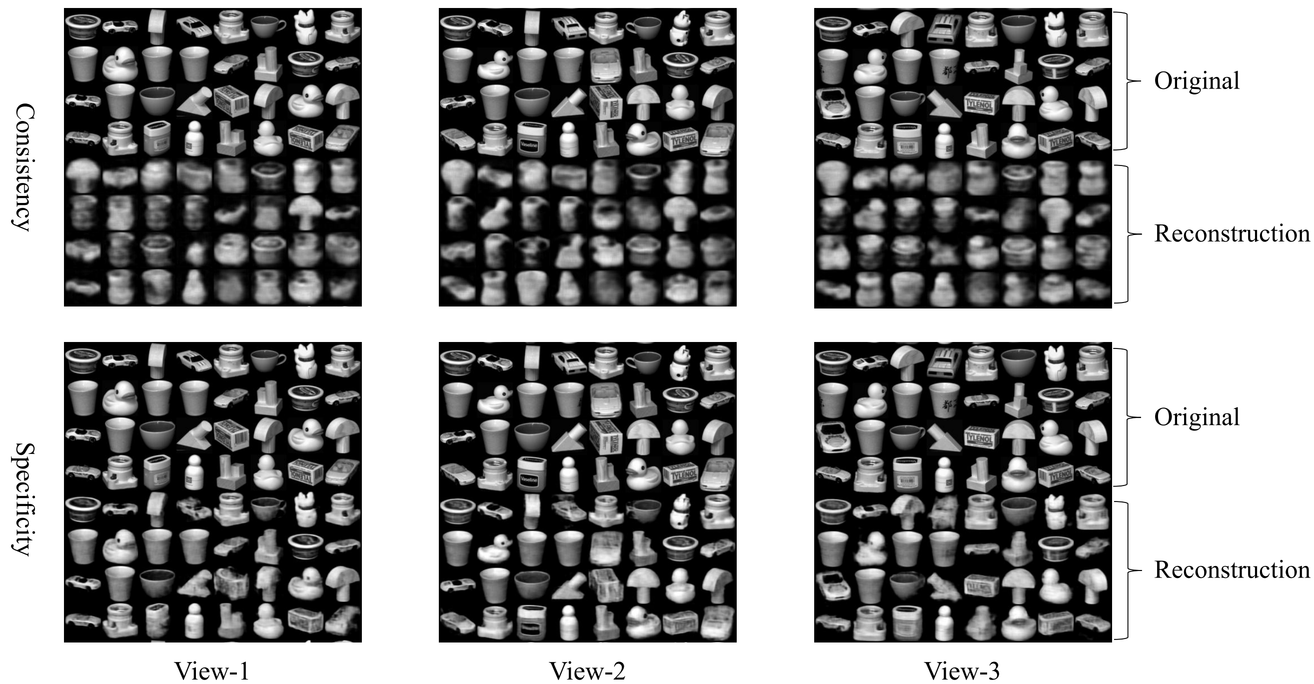


Figure 4. Visualization of reconstruction samples of consistency and specificity on the COIL-20 dataset.

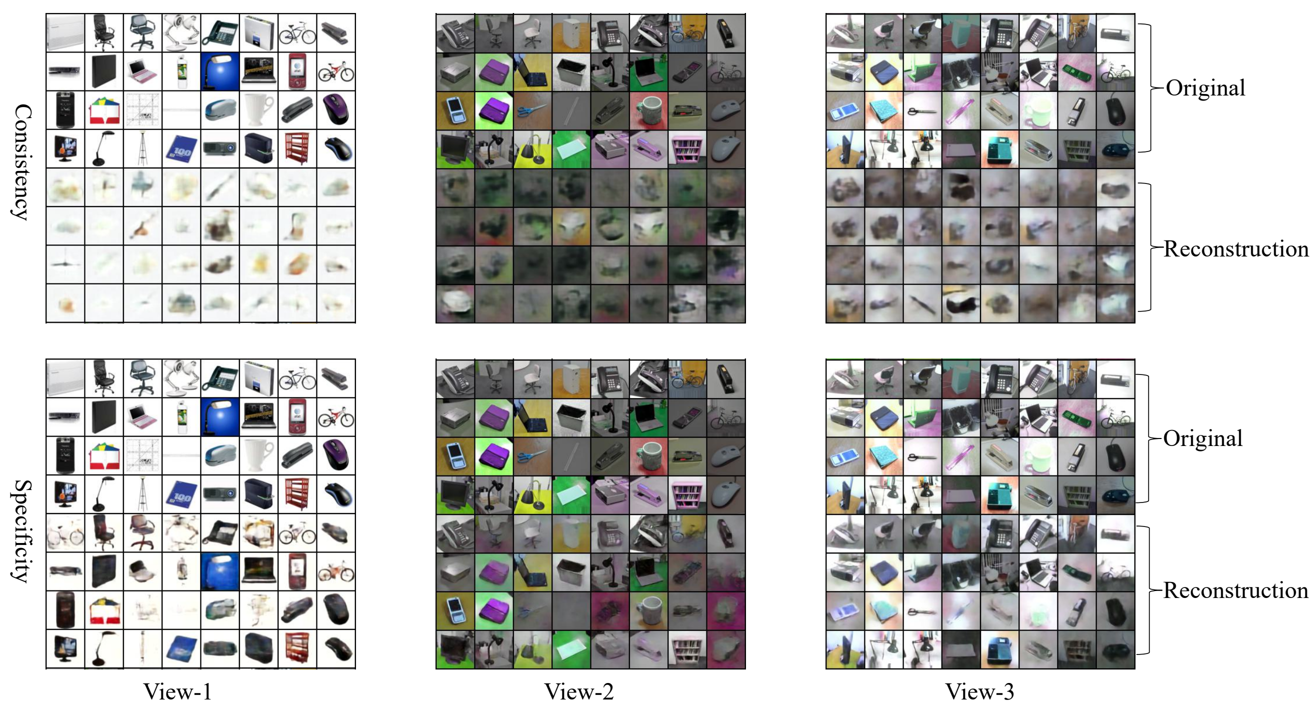


Figure 5. Visualization of reconstruction samples of consistency and specificity on the Office-31 dataset.

specific representations, a significant improvement in reconstruction quality is observed. This suggests that view-specific representations contain information such as textures, details, and other nuanced aspects of objects.

References

[1] Galen Andrew, Raman Arora, Jeff Bilmes, and Karen Livescu. Deep canonical correlation analysis. In *ICML*, pages 1247–1255. PMLR, 2013. 1

[2] Feng Bao. Disentangled variational information bottleneck

for multiview representation learning. In *CICAI*, pages 91–102. Springer, 2021. 1, 3

[3] Mohamed Ishmael Belghazi, Aristide Baratin, Sai Rajeshwar, Sherjil Ozair, Yoshua Bengio, Aaron Courville, and Devon Hjelm. Mutual information neural estimation. In *ICML*, pages 531–540. PMLR, 2018. 1

[4] Maria Brbić and Ivica Kopriva. Multi-view low-rank sparse subspace clustering. *Pattern Recognition*, 73:247–258, 2018. 1

[5] Guoqing Chao and Shiliang Sun. Consensus and complementarity based maximum entropy discrimination for multi-view classification. *Information Sciences*, 367:296–310, 2016. 1

[6] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *CVPR*, pages 886–893. IEEE, 2005. 1

[7] Emilien Dupont. Learning disentangled joint continuous and discrete representations. *NIPS*, 31, 2018. 3

[8] Marco Federici, Anjan Dutta, Patrick Forré, Nate Kushman, and Zeynep Akata. Learning robust representations via multi-view information bottleneck. *ICLR*, 2020. 3

[9] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. In *ICLR*, 2016. 3

[10] Zhenyu Huang, Joey Tianyi Zhou, Hongyuan Zhu, Changqing Zhang, Jiancheng Lv, and Xi Peng. Deep spectral representation learning from multi-view data. *TIP*, 30: 5352–5362, 2021. 1

[11] Guanzhou Ke, Zhiyong Hong, Zhiqiang Zeng, Zeyi Liu, Yangjie Sun, and Yannan Xie. Conan: contrastive fusion networks for multi-view clustering. In *Big Data*, pages 653–660. IEEE, 2021. 1, 3

[12] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955. 2

[13] Yingming Li, Ming Yang, and Zhongfei Zhang. A survey of multi-view representation learning. *TKDE*, 31(10):1863–1883, 2019. 1

[14] Jialu Liu, Chi Wang, Jing Gao, and Jiawei Han. Multi-view clustering via joint nonnegative matrix factorization. In *ICDM*, pages 252–260. SIAM, 2013. 1

[15] Nikhil Rasiwasia, Jose Costa Pereira, Emanuele Coviello, Gabriel Doyle, Gert RG Lanckriet, Roger Levy, and Nuno Vasconcelos. A new approach to cross-modal multimedia retrieval. In *ACMMM*, pages 251–260, 2010. 1

[16] Shiliang Sun and Guoqing Chao. Multi-view maximum entropy discrimination. In *IJCAI*, pages 1706–1712, 2013. 1

[17] Xiukun Sun, Miaomiao Cheng, Chen Min, and Liping Jing. Self-supervised deep multi-view subspace clustering. In *ACML*, pages 1001–1016. PMLR, 2019. 1

[18] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. In *ECCV*, pages 776–794. Springer, 2020. 3

[19] Daniel J Trosten, Sigurd Lokse, Robert Jenssen, and Michael Kampffmeyer. Reconsidering representation alignment for multi-view clustering. In *CVPR*, pages 1255–1265, 2021. 1

[20] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9 (11), 2008. 3

[21] Weiran Wang, Raman Arora, Karen Livescu, and Jeff Bilmes. On deep multi-view representation learning. In *ICML*, pages 1083–1092. PMLR, 2015. 1

[22] Xiaobo Wang, Zhen Lei, Xiaojie Guo, Changqing Zhang, Hailin Shi, and Stan Z Li. Multi-view subspace clustering with intactness-aware similarity. *Pattern Recognition*, 88: 50–63, 2019. 1

[23] Yang Wang, Lin Wu, Xuemin Lin, and Junbin Gao. Multi-view spectral clustering via structured low-rank matrix factorization. *TNNLS*, 29(10):4833–4843, 2018. 1

[24] Cai Xu, Wei Zhao, Jinglong Zhao, Ziyu Guan, Yaming Yang, Long Chen, and Xiangyu Song. Progressive deep multi-view comprehensive representation learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 10557–10565, 2023. 1

[25] Jie Xu, Yazhou Ren, Huayi Tang, Xiaorong Pu, Xiaofeng Zhu, Ming Zeng, and Lifang He. Multi-vae: Learning disentangled view-common and view-peculiar visual representations for multi-view clustering. In *CVPR*, pages 9234–9243, 2021. 1, 3

[26] Jie Xu, Yazhou Ren, Xiaoshuang Shi, Heng Tao Shen, and Xiaofeng Zhu. Untie: Clustering analysis with disentanglement in multi-view information fusion. *Information Fusion*, 100:101937, 2023. 3

[27] Changqing Zhang, Yeqing Liu, and Huazhu Fu. Ae2-nets: Autoencoder in autoencoder networks. In *CVPR*, pages 2577–2585, 2019. 1

[28] Handong Zhao, Zhengming Ding, and Yun Fu. Multi-view clustering via deep matrix factorization. In *AAAI*, 2017. 1

[29] Runwu Zhou and Yi-Dong Shen. End-to-end adversarial-attention network for multi-modal clustering. In *CVPR*, pages 14619–14628, 2020. 1