Robust Multi-view Clustering with Incomplete Information

Mouxing Yang, Yunfan Li, Peng Hu, Jinfeng Bai, Jiancheng Lv, Xi Peng

Abstract—The success of existing multi-view clustering methods heavily relies on the assumption of view consistency and instance completeness, referred to as the complete information. However, these two assumptions would be inevitably violated in data collection and transmission, thus leading to the so-called Partially View-unaligned Problem (PVP) and Partially Sample-missing Problem (PSP). To overcome such incomplete information challenges, we propose a novel method, termed robuSt mUlti-view clusteRing with incomplEte information (SURE), which solves PVP and PSP under a unified framework. In brief, SURE is a novel contrastive learning paradigm which uses the available pairs as positives and randomly chooses some cross-view samples as negatives. To reduce the influence of the false negatives caused by random sampling, SURE is with a noise-robust contrastive loss that theoretically and empirically mitigates or even eliminates the influence of the false negatives. To the best of our knowledge, this could be the first successful attempt that simultaneously handles PVP and PSP using a unified solution. In addition, this could be one of the first studies on the noisy correspondence problem (*i.e.*, the false negatives) which is a novel paradigm of noisy labels. Extensive experiments demonstrate the effectiveness and efficiency of SURE comparing with 10 state-of-the-art approaches on the multi-view clustering task.

Index Terms—Unsupervised Multi-view Representation Learning, Multi-view Clustering, Partially View-unaligned Problem, Partially Sample-missing Problem, False Negatives.

1 Introduction

Multi-view clustering (MvC) [1]–[4] aims at learning a common representation for multi-view data and then employing clustering on the representation. The success of MvC relies on the assumption of information completeness (Fig. 1(a)) which is two-fold: i) view consistency: it assumes that samples of the same instance are well-aligned. Taking two view matrices $\mathbf{X}^{(1)}$ and $\mathbf{X}^{(2)}$ as a showcase, it refers to that $\mathbf{X}^{(1)}$ and $\mathbf{X}^{(2)}$ have the corrected correspondence in row-wise, where each row denotes a sample; ii) instance completeness: it assumes that all instances are existing in all views, namely, $\mathbf{X}^{(1)}$ and $\mathbf{X}^{(2)}$ are with the same number of rows. In practice, however, either of the two assumptions would be violated in data collection and transmission, thus leading to the so-called Partially View-unaligned Problem (PVP, see Fig. 1(b)) and Partially Sample-missing Problem (PSP, see Fig. 1(c)).

During past years, some efforts have been devoted to solving PSP by imputing the missing samples with various data recovery methods [5]–[7]. In other words, these methods recover the missing samples by utilizing the information contained in the existing cross-view counterparts. Different from PSP, PVP is a less-touched problem revealed in very recent [8]. A feasible solution to PVP is first realigning the data using the Hungarian algorithm [9] and then achieving MvC based on the realigned data. However, such a two-stage approach cannot lead to encouraging performance

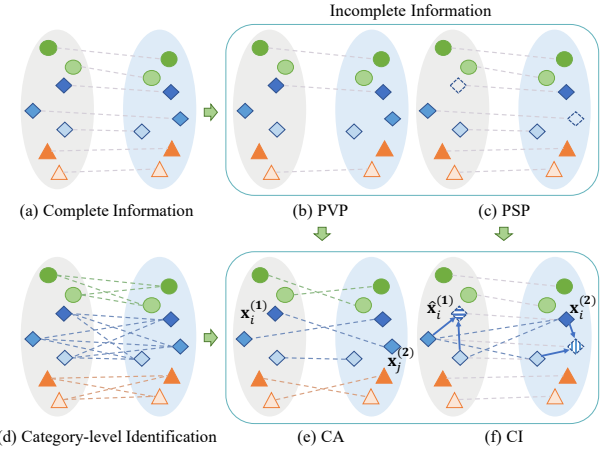


Fig. 1. Our basic idea. Taking a bi-view data as a showcase, we use two oval panels to denote two views, polygons with different colors and shapes to indicate different instances and categories. The grey dotted lines indicate that the cross-view correspondences are available. **(a)** Complete Information; **(b)** PVP: some of the cross-view correspondences are unavailable. **(c)** PSP: some samples are missing (denoted by hollow polygons); **(d)** Category-level Identification: establish the cross-view correspondences at category level by identifying cross-view and within-category counterparts for each sample, where the desirable correspondences are denoted by colored dotted lines; **(e)** Category-level Alignment (CA): solve PVP by realigning each sample $\mathbf{x}_i^{(1)}$ with its counterpart $\mathbf{x}_j^{(2)}$; **(f)** Category-level Imputation (CI): recovers each missing sample $\hat{\mathbf{x}}_i^{(1)}$ by using k counterparts of $\mathbf{x}_i^{(2)}$. One could observe that both CA and CI aim at identifying the within-category samples from different views. The only difference between them is that CA aims to identify one counterpart while CI aims to seek multiple ones. In other words, PVP and PSP could be solved by CA and CI which are unified into the same category-level identification framework.

- *This work was supported in part by the National Key R&D Program of China under Grant 2020AAA0104500; in part by NSFC under Grant U21B2040, 62176171, 61836006, U19A2078; in part by the Fund of Sichuan University Tomorrow Advancing Life; in part by Open Research Projects of Zhejiang Lab under Grant 2021KH0AB02.*
- *M. Yang, Y. Li, P. Hu, J. Lv, and X. Peng are with College of Computer Science, Sichuan University, Chengdu, 610065, China.*
  E-mail: {yangmouxing, yunfanli.gm, penghu.ml, pengx.gm}@gmail.com, lvjiancheng@scu.edu.cn
- *J. Bai is with TAL AI Lab, Beijing, China.*
  E-mail: baijinfeng1@tal.com.

Corresponding author: X. Peng.

as pointed out in [8]. Hence, [8] reformulates the Hungarian algorithm as a neural module so that instance-level data alignment and representation learning could be simultaneously performed. Although some promising results have been achieved by these studies, almost all of them can only solve either PSP or PVP, and it is unknown how to simultaneously conquer them using a unified framework.

In this paper, we observe that the solutions to PVP and PSP could be unified into a category-level identification framework. As shown in Fig. 1(d), for each sample, the framework aims at identifying its cross-view counterparts of the same category, *i.e.*, establishing the cross-view correspondences at category level. Clearly, it is natural to solve PVP in such a process, and PSP could also be solved by further exploiting the correspondence. For clarity, we refer to these two solutions as *Category-level Alignment* (CA) and *Category-level Imputation* (CI). The only difference between them is that CA aims at identifying one counterpart whereas CI aims at identifying multiple ones. With the established correspondence, CA solves PVP by directly realigning the sample $\mathbf{x}_i^{(1)}$ to its counterpart $\mathbf{x}_j^{(2)}$ as shown in Fig. 1(e); Likewise, CI conquers PSP by recovering the missing sample $\hat{\mathbf{x}}_i^{(1)}$ using k counterparts of $\mathbf{x}_i^{(2)}$ as shown in Fig. 1(f).

Based on the above observations, we propose a novel method which implements category-level identification to conquer PVP and PSP. In brief, the proposed *robuSt mUlti-view clusteRing with incomplEte information* (SURE) aims to learn categorical similarities and establish correspondences across views by resorting to a novel noise-robust contrastive learning paradigm. In detail, SURE treats the samples with complete information, *i.e.*, the aligned and observed samples, as positive pairs. As the category annotation is unavailable, we construct negative pairs by randomly choosing some samples across views. Clearly, such a pair construction approach would wrongly treat some within-cateogory samples as negatives, which results in false-negative pairs (FNPs). If such an issue is neglected, models will converge to the sub-optimal even wrong solutions. To mitigate or even eliminate the influence of FNPs, we propose a novel noise-robust contrastive loss which theoretically and experimentally enjoys the property of reversed and slow optimization (see Theorem 2 and 3 in Section 3.2). The contributions and novelties of this work could be summarized as follows:

- We propose treating MvC with incomplete information as a category-level identification task. To the best of our knowledge, although some developments have been achieved in either PVP or PSP, there is no a unified framework to simultaneously conquer both of them.
- To implement the category-level identification, we propose a novel noise-robust contrastive loss that could mitigate or even eliminate the influence of FNPs introduced during the pair construction.
- As far as we know, this could be one of the first attempts which enable contrastive learning robust against noisy correspondence, *i.e.*, FNPs. Notably, the standard noisy labels refer to as incorrect class annotation of a given sample, whereas our noisy correspondence denotes incorrect correspondence between two samples. Hence, this work might also provide some novel insights to the community of learning with noisy labels.

2 Related Works

In this section, we briefly review three topics related to this work, *i.e.*, multi-view clustering, contrastive learning, and learning with noisy labels. Besides, we elaborate on the differences between our prior work [10] and this study.

2.1 Multi-view Clustering

Almost all existing MvC methods implicitly or explicitly take the complete information assumption. However, once the assumption is violated, the view correspondence and instance completeness will be destroyed, thus leading to PVP and PSP. Based on the robustness against PVP and PSP, most of existing works could be roughly classified into the following three categories, namely, i) the vanilla MvC methods [1], [2], [4], [11]–[14], which strive to learn discriminative representations by utilizing the consistent and complementary information from different views; ii) PVP-oriented MvC methods [8], which aim at establishing the cross-view correspondence at the instance level in a unsupervised manner; and iii) PSP-oriented MvC methods [5]–[7], [15]–[17], which utilize the existing views to recover the missing ones.

The differences between existing approaches and our SURE are two-fold. On the one hand, SURE could simultaneously handle PVP and PSP whereas the existing works could only cope with one or neither of them. On the other hand, SURE aims to achieve view alignment and data recovery at category- instead of instance-level. The category-level alignment enables SURE to enjoy higher accessibility and scalability for clustering as verified in the experiments. More specifically, for two cross-view samples, the alignment probabilities at the instance and category levels are $1/N$ and $1/K$ respectively, where N and K denote the number of instances and categories, and $K \ll N$ in general.

2.2 Contrastive Learning

Recently, the contrastive learning methods [18]–[24] have shown unprecedented power in unsupervised representation learning. The major differences of most existing studies mainly lie in the choice of data augmentation strategy and contrastive loss. To be specific, most contrastive learning methods first construct sample pairs at the instance level by employing a series of augmentations on the raw data. The augmented samples of the same instance are defined as positive, while the others are considered as negative. With the augmented data, a variety of losses [21]–[23], [25], [26] have been proposed to learn the representations by maximizing the similarities of positives while minimizing those of negatives.

The major differences between this work and the existing contrastive learning methods are given below. First, our SURE is equipped with a novel contrastive loss which is robust against false negatives, whereas most of these methods cannot handle this issue. Second, these methods cannot be applied to multi-view data with PVP and PSP, whereas SURE is specifically proposed for handling such an incomplete information case. Third, we construct data pairs using the available complete information, whereas these methods resort to various data augmentations.

2.3 Learning with Noisy Labels

In recent years, a series of works [27]–[31] have been conducted to endow neural networks with the robustness against noisy labels. Based on the way of achieving robustness, the existing works could be roughly grouped into four categories, namely,

i) robust loss based methods [30], [32], which design a loss function which is tolerant to noisy labels; ii) robust architecture based methods [33], [34], which modify the network architecture to simulate the label transition matrix; iii) sample re-weighting methods [35], [36], which iteratively compute the confidences of samples as clean and reweigh their importances to guide the network optimization; and iv) semi-supervised learning methods [27], [37], which first identify the clean samples from the noisy ones and then optimize the network by treating the clean samples as labeled and the noisy ones as unlabeled.

Different from the above studies, we consider a novel noisy label paradigm, *i.e.*, the correspondence rather than the category annotation is incorrect. In addition, SURE is proposed to cluster multi-view data, whereas almost all of these methods are proposed for classification.

2.4 Differences from the Preliminary Version

This study is a journal extension of the conference paper (MvCLN) [10] with the following differences and improvements. To be specific,

- The motivations are different. In detail, MvCLN takes PVP into consideration and proposes solving this problem by establishing category-level correspondeces with the help of the noise-robust contrastive loss. In contrast, SURE, which solves two incomplete information challenges including PVP and PSP, is more general than MvCLN. Note that, SURE is the first unified framework which could simultaneously handle PVP and PSP.
- The loss functions are different. To solve the PVP challenge, MvCLN proposes a noise-robust contrastive loss by enforcing the consistency across views. However, with the loss, MvCLN might overemphasize the consistency between views, thus leading to the insufficient view-specific information preserved in the representations. Such representations do not favor the data recovery, thus failing in tackling PSP. In contrast, SURE takes the information sufficiency into consideration by developing a sufficiency-preserving versatile learning loss. The addition loss not only endows SURE with the data recovering ability for solving PSP but also boosts the performance of handling PVP, as verified in our experiments.
- The model architectures are different. In brief, MvCLN adopts contrastive learning like feedforward network structure, whereas SURE adopts an auto-encoder-like structure.

3 Method

In this section, we introduce SURE, a robust multi-view clustering method that simultaneously solves PVP and PSP under a category-level identification framework. In Section 3.1, we first present the related formal formulations. In Section 3.2, we introduce the proposed noise-robust contrastive objective which could implement the category-level identification. In Section 3.3, we introduce another objective which is designed to preserve the sufficiency of the learned representations. Finally, Section 3.4 elaborates on the implementation details of SURE.

3.1 Problem Formulation

In this work, we aim to explore how to achieve robust multi-view clustering with incomplete information. Formally, we have the following formal definitions.

Definition 1. ***Incomplete Information***. *For a multi-view dataset* $\{\mathbf{X}^{(v)}\}_{v=1}^{V} = \{\mathbf{x}_1^{(v)}, \mathbf{x}_2^{(v)}, \ldots, \mathbf{x}_{N_x}^{(v)}\}_{v=1}^{V}$, *it consists of* $\{\mathbf{S}^{(v)}\}_{v=1}^{V} = \{\mathbf{s}_1^{(v)}, \mathbf{s}_2^{(v)}, \ldots, \mathbf{s}_{N_s}^{(v)}\}_{v=1}^{V}$ *and* $\{\mathbf{W}^{(v)}\}_{v=1}^{V} = \{\mathbf{w}_1^{(v)}, \mathbf{w}_2^{(v)}, \ldots, \mathbf{w}_{N_w}^{(v)}\}_{v=1}^{V}$, *where* $v \in [1, V]$ *denotes the view index,* V *is the view number,* $N_x = N_s + N_w$ *represents the number of instances, and* $\{\mathbf{S}^{(v)}\}_{v=1}^{V}$/$\{\mathbf{W}^{(v)}\}_{v=1}^{V}$ *denotes the data without/with either or both of PVP and PSP.*

Definition 2. ***Partially View-unaligned Problem (PVP)***. *The dataset* $\{\mathbf{X}^{(v)}\}_{v=1}^{V}$ *is partially view-unaligned when*

$$\sum_{v_1}^{V} \sum_{v_2 \neq v_1}^{V} I(\mathbf{w}_i^{(v_1)}, \mathbf{w}_i^{(v_2)}) < V(V-1), \ \forall i \in [1, N_w], \tag{1}$$

where $I(a, b)$ *is an indicator function evaluating to* 1 *i.f.f. samples* a *and* b *belong to the same instance.*

Definition 3. ***Partially Sample-missing Problem (PSP)***. *The dataset* $\{\mathbf{X}^{(v)}\}_{v=1}^{V}$ *is partially sample-missing when*

$$1 \leq |\{\mathbf{w}_i^{(v)}\}_{v=1}^{V}| < V, \ \forall i \in [1, N_w], \tag{2}$$

where $|\cdot|$ *refers to the number of non-missing samples.*

To explore the unified solution to PVP and PSP, we propose a category-level identification framework by establishing the cross-view correspondences. Formally,

Definition 4. ***Category-level Identification***. *For each sample* $\mathbf{x}_i^{(v_1)}$, *it aims at identifying its with-category counterparts* $\mathbf{x}_j^{(v_2)}$ *from another views so that*

$$\sum_{v_1}^{V} \sum_{v_2 \neq v_1}^{V} C(\mathbf{x}_i^{(v_1)}, \mathbf{x}_j^{(v_2)}) = KV(V-1), \tag{3}$$

where $C(a, b)$ *is an indicator function evaluating to 1 i.f.f.* a *and* b *belong to the same category, and* K *denotes the number of instances for each category.*

With the established category-level correspondences, the sample $\mathbf{x}_i^{(v_1)}$ could be realigned with their counterparts $\mathbf{x}_j^{(v_2)}$. Similarly, the missing sample $\mathbf{x}_i^{(v_1)}$ could be recovered by their k peers $\mathbf{x}_k^{(v_1)}$ in the same view which are identified through $\mathbf{x}_j^{(v_2)}$ by resorting the established correspondences. Therefore, both PVP and PSP could be solved through performing category-level identification. The details are presented in Section 3.4.

To establish the cross-view correspondences, one feasible solution is the supervised contrastive learning [38] which aims at maximizing similarities of within-category samples (positives) while minimizes those of between-category samples (negatives). However, such a paradigm relies on the category annotations for pair construction, which is infeasible under the clustering setting. To get rid of the dilemma, we propose SURE which is composed of three modules, namely, pair construction, noise-robust optimization, and versatile learning. For ease of representation, we take $V = 2$ in the following without loss of generality. As shown in Fig. 2, the pair construction module uses $(\mathbf{s}_i^{(1)}, \mathbf{s}_i^{(2)})$ as positive pairs, and stochastically selects cross-view samples to form negative pairs $(\mathbf{s}_i^{(1)}, \mathbf{s}_j^{(2)})$. As the random sampling would

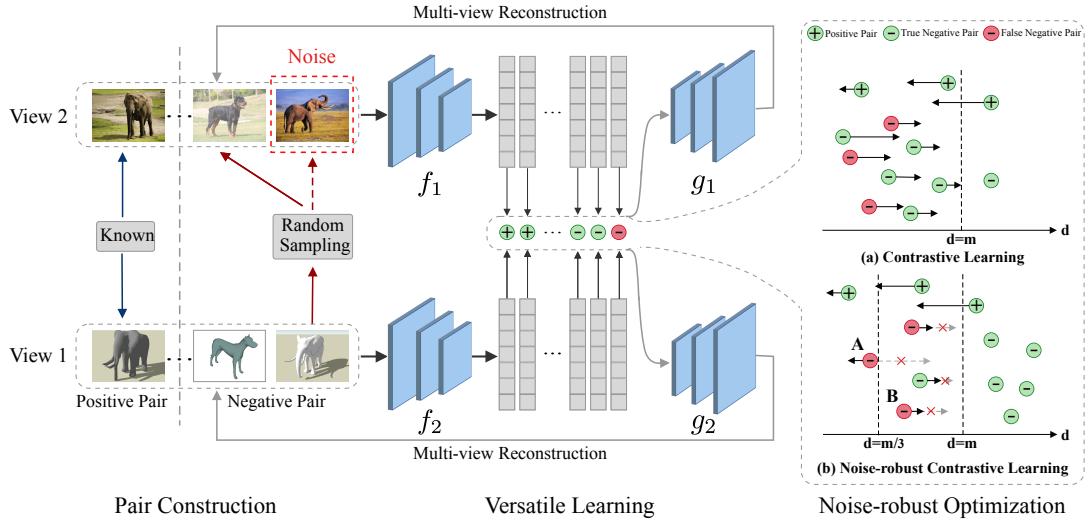


Fig. 2. The pipeline of the proposed SURE. It consists of three modules, *i.e.*, pair construction, noise-robust optimization, and versatile learning. For pair construction, SURE constructs positive pairs using the known correspondences, and forms negative pairs by random sampling on the fully-aligned and complete data $\{\mathbf{S}^{(v)}\}_{v=1}^{2}$. Such a random sampling strategy would inevitably introduce some false-negative pairs (FNPs) which should be treated as positive. To prevent these FNPs from dominating the network update, SURE adopts a two-stage optimization scheme. Specifically, (a) contrastive learning: the network is first warmed up with the vanilla contrastive loss until the mean distance of negatives is larger than the adaptive margin m. Then, SURE switches to (b) noise-robust contrastive learning: it will mitigate or even eliminate the influences of FNPs by reducing (see point B) or even reversing their gradient (see point A). In (a) and (b), the direction and length of the arrows refer to the direction and magnitude of the gradients, respectively. Moreover, to preserve the view-specific information, SURE imposes the versatile learning by reconstructing inputs using the common representation.

inevitably introduce noisy labels (*i.e.*, false-negative pairs), to mitigate or even eliminate the influence of these special noisy labels, we design a noise-robust optimization module which is equiped with a novel noise-robust contrastive loss $\mathcal{L}^{ncl}$. To maintain the sufficiency of the representations, SURE further adopts the versatile learning module with the versatile loss $\mathcal{L}^{ver}$ to reconstruct the input samples from the common representations. The overall loss function of SURE is

$$\mathcal{L} = \mathcal{L}^{ncl} + \lambda \mathcal{L}^{ver}, \tag{4}$$

where λ is a trade-off parameter which is fixed to 0.5 in our implementation.

3.2 Noise-robust Contrastive Learning

To mitigate or even eliminate the influence of false-negative pairs, we propose the following noise-robust contrastive loss, *i.e.*,

$$\mathcal{L}^{ncl} = \frac{1}{2N}\sum_{i=1}^{N}\left(Y\mathcal{L}_i^{pos} + (1-Y)\mathcal{L}_i^{neg}\right), \tag{5}$$

where N denotes the number of contrastive pairs, and $Y = 1/0$ for positive/negative pairs. Clearly, $\mathcal{L}_i^{pos}$ and $\mathcal{L}_i^{neg}$ contribute to positive and negative pairs, respectively.

Given a positive pair $(\mathbf{s}_i^{(1)}, \mathbf{s}_i^{(2)})$, SURE aims at minimizing its distance in the latent space by minimizing

$$\mathcal{L}_i^{pos} = d^2\left(\mathbf{s}_i^{(1)}, \mathbf{s}_i^{(2)}\right), \tag{6}$$

$$d\left(\mathbf{s}_i^{(1)}, \mathbf{s}_i^{(2)}\right) = \left\| f_1\left(\mathbf{s}_i^{(1)}\right) - f_2\left(\mathbf{s}_i^{(2)}\right)\right\|_2, \tag{7}$$

where f_1 and f_2 represent two view-specific neural networks for feature extraction.

As simply minimizing Eq. 6 would lead to a trivial solution (*i.e.*, all samples will collapse into a single point), the following contrastive term is often used to prevent model collapsion, namely,

$$\mathcal{L}_i^{ctr} = \max\left(m - d\left(\mathbf{s}_i^{(1)}, \mathbf{s}_j^{(2)}\right), 0\right)^2, \tag{8}$$

where m is a margin which enforces the distance of negatives to be moderately large. Integrating Eq. 6 and Eq. 8, we obtain the vanilla loss of SIAMESE network [18], *i.e.*,

$$\mathcal{L}^{van} = d^2\left(\mathbf{s}_i^{(1)}, \mathbf{s}_i^{(2)}\right) + \max\left(m - d\left(\mathbf{s}_i^{(1)}, \mathbf{s}_j^{(2)}\right), 0\right)^2. \tag{9}$$

As demonstrated in Fig. 3(a) and 3(b), the above vanilla loss fails to handle noisy labels since it would confuse the true- and false-negative pairs (*i.e.*, TNPs and FNPs), which ends up with the performance degradation. Therefore, to embrace the robustness against FNPs, we propose the following noise-robust contrastive term, namely,

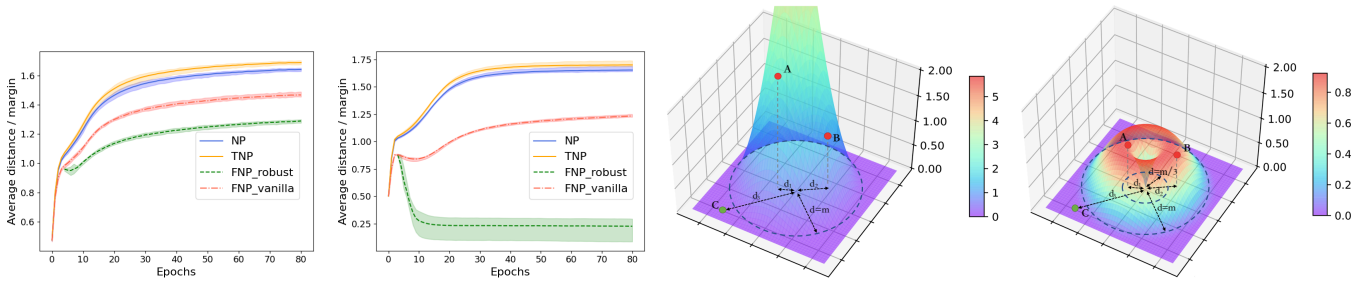
$$\mathcal{L}_i^{neg} = \frac{1}{m}\max\left(md^{\frac{1}{2}}\left(\mathbf{s}_i^{(1)}, \mathbf{s}_j^{(2)}\right) - d^{\frac{3}{2}}\left(\mathbf{s}_i^{(1)}, \mathbf{s}_j^{(2)}\right), 0\right)^2. \tag{10}$$

Considering the diverse data distribution, the optimal m may differ in different datasets. To avoid laboriously parameter selection, we propose adaptively computing m for each dataset at the initial state. Mathematically,

$$m = \frac{1}{N_p}\sum d\left(\mathbf{s}_i^{(1)}, \mathbf{s}_i^{(2)}\right) + \frac{1}{N_n}\sum d\left(\mathbf{s}_i^{(1)}, \mathbf{s}_j^{(2)}\right), \tag{11}$$

where N_p and N_n denote the number of positive and negative pairs, respectively. Notably, m is only computed once after the network initialization and will be fixed in the later training process.

In the following, we mathematically and empirically explain why the proposed noise-robust contrastive term (Eq. 10) could prevent the network from fitting FNPs or even revise the wrong optimization direction. To begin with, we plot the loss surface of



(a) (b) (c) (d)

Fig. 3. Empirical and mathematical analysis of the vanilla contrastive term (*i.e.*, Eq. 8) and the proposed noise-robust contrastive term (*i.e.*, Eq. 10). (a–b) The ratio of the average distance to margin *w.r.t.* training epochs on NoisyMNIST and Reuters, where NP, FNP_vanilla, and FNP_robust denote negative pairs, false-negative pairs optimized by Eq. 8, and false-negative pairs optimized by Eq. 10, respectively. The colored regions denote the standard variances under five different initializations. It shows that our loss could prevent the distance of false negatives from wrongly increasing or even reverse the optimization direction to correctly treat them as positive pairs as desired. (c–d) The loss surface of Eq. 8 and Eq. 10. We take points in all three possible cases (A, B, C) as showcases to manifest the robustness of our noise-robust term compared to the vanilla term. Specifically, A, B, and C refer to the false-negative pairs (FNPs) with the distance of $d_1 < m/3$ and $m/3 < d_2 < m$, and true negative pairs (TNPs) with the distance of $d_3 > m$, respectively. In panel (c), the vanilla loss in Eq. 8 monotonously increases the distance for all negative pairs A, B, C and holds no robustness against noisy labels. In contrast, panel (d) illustrates that our robust term in Eq. 10 could decrease the distance of point A while slowly increasing that of B, thus embracing the robustness against noisy labels.

Eq. 8 and Eq. 10 *w.r.t.* the distance of negative pairs in Fig. 3(c) and 3(d), respectively. One could observe that optimizing our noise-robust term (*i.e.*, Eq. 10) would not monotonically increase the distance of negative pairs, in contrast to the vanilla term (*i.e.*, Eq. 8). This observation could also be theoretically supported by the following theorem.

Theorem 1. *The gradient of our noise-robust term (i.e., Eq. 10) is nonmonotonic.*

Proof. *Let the distance of negative pairs be d, as Eq. 10 produces no gradient when $m > d$, we only need to consider the case when $m \leq d$. The gradient of $\mathcal{L}^{neg}$ w.r.t. d is in the form of*

$$\begin{aligned}\frac{\partial \mathcal{L}^{neg}}{\partial d} &= \frac{\partial\left(\frac{1}{m}d^3 - 2d^2 + md\right)}{\partial d} \\ &= \frac{3}{m}d^2 - 4d + m \\ &= \left(\frac{3d}{m} - 1\right)(d - m)\,, \end{aligned} \tag{12}$$

which equals to zero iff $d = m/3$ or $d = m$.

Based on Theorem 1, the loss surface of Eq. 10 could be divided into two areas, namely, the hole area ($0 < d < m/3$) and the deceleration area ($m/3 < d < m$) as shown in Fig. 3(d). Accordingly, we could further derive the following two theorems.

Theorem 2 (**Reversed Optimization**). *The gradient direction of our noise-robust term (i.e., Eq. 10) is reversed compared with the vanilla term in (i.e., Eq. 8) when $0 < d < m/3$.*

Proof. *The product of the gradients of $\mathcal{L}^{neg}$ and $\mathcal{L}^{ctr}$ w.r.t. d is*

$$\begin{aligned}\frac{\partial \mathcal{L}^{neg}}{\partial d}\frac{\partial \mathcal{L}^{ctr}}{\partial d} &= \left(\frac{3d}{m} - 1\right)(d - m)\,(2(d - m)) \\ &< 0,\ \forall d \in (0, m/3)\,. \end{aligned} \tag{13}$$

Based on Theorem 2, one could observe that the gradient of Eq. 10 is contrary to the one of Eq. 8. In other words, for the pairs locating into the hole area (such as point A in Fig. 3(d)), the gradient of Eq. 10 will be reversed compared to that of Eq. 8.

Theorem 3 (**Slow Optimization**). *The optimization of our noise-robust term (i.e., Eq. 10) is decelerated compared to the vanilla term in (i.e., Eq. 8) when $m/3 < d < m$.*

Proof. *Let Δ_d be the numerical difference between gradients of $\mathcal{L}^{neg}$ and $\mathcal{L}^{ctr}$ w.r.t. d, it could proved that $\Delta_d < 0$ when $m/3 < d < m$ by*

$$\begin{aligned}\Delta_d &= \left|\frac{\partial \mathcal{L}_i^{neg}}{\partial d}\right| - \left|\frac{\partial \mathcal{L}_i^{ctr}}{\partial d}\right| \\ &= \left|\left(\frac{3d}{m} - 1\right)(d - m)\right| - |2\,(d - m)| \\ &= -\frac{3}{m}(d - m)^2 < 0. \end{aligned} \tag{14}$$

Note that, since $\partial \mathcal{L}_i^{neg}/\partial d < 0$, $\partial \mathcal{L}_i^{ctr}/\partial d < 0\ \forall d \in (m/3, m)$, Eq. (14) uses their absolute value in calculation. It shows that for the pairs in the deceleration area (such as point B in Fig. 3(d)), their gradients will be reduced by Eq. 10 compared with Eq. 8.

According to Theorem 2 and 3, for any FNPs in $(0, m/3)$, the proposed SURE could correctly decrease their distance by reversing their gradient. As for any FNPs in $(m/3, m)$, SURE could decrease the unwanted distance increment, thus preventing the network from fitting FNPs.

Notably, although our noise-robust term (*i.e.*, Eq. 10) endows SURE with the robustness to FNPs, it may hinder the network from fitting TNPs as well. To reconcile the contradiction between the robustness to FNPs and the optimization of TNPs, we adopt a two-stage optimization scheme motivated by [39]. Specifically, Bengio *et al.* [39] empirically observes that the neural networks apt to fit simple patterns first, which inspires us to design a warm-up stage by regarding FNPs and TNPs as hard and easy patterns, respectively.

In the warm-up stage, our SURE first optimizes the network with Eq. 9 until the average distance of negative pairs is larger than m, thus leading to faster fitting of TNPs comparing with FNPs (see Fig. 3(a) and 3(b)). In consequence, masses of TNPs will have a distance of $d > m$ while most FNPs will fall into the area of $d < m$. This promises that our noise-robust loss mainly affects FNPs instead of TNPs. After that, the proposed

noise-robust contrastive loss (*i.e.*, Eq. 5) is used in the second optimization stage. In this stage, as most FNPs locate in the area of either $0 < d < m/3$ or $m/3 < d < m$, their distances will either decrease (see FNP_robust in Fig. 3(b)) or slowly increase (see FNP_robust in Fig. 3(a)). As a result, the influence of noisy labels is eliminated or mitigated. Notably, the second stage only imposes negligible effects on TNPs as most of their distance is larger than m after the warm-up stage.

3.3 Sufficiency-preserving Versatile Learning

As the contrastive learning paradigm might overemphasize the consistency between views, we further propose a versatile learning module to encourage the common representation to keep sufficiency information as well. The definition of sufficiency in multi-view representation learning is given below.

Definition 5. ***Sufficiency of multi-view representation***. *Let $\boldsymbol{h}_i^{(v)}$ be the view-specific representation of the i-th sample $\mathbf{s}_i^{(v)}$, and $\boldsymbol{h}_i$ be the common representation. $\boldsymbol{h}_i$ is of sufficiency if $\forall v \in [1, V]$, $\mathbf{s}_i^{(v)}$ could be reconstructed from $\boldsymbol{h}_i$ via a mapping $\phi_v(\cdot)$.*

Accordingly, the following versatile loss is proposed to preserve the sufficiency of the learned representation, *i.e.*,

$$\mathcal{L}^{ver} = \frac{1}{2N}\sum_{i=1}^{N}\sum_{v=1}^{2}\left\|\mathbf{s}_i^{(v)} - g_v\left(\left[\mathbf{h}_i^{(1)}, \mathbf{h}_i^{(2)}\right]\right)\right\|_2^2, \tag{15}$$

where g_v is the decoder for the v-th view, and $[\cdot,\cdot]$ denotes the concatenation operation.

3.4 Category-level Alignment and Imputation

With the established cross-view correspondences, we design two strategies to handle PVP and PSP in the inference stage. In brief, the category-level alignment strategy is developed to realign the cross-view samples, and the imputation strategy is used to recover the missing samples. Formally,

Definition 6. ***Category-level Alignment (CA)***. *For each sample $\mathbf{x}_i^{(v_1)}$ from view v_1, CA realigns it with its counterparts $\mathbf{x}_j^{(v_2)}$ in each view v_2 so that*

$$\sum_{v_1}^{V}\sum_{v_2\neq v_1}^{V} C(\mathbf{x}_i^{(v_1)}, \mathbf{x}_j^{(v_2)}) = V(V-1), \tag{16}$$

Definition 7. ***Category-level Imputation (CI)***. *CI imputes the missing sample $\hat{\mathbf{x}}_i^{(v_1)}$ by the weighted sum of its peers $\hat{\mathbf{x}}_j^{(v_1)}$ in the same view,i.e.,*

$$\hat{\mathbf{x}}_i^{(v_1)} = \sum_{j\in\mathcal{E}_i^{v_1}} p_{ij}\mathbf{x}_j^{(v_1)}, \tag{17}$$

where p_{ij} are the weight parameters which sum up to 1, and $\mathcal{E}_i^{v_1}$ is the set of the indices for k cross-view and within-category counterparts of the observable counterpart $\mathbf{x}_i^{(v_2)}$ which is identified by

$$\sum_{j}\sum_{v_1}^{V}\sum_{v_2\neq v_1}^{V} C(\mathbf{x}_i^{(v_2)}, \mathbf{x}_j^{(v_1)}) = kV(V-1), \tag{18}$$

The implementation details of SURE is presented in Algorithm 1.

Algorithm 1 SURE for PVP and PSP

Input: dataset $\{\mathbf{X}^{(v)}\}_{v=1}^2$ of size N_x; the indefective portion $\{\mathbf{S}^{(v)}\}_{v=1}^2$ of size N_s; networks $\{f_v, g_v\}_{v=1}^2$; ratio of negatives to positives M; batch size B; training epoch E.
Output: cluster assignments.

```
// Pair Construction
for i = 1 to N_s do
    From {S^(v)}_{v=1}^2, construct positives {(s_i^(1), s_i^(2))}.
    for cnt = 1 to M do
        From {S^(v)}_{v=1}^2, construct negatives {(s_i^(1), s_j^(2)); i ≠ j}
        by randomly sampling.
    end
end
// Training (Category-level Identification)
Compute margin m using Eq. 11.
Warmup the network using Eq. 9 as introduced in Section 3.2.
for epoch = 1 to E do
    for b = 1 to num_batch do
        Construct a mini-batch of size B from the constructed
        positive and negative pairs.
        Compute the overall loss L in Eq. 4.
        Update {f_v, g_v}_{v=1}^2 through gradient descent to minimize
        L.
    end
end
// Inference (Solutions to PSP and PVP)
for b = 1 to num_batch do
    Extract features from the current batch X of size B via
    f_1(x_i^(1)) and f_2(x_i^(2)).
    Compute the cross-view Euclidean distance matrix D ∈
    R^{B×B} of the learned features.
    // Category-level Imputation
    if {X^(v)}_{v=1}^2 is of PSP (Eq. 2) then
        for the missing samples x_i^(1) in X do
            Compute the indices E_i^1 of k cross-view counterparts
            f_1(x_j^(1)) of the existing representation f_2(x_i^(2)) with
            smallest distances D_ij (j ≠ i).
            Recover the feature h_i^(1) using Eq. 17 where p_ij =
            1/k for simplicity.
        end
    end
    // Category-level Alignment
    if {X^(v)}_{v=1}^2 is of PVP (Eq. 1) then
        for x_i^(1) in X do
            Realign it with its category-level counterpart x_j^(2)
            through j = argmin_{j≠i} D_ij.
        end
    end
end
Perform clustering on the realigned and recovered features of
{X^(v)}_{v=1}^2.
```

4 Experiments

In this section, we evaluate the proposed SURE on the clustering task under the setting of PVP, PSP, and the hybrid of them, respectively. The structure of this section is as follows. First, we present the details about the network architectures and experimental configurations in Section 4.1. Then, we carry out a

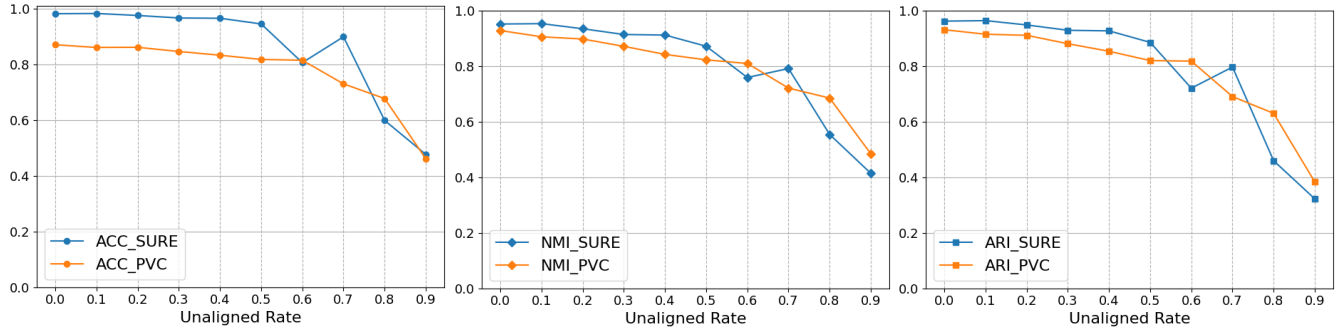


Fig. 4. Performance analysis on NoisyMNIST under the PVP setting with unaligned rates vary from 0.0 t0 0.9 with an interval of 0.1.

TABLE 1
Partially view-unaligned clustering comparisons on four widely-used multi-view datasets including three handcraft-feature-based and the NoisyMNIST datasets, where the first and second best results are in **bold** and underline, respectively. "–" indicates that the method is impractical due to the over-high time or memory consumption.

Aligned	Methods	Scene-15			Caltech-101			Reuters			NoisyMNIST		
		ACC	NMI	ARI	ACC	NMI	ARI	ACC	NMI	ARI	ACC	NMI	ARI
Partially	CCA (NeurIPS'03)	32.73	34.24	18.80	20.06	41.56	16.62	40.87	15.82	12.68	34.46	29.83	17.89
	KCCA (JMLR'02)	33.09	31.43	16.35	12.57	31.36	7.65	40.08	11.80	11.27	26.57	18.19	10.55
	DCCA (ICML'13)	34.27	36.55	18.83	12.52	32.13	7.63	39.71	13.83	14.38	29.22	20.24	11.08
	DCCAE (ICML'15)	33.62	36.56	18.54	11.75	30.54	6.60	41.42	12.82	13.61	27.61	19.45	10.00
	LMSC (CVPR'17)	26.27	20.45	10.93	21.54	40.26	15.51	32.17	11.34	7.19	–	–	–
	MvC-DMF (AAAI'17)	28.49	24.31	11.22	9.54	23.41	3.84	32.58	12.36	11.08	27.34	22.96	6.85
	SwMC (IJCAI'17)	31.03	30.39	12.94	19.03	22.75	3.73	31.92	11.03	5.40	–	–	–
	BMVC (TPAMI'18)	36.81	36.55	20.20	12.13	31.33	7.11	38.15	11.57	12.07	28.47	24.69	14.19
	AE^2-Nets (CVPR'19)	28.56	26.58	12.96	10.45	29.51	7.90	35.49	10.61	8.07	38.25	34.32	22.02
	PVC (NeurIPS'20)	37.88	39.12	20.63	22.11	**47.82**	17.98	42.07	20.43	16.95	81.84	82.29	82.03
	MvCLN (CVPR'21)	38.53	39.90	**24.26**	30.09	43.07	38.34	**50.16**	**30.65**	**24.90**	91.05	84.15	83.56
	SURE (ours)	**40.32**	**40.33**	23.08	**30.87**	44.25	**39.89**	49.99	29.46	24.60	**95.17**	**88.24**	**89.72**
Fully	CCA (NeurIPS'03)	36.37	36.91	19.82	20.25	45.41	16.34	44.31	20.34	14.52	71.31	52.60	48.46
	KCCA (JMLR'02)	37.93	37.42	21.38	21.45	45.58	17.62	**50.87**	22.34	20.61	96.85	92.10	93.23
	DCCA (ICML'13)	36.61	39.20	21.03	27.60	47.84	30.86	47.95	26.57	12.71	89.64	88.33	83.95
	DCCAE (ICML'15)	34.58	39.01	19.65	19.84	45.05	14.57	41.98	20.30	8.51	78.00	81.24	68.15
	LMSC (CVPR'17)	38.46	35.50	20.54	26.87	48.80	18.06	38.56	20.12	15.48	–	–	–
	MvC-DMF (AAAI'17)	30.99	31.35	15.68	24.35	44.98	14.82	33.83	14.89	12.59	74.39	63.22	49.79
	SwMC (IJCAI'17)	33.89	32.98	11.78	30.74	36.07	7.75	33.65	16.02	5.90	–	–	–
	BMVC (TPAMI'18)	40.74	41.67	24.19	27.59	46.43	21.28	42.39	21.86	15.14	88.31	77.01	76.58
	AE^2-Nets (CVPR'19)	37.17	40.47	22.24	20.79	45.01	15.89	42.39	19.76	14.87	42.11	43.38	30.42
	PVC (NeurIPS'20)	38.01	39.82	21.06	21.74	**49.31**	18.48	38.03	20.30	10.05	87.1	92.84	93.14
	MvCLN (CVPR'21)	37.90	42.31	**25.58**	30.41	46.90	42.99	50.60	**29.63**	**25.7**	97.30	94.16	95.31
	SURE (ours)	**42.75**	**42.48**	24.57	**34.16**	48.04	**51.45**	48.35	28.53	23.71	**98.39**	**95.41**	**96.50**

series of quantitative analyses and ablation studies to verify the effectiveness of the proposed SURE on handling PVP and PSP in Section 4.2 and 4.3, respectively. Moreover, we investigate the robustness of SURE under a more challenging situation, where the data simultaneously suffers from PVP and PSP in Section 4.4. Besides, we further conduct experiments on different types of datasets to verify the generalization of SURE in Section 4.5. Finally, we qualitatively analyze the proposed noise-robust contrastive loss to give further understandings in Section 4.6.

4.1 Architectures and Configurations

The encoders and decoders are all of the fully-connected architectures. Specifically, the two encoders f_1 and f_2 have the same structure of *D-1024-1024-1024-10*, where D denotes the dimension of inputs. The two independent decoders are with a dimensionality of *20-1024-1024-1024-D*.

The proposed SURE is implemented in PyTorch 1.5.0 [40] and all the evaluations are carried out on NVIDIA 2080Ti GPUs on the Ubuntu 16.04 platform. To optimize SURE, we adopt the standard Adam optimizer [41] with an initial learning rate of 0.001 without scheduler and weight decay. We train the model for 80 epochs on all datasets, with a batch size of 1024. Besides, the trade-off parameter λ, the number of neighbors k for sample recovering and the negative/positive ratio M are set to 0.5, 3 and 30 on all datasets, respectively.

Seven multi-view datasets (three handcraft-feature-based, two deep-feature-based and two raw) are used in the experiments including

- **Scene15** [42]: The dataset is composed of 4,485 images associated with 15 indoor and outdoor scene categories. Similar to [43], 20-dim GIST feature and 59-dim PHOG feature, are used as two different views;
- **Caltech101** [44]: The dataset consists of 9,144 images distributed over 102 categories. Following [14], two features, *i.e.*, 1,984-dim HOG feature and 512-dim GIST feature, are extracted as two views;
- **Reuters** [45]: The used subset contains 18,758 samples from six classes. Similar to [46], the first two languages

(English and French) are projected into a 10-dim space by a standard autoencoder and used as two views;

- **Deep features of Caltech-101**: Following [47], we use two types of 4096-dim features extracted by DECAF [48] and VGG19 [49] networks as two views.
- **Deep features of Animal**: The raw dataset consists of 10,158 images from 50 classes. Following [5], two types of 4096-dim features extracted by DECAF and VGG19 networks are used as two views.
- **NoisyMNIST** [2]: The whole dataset consists of 70,000 instances of 10 classes. As some baselines cannot deal with such a large-scale dataset, we randomly select 30,000 instances for evaluation.
- **MNIST-USPS**: Following [50], the USPS and MNIST datasets are used as two views. For each dataset, 5,000 samples distributed over 10 digits are randomly selected to constitute this dataset. The MNIST image is of 784-dim and the USPS image is of 256-dim.

If not stated otherwise, the dataset $\{\mathbf{X}^{(v)}\}_{v=1}^2$ will be randomly divided into two equal-sized subsets to simulate the fully-aligned and complete data $\{\mathbf{S}^{(v)}\}_{v=1}^2$ and data $\{\mathbf{W}^{(v)}\}_{v=1}^2$ plagued with PVP or PSP, respectively. In the experiment, after CI and CA, we simply concatenate the view-specific representations as the common representation which is further fed into k-means to obtain the clustering results like the traditional fashion [1], [8], [16], [51]

4.2 Partially View-unaligned Clustering

In this section, we apply SURE on the partially view-unaligned clustering task and compare it with 11 multi-view clustering methods, followed by the time cost and visualization analyses.

4.2.1 Comparisons with State of the Arts

We compare SURE with 11 multi-view clustering baselines including CCA [12], KCCA [11], DCCA [1], DCCAE [2], LMSC [52], MvC-DMF [53], SwMC [54], BMVC [14], AE^2-Nets [4], PVC [8], and MvCLN [10]. For all baselines, we tune their parameters as suggested in the referred works. As most baselines except PVC [8] and MvCLN [10] cannot handle PVP directly, we adopt the following two settings for a fair comparison:

- **Partially view-unaligned**: In this setting, we shuffle $\{\mathbf{W}^{(v)}\}_{v=1}^2$ to construct the unaligned views $\{\mathbf{W}^{(v)}\}_{v=1}^2$ which is then combined with the aligned data $\{\mathbf{S}^{(v)}\}_{v=1}^2$ to obtain the partially view-unaligned data $\{\mathbf{X}^{(v)}\}_{v=1}^2$. PCA is used to project $\{\mathbf{X}^{(v)}\}_{v=1}^2$ into a latent space with the same dimension of SURE and then the Hungarian algorithm is used to establish the cross-view correspondences. After that, the baselines are conducted on the realigned data. Note that for SURE, MvCLN and PVC, we directly evaluate them on $\{\mathbf{X}^{(v)}\}_{v=1}^2$.
- **Fully view-aligned**: In this setting, all baselines excepted PVC and MVCLN are directly conducted on the original data which is fully-aligned. Notably, PVC, MvCNL and SURE still realign views after training because the ground truth correspondences are unavailable to them.

After learning the representations for different views, following [8], [10], [55], [56], we simply concatenate them and conduct k-means to achieve clustering except for Mvc-DMF, SwMC, and BMVC since they could directly obtain the clustering results. The clustering performance on three handcraft-feature-based and the raw NoisyMNIST datasets is reported in Table 1 from which one could observe that:

- In the first setting, our SURE significantly outperforms nearly all the baselines on all four datasets. Specifically, SURE surpasses the best baseline by 4.52%, 4.65% in terms of ACC on NoisyMNIST and Scene-15, respectively. This verifies the effectiveness of SURE on handling PVP.
- In the second setting, our SURE again almost achieves the state-of-the-art performance even though most baselines are with the ground-truth correspondences. In some cases, PVC, MvCLN and SURE give inferior results compared to the former setting as they realign all the data.

4.2.2 Time Cost Comparison

The running time comparison between the Hungarian algorithm [9], PVC [8] and SURE are summarized in Table 2. The Hungarian algorithm is implemented with the package in Scipy [57]. As shown in Table 2, our method is remarkably efficient than the Hungarian algorithm and PVC on all datasets, which verifies the higher accessibility of our category-level alignment strategy compared to the instance-level one. Particularly, the increasing data size highlights the superiority of SURE. Note that PVC and SURE runs on the GPU, whereas the Hungarian algorithm run on the CPU.

TABLE 2
Time cost comparisons (in seconds). "()" indicates the time speedup and "–" means the method does not need that stage.

Dataset	Method	Training Time	Inference Time
Scene-15	Hungarian [9]	–	3 (1×)
	PVC [8]	10907 (1×)	2 (1.5×)
	SURE	**174** (**62.7**×)	**1** (**3.0**×)
Caltech-101	Hungarian [9]	–	49 (1×)
	PVC [8]	11840 (1×)	7 (7.0×)
	SURE	**408** (**29.0**×)	**2** (**24.5**×)
Reuters	Hungarian [9]	–	290 (1×)
	PVC [8]	18715 (1×)	30 (9.7×)
	SURE	**773** (**24.2**×)	**3** (**96.7**×)
NoisyMNIST	Hungarian [9]	–	3778 (1×)
	PVC [8]	53071 (1×)	34 (111.1×)
	SURE	**1333** (**39.8**×)	**6** (**629.7**×)

TABLE 3
Effectiveness of two losses on PVP. "✓" and "✗" represent SURE with and without the corresponding loss, respectively.

$\mathcal{L}^{ncl}$	$\mathcal{L}^{ver}$	ACC	NMI	ARI	CAR
✗	✗	76.48	65.83	62.08	81.15
✗	✓	81.12	69.98	67.58	82.05
✓	✗	91.05	84.15	83.56	87.36
✓	✓	**95.17**	**88.24**	**89.72**	**87.62**

4.2.3 Ablation Studies

In this section, we carry out the following ablation studies on NoisyMNIST to investigate the effectiveness of each module. To help understand the performance of our realignment, we introduce

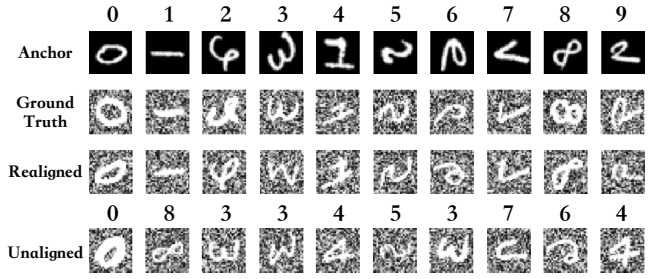


Fig. 5. Visualization of cross-view correspondences on NoisyMNIST. From top to bottom, the rows correspond to the anchor view, the view with the ground truth correspondence, the category-level view realigned by SURE, and the unaligned view, respectively.

TABLE 4
Effectiveness of two losses on PSP. "✓" and "×" denote SURE with and without the corresponding loss, respecitively.

$\mathcal{L}^{ncl}$	$\mathcal{L}^{ver}$	ACC	NMI	ARI
×	×	82.50	70.62	68.07
×	✓	83.19	72.51	70.45
✓	×	92.30	84.77	84.61
✓	✓	93.01	85.40	85.92

a new metric termed Category-level Alignment Rate (CAR). Mathematically,

$$CAR = \frac{1}{N}\sum_{i=1}^{N}\delta\left(C\left(\mathbf{x}_i^{(1)}, \mathbf{x}_i^{(2)}\right)\right), \tag{19}$$

where δ is the Dirichlet function, and N is the number of data pairs.

Effectiveness of Two Losses: To evaluate the indispensability of each loss in SURE, we conduct the following experiments by only using one of the noise-robust contrastive loss $\mathcal{L}^{ncl}$ and the versatile loss $\mathcal{L}^{ver}$. The results are reported in Table 3. Note that the vanilla loss $\mathcal{L}^{van}$ is used as an alternative of $\mathcal{L}^{ncl}$ when the latter is not used. As shown in Table 3, $\mathcal{L}^{ncl}$ and $\mathcal{L}^{ver}$ play critic roles in SURE. Particularly, $\mathcal{L}^{ncl}$ remarkably improves the performance as it could help SURE learning view-consistent representations while preventing the FNPs from dominating the network optimization. Besides, one could observe that $\mathcal{L}^{ver}$ contributes more to the improvement on clustering performance compared to the CAR. This phenomenon verifies our claim that the versatile loss could help the common representation retain the sufficiency to boost the clustering performance.

Influence of Different Unaligned Rates: To evaluate the performance of SURE on partially view-unaligned data *w.r.t.* different unaligned rates, we compare SURE with PVC [8] by increasing the unaligned rate from 0% to 90% with an interval of 10% on NoisyMNIST. The results are summarized in Fig. 4, from which one could have the following observations. First, 50% aligned data (*i.e.*, 50% unaligned rate) is adequate for SURE to learn the latent alignment patterns. Second, SURE outperforms PVC under different unaligned rates in most cases, which demonstrates the effectiveness and robustness of SURE in handling PVP.

4.2.4 Visualization

To further understand the effect of the realignment solution, we visualize some realigned samples on NoisyMNIST in Fig. 5. The results show that SURE establishes the correct correspondence for the anchors, which illustrates the powerful alignment ability of SURE.

4.3 Partially Sample-missing Clustering

In this section, we evaluate SURE on the task of partially sample-missing clustering comparing with 10 multi-view clustering methods to verify the effectiveness of SURE on handling PSP.

4.3.1 Comparisons with State of the Arts

We compare SURE with 10 multi-view clustering baselines including CCA [12], KCCA [11], DCCA [1], DCCAE [2], BMVC [14], AE^2-Nets [4], PMVC [51], UEAF [58], DAIMC [16] and EERIMC [17]. To evaluate their performance on the partially sample-missing clustering task, we randomly discard some samples of several instances in an arbitrary view, resulting in the incomplete dataset. The missing rate is defined as $\gamma = m/n$, where n is the size of dataset and m is the number of instances with missing samples. For fair comparisons, we conduct SURE and baselines under two settings, *i.e.*, $\gamma = 0.5$ (denoted by *Incomplete*) and $\gamma = 0$ (denoted by *Complete*). As the first six baselines cannot handle the incomplete data directly, we preprocess them by filling the missing samples with the mean of the entire view.

According to Table 5, SURE is remarkably superior to other baselines in both two settings on three handcraft-feature-based and the raw NoisyMNIST datasets. Particularly, in the *Incomplete* setting, SURE achieves an NMI improvement of 33.7% and 35.8% on the Scene and NoisyMNIST dataset, respectively. Besides, in the *Complete* setting, SURE still gains an ARI improvement of 14.9% and 58.1% on the NoisyMNIST and Caltech-101 dataset, respectively. The promising performance improvement indicates the robustness of SURE against sample missing.

4.3.2 Ablation Studies

In this section, we conduct the following ablation studies to evaluate the influence of two losses of SURE, the missing rate γ, the peer numbers on partially sample-missing clustering.

Effectiveness of Two Losses: Similar to the ablation study on the PVP task, we investigate the importance of $\mathcal{L}^{ncl}$ and $\mathcal{L}^{ver}$ for the PSP task. As shown in Table 4, both two losses are indispensable to conquer PSP.

Performance under Different Missing Rate: We investigate the performance of SURE by conducting experiments on the NoisyMNIST dataset with missing rates varying from 0 to 0.9 with an interval of 0.1. As depicted in Fig. 6, SURE significantly outperforms all baselines under different missing rates, which demonstrates the effectiveness of SURE in handling PSP.

Influence of Different Peer Numbers: As elaborated in Section 3.1 and 3.4, SURE recovers the missing samples using the combination of their k peers. To evaluate the influence of k, we investigate its value in the range of $\{1, 3, 5, 7, 9\}$. Fig. 8 shows that our SURE is quite robust to the number of peers. Notably, in our implementation, we set $k = 3$ for all datasets for simplicity.

4.3.3 Visualization

In this section, we show the powerful recovering ability of SURE by visualizing the imputed samples in Fig 7. From the results, one could see that SURE successfully recovers the missing samples. It is interesting to note that the recovered samples rarely contain the interfering information (*e.g.*, noises) comparing with the original samples, which could boost the downstream discrimination task such as clustering.

TABLE 5
Partially sample-missing clustering comparisons on four widely-used multi-view datasets, where the first and second best results are in **bold** and underline, respectively.

Missing	Methods	Scene-15			Caltech-101			Reuters			NoisyMNIST		
		ACC	NMI	ARI	ACC	NMI	ARI	ACC	NMI	ARI	ACC	NMI	ARI
Incomplete	CCA (NeurIPS'03)	27.11	25.78	10.49	16.07	35.60	3.54	36.82	10.96	6.63	50.44	36.57	19.59
	KCCA (JMLR'02)	27.15	26.24	10.56	12.38	29.02	6.55	46.13	17.44	15.26	53.79	48.15	28.84
	DCCA (ICML'13)	28.78	28.35	13.24	15.39	38.38	13.13	45.84	26.08	18.00	65.75	61.72	41.17
	DCCAE (ICML'15)	29.01	29.13	12.86	15.34	38.11	12.76	47.04	28.00	14.48	65.42	62.87	38.32
	BMVC (TPAMI'18)	32.45	30.87	11.56	18.92	34.16	4.3	32.1	6.98	2.89	30.71	19.16	10.6
	AE^2-Nets (CVPR'19)	22.44	23.43	9.56	6.93	20.29	3.95	29.08	7.55	4.84	29.88	23.78	11.81
	PMVC (AAAI'14)	25.47	25.37	11.31	21.79	44.74	19.05	29.32	7.42	4.42	33.13	25.49	14.62
	UEAF (AAAI'19)	28.95	26.92	8.37	21.50	42.19	13.78	33.32	20.06	12.19	37.45	34.42	25.71
	DAIMC (IJCAI'18)	27.00	23.47	10.62	23.57	44.11	17.2	40.94	18.66	15.04	33.81	26.42	15.96
	EERIMVC (TPAMI'20)	31.50	31.11	14.82	27.06	45.08	19.98	29.77	12.01	4.21	55.62	45.92	36.76
	SURE (ours)	**39.60**	**41.58**	**23.49**	**32.19**	**45.60**	**43.26**	**47.18**	**30.89**	**23.32**	**93.01**	**85.40**	**85.92**
Complete	CCA (NeurIPS'03)	36.37	36.91	19.82	20.25	45.41	16.34	44.31	20.34	14.52	71.31	52.60	48.46
	KCCA (JMLR'02)	37.93	37.42	21.38	21.45	45.58	17.62	**50.87**	22.34	20.61	85.54	86.51	82.58
	DCCA (ICML'13)	36.61	39.20	21.03	27.60	47.84	30.86	47.95	26.57	12.71	89.64	88.33	83.95
	DCCAE (ICML'15)	34.58	39.01	19.65	19.84	45.05	14.57	41.98	20.30	8.51	78.00	81.24	68.15
	BMVC (TPAMI'18)	40.50	41.20	24.11	27.59	46.43	21.28	42.39	21.86	15.14	88.31	77.01	76.58
	AE^2-Nets (CVPR'19)	37.17	40.47	22.24	20.79	45.01	15.89	42.39	19.76	14.87	52.83	51.24	39.52
	PMVC (AAAI'14)	30.83	31.05	14.98	26.92	**50.5**	26.00	32.5	11.11	7.48	41.09	36.36	24.47
	UEAF (AAAI'19)	34.37	36.69	18.52	25.30	46.02	18.46	40.19	24.34	15.94	66.22	64.34	54.83
	DAIMC (IJCAI'18)	32.09	33.55	17.42	26.4	49.18	19.00	40.78	21.15	15.98	38.40	34.66	22.98
	EERIMVC (TPAMI'20)	39.60	38.99	22.06	23.98	45.61	17.19	33.21	14.28	3.9	65.66	57.60	51.34
	SURE (ours)	**40.95**	**43.19**	**25.01**	**34.59**	48.30	**48.79**	49.06	**29.91**	**23.56**	**98.36**	**95.38**	**96.43**

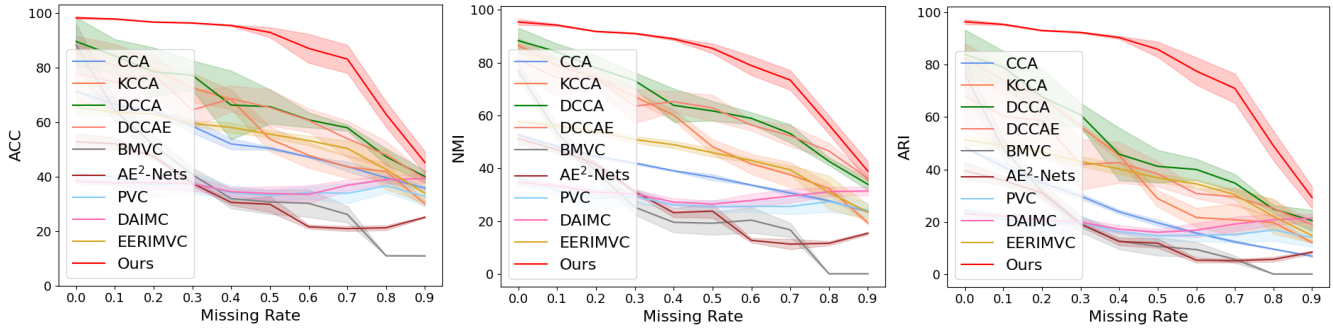


Fig. 6. Performance analysis on NoisyMNIST with different missing rates.

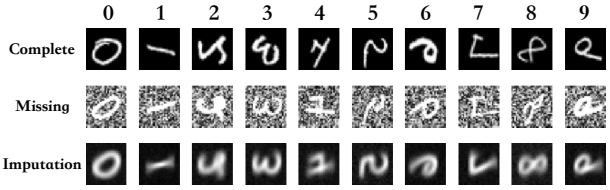


Fig. 7. Samples imputed by SURE on NoisyMNIST. The first and second row corresponds to the complete view and missing view, respectively. The last row are the samples recovered by our SURE.

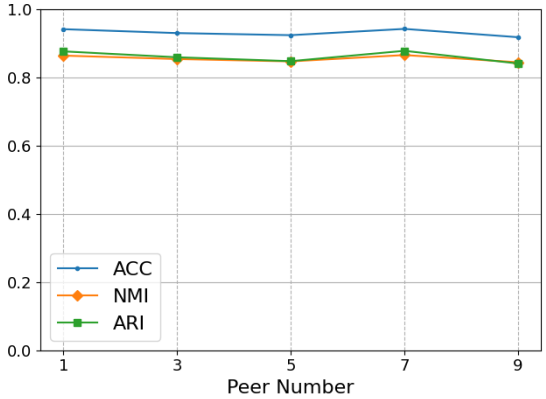


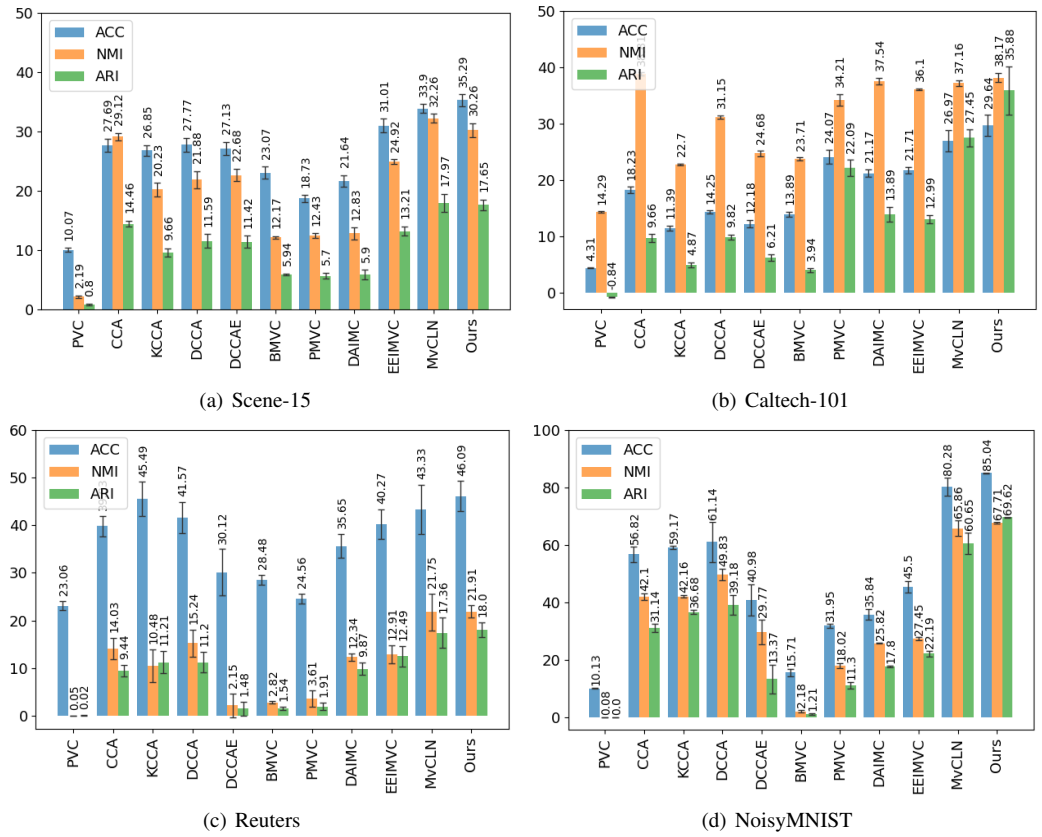
Fig. 8. Performance on NoisyMNIST with different of peer numbers.

4.4 Robust Multi-view Clustering

In this section, we investigate the effectiveness of SURE on the clustering task under a more challenging setting where PSP and PVP simultaneously occur.

4.4.1 *Comparisons with State of the Arts*

As there is no solution for handling the case where PSP and PVP simultaneously occur, we choose the best baselines for PVP and PSP, namely, PVC [8], MvCLN [10], CCA [12], KCCA [11], DCCA [1], DCCAE [2], BMVC [14], PMVC [51], DAIMC [16] and EERIMC [17] as comparative methods in this setting. To simulate the setting, we randomly remove some samples with a missing rate of $\gamma = 0.5$ in the previously constructed unaligned portion $\{\mathbf{W}^{(v)}\}_{v=1}^{2}$. Again, since the former six baselines cannot directly handle PSP, we first impute the missing samples by using the mean of their corresponding view and conduct these baselines



(a) Scene-15 (b) Caltech-101

(c) Reuters (d) NoisyMNIST

Fig. 9. Clustering comparisons on four widely-used multi-view datasets under the setting where PSP and PVP simultaneously occur. The numbers and bars denote the average performances and standard deviations.

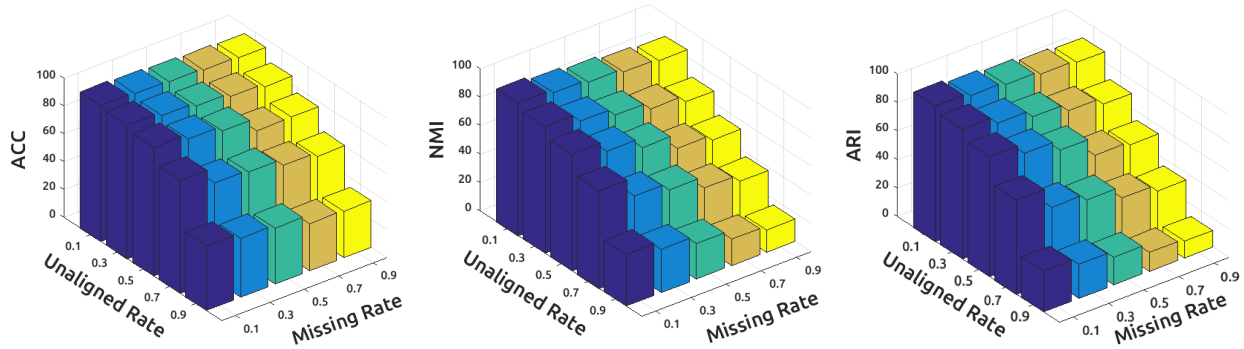


Fig. 10. Performance on NoisyMNIST with different unaligned rates and missing rates. The Heights of the bar denote the performance.

on the preprocessed dataset. For the other three baselines, we directly apply them on the constructed datasets.

As depicted in Fig. 9, SURE still achieves state-of-the-art performance on three handcraft-feature-based and the raw NoisyMNIST datasets, which proves the robustness of our method even under such a challenging case. Notbly, PVC might achieve trivial solutions, and the possible reason is the severely damaged view consistence caused by the wrong realignments on naively imputed data.

4.4.2 *Ablation Studies*

To further investigate the robustness of SURE on simultaneously handling PVP and PSP, following [59], we carry out experiments on the cases where both unaligned rate and missing rate vary from 0.1 to 0.9 with an interval of 0.2. As shown in Fig. 10, SURE achieves stable results with different missing rates. Besides, as the unaligned rate decreases, SURE gradually achieves better performances, which demonstrates that SURE benefits from more training data.

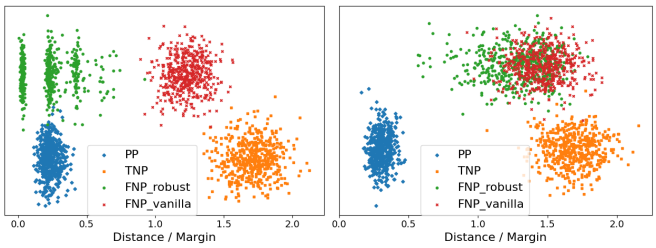
4.5 Generalization among Different Types of Dataset

The quantitative results and qualitative analyses have well validated the effectiveness of SURE on four widely used datasets including three handcraft-feature-based and one raw-data-based. In this section, to further verify the generalization of SURE among different types of datasets, we further conduct experiments on deep features and end-to-end learning. For the evaluation on deep features, Caltech-101 [47] and Animal [5] were used. For

TABLE 6
Clustering performance on different types of datasets under the setting of PVP, PSP and both of them respectively, where the first and second best results of each setting are in **bold** and underline

Type	Methods	Deep Caltech-101			Deep Animal			MNIST-USPS		
		ACC	NMI	ARI	ACC	NMI	ARI	ACC	NMI	ARI
Partially Aligned	PVC (NeurIPS'20)	18.59	48.89	14.60	3.83	0.00	0.00	86.54	78.08	74.60
	MvCLN (CVPR'21)	35.55	60.99	**40.90**	26.24	40.24	19.74	89.96	81.36	80.40
	SURE (ours)	**46.18**	**70.68**	32.98	**27.65**	**40.76**	**19.85**	**92.14**	**82.83**	**83.47**
Fully Aligned	PVC (NeurIPS'20)	20.54	51.40	15.66	3.83	0.00	0.00	95.28	90.36	90.05
	MvCLN (CVPR'21)	39.55	65.29	**32.81**	35.28	**54.19**	29.37	98.76	96.47	97.27
	SURE (ours)	**43.77**	**70.05**	29.46	**35.76**	53.62	**29.51**	**99.12**	**97.49**	**98.05**
Incomplete	DCCA (ICML'13)	27.33	57.60	20.70	26.64	41.54	15.74	78.29	75.69	68.33
	DCCAE (ICML'15)	29.08	**58.79**	**23.38**	22.95	37.60	12.61	79.52	79.19	68.40
	SURE (ours)	**34.55**	57.77	19.87	**30.34**	**48.92**	**24.80**	**92.34**	**84.99**	**84.31**
Complete	DCCA (ICML'13)	45.13	67.07	33.95	33.67	47.86	22.38	87.19	91.65	86.73
	DCCAE (ICML'15)	**45.81**	68.56	**37.65**	30.00	43.82	17.97	96.80	97.73	96.58
	SURE (ours)	42.03	**69.01**	28.89	**36.80**	**55.00**	**30.74**	**99.31**	**98.06**	**98.47**
PSP+PVP	DCCA (ICML'13)	24.35	43.26	17.56	21.87	28.92	10.14	69.02	46.47	44.22
	MvCLN (CVPR'21)	33.94	56.01	**32.29**	24.99	**36.56**	14.95	81.02	**69.81**	59.44
	SURE (ours)	**38.87**	**61.37**	23.40	**25.18**	33.92	**15.97**	**83.02**	65.30	**66.07**

the evaluation on end-to-end learning, the MNIST-USPS raw dataset [50] was used. For each test, we compared SURE with two most competitive methods. Note that the experiments on the NoisyMNIST (Section 4.2-4.4) and MNIST-USPS raw dataset are both conducted in an end-to-end manner. The difference is that the two views of NoisyMNIST are constructed from the MNIST dataset while the ones of MNIST-USPS are from different datasets which is more general. In the new experiments, we still used the same backbone and fixed the hyper-parameters for all the datasets. For the baselines, we have tuned their parameters as suggested in the referred works. As shown in Table 6, SURE achieves the best performance on three new datasets under three different settings in most case. Noticed, SURE performs remarkably better on deep features than handcrafted ones *w.r.t.* Caltech-101. This result proves that the performance of SURE could be further improved if more powerful deep features are used.



(a) NoisyMNIST (b) Reuters

Fig. 11. The distribution of four kinds of pairs on NoisyMNIST and Reuters after training, where PP, TNP, FNP_robust, and FNP_vanilla denote positive pairs, true negative pairs, false-negative pairs optimized by our loss in Eq. 5, and false-negative pairs optimized by the vanilla loss in Eq. 9. For a clear illustration, we randomly sample 500 pairs of each kind and let them obey the normal distribution on the y-axis. The x-axis denotes the ratio of the average distance to the margin m.

4.6 Effectiveness of the Noise-robust Contrastive Loss

In this section, we conduct a series of qualitative and parameter analyses on the NoisyMNIST and Reuters dataset to verify the reversed and slow optimization properties of the proposed noise-robust loss.

4.6.1 Pairwise Distance Distribution

To show that our noise-robust loss could mitigate or even eliminate the influence of FNPs, we plot the distribution of four kinds of pairs after training with our noise-robust loss and vanilla loss, respectively. As shown in Fig 11(a), after training with our noise-robust loss, FNPs are enforced to decrease their distance just as the positive pairs, which verifies its "reversed optimization" property once. In Fig. 11(b), after training with our noise-robust loss, the distance of FNPs is smaller than that trained with the vanilla loss, which verifies its "slow optimization" property.

4.6.2 Influence of the Positive/Negative Pairs Ratio

Our SURE uses the fully-aligned and complete data as positives and randomly selects cross-view samples as negatives. In other words, it is feasible to obtain more negative pairs. However, an exorbitant negatives/positive ratio M could contribute to the unbalanced data distribution. To explore the influence of M, we investigate the performance of SURE by increasing M from 1 to 50 with an interval of 5. As depicted in Fig. 12(a), a moderately large M boosts the performance. Notably, SURE achieves stable performances for $M \in [20, 40]$, which demonstrates that SURE is insensitive to this parameter.

4.6.3 Influence of the Switching Time between Two Optimization Stages

As elaborated in Section 3.2, the optimization of SURE is composed of two stages which are automatically switched in a data-driven way. In this section, we carry out experiments to evaluate the performance of SURE with the following seven switching criteria, *i.e.*, we switch to the second optimization stage when the mean distance of negative pairs reaches $0.0m$, $0.2m$, $0.4m$, $0.6m$, $0.8m$, $1.0m$, or $1.2m$, where the margin m is calculated by Eq. 11. As depicted in Fig. 12(b), SURE performs stably within the range of $[0.2m, 1.0m]$. On the one hand, without the warm-up stage (*i.e.*, the $0.0m$ case), SURE will achieve an inferior result because TNPs and FNPs are not separated well as discussed before. On the other hand, when the switching time is too late ($1.2m$), the distance of most FNPs may approach or even surpass

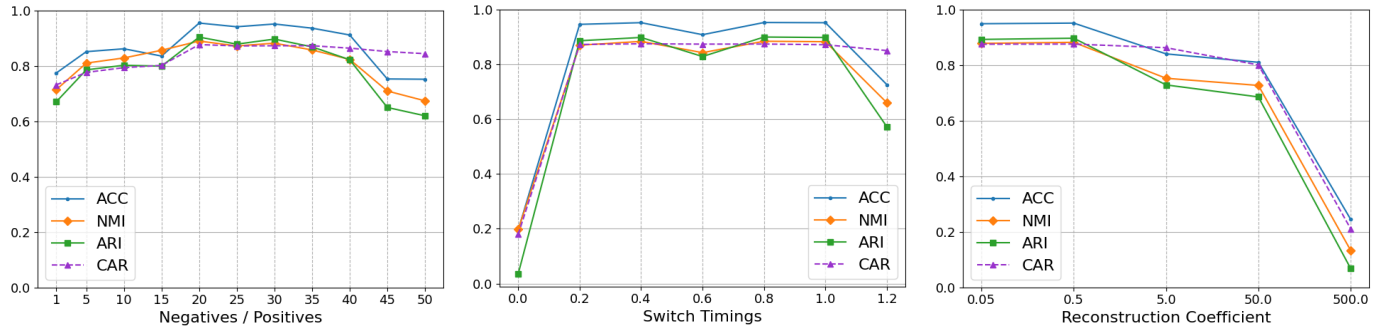


Fig. 12. Clustering performance *w.r.t.* **(a)** negatives/positives ratio M; **(b)** different switching times of two optimization stages; and **(c)** reconstruction coefficient λ on the NoisyMNIST dataset.

m, and thus degrading the effect of our noise-robust contrastive loss.

4.6.4 Parameter Analysis

To investigate the influence of the trade-off parameter λ in Eq. 4, we try different values within the range of $\{0.05, 0.5, 5.0, 50.0, 500.0\}$ on NoisyMNIST. As shown in Fig 12(c), an over-high value of λ is harmful to SURE as our noise-robust contrastive loss would be almost neglected.

5 Conclusion

In this paper, we propose a robust multi-view clustering method which could be the first unified framework for handling PVP and PSP. Different from most existing works that resort to instance-level alignment or imputation, we treat PVP and PSP as a unified category-level identification task which is achieved using a novel noise-robust contrastive loss. We theoretically and experimentally show that our loss could mitigate or even eliminate the influence of the false-negative pairs introduced during the pair construction. Extensive experiments verify the effectiveness and efficiency of the proposed method. In the future, we would like to investigate how to endow our method with the ability to cope with fully instead of partially view-unaligned and sample-missing data. Besides, although our method could be easily extended to handle the data with a larger number of views by focusing on two of them at a time, it is worth exploring how to reduce its computational complexity for handling such a case.

Acknowledgments

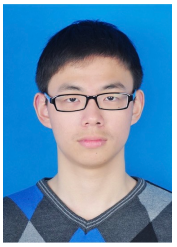
The authors would thank to the associate editor and anonymous reviewers for the constructive comments and valuable suggestions that greatly improve this work,

References

[1] G. Andrew, R. Arora, J. Bilmes, and K. Livescu, "Deep canonical correlation analysis," in *ICML*, 2013, pp. 1247–1255.
[2] W. Wang, R. Arora, K. Livescu, and J. Bilmes, "On deep multi-view representation learning," in *ICML*, 2015, pp. 1083–1092.
[3] Y. Li, M. Yang, and Z. Zhang, "A survey of multi-view representation learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 31, no. 10, pp. 1863–1883, 2018.
[4] C. Zhang, Y. Liu, and H. Fu, "Ae2-nets: Autoencoder in autoencoder networks," in *CVPR*, 2019, pp. 2577–2585.
[5] C. Zhang, Y. Cui, Z. Han, J. T. Zhou, H. Fu, and Q. Hu, "Deep partial multi-view learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.
[6] Y. Jiang, Q. Xu, Z. Yang, X. Cao, and Q. Huang, "Dm2c: Deep mixed-modal clustering." in *NeurIPS*, 2019, pp. 5880–5890.
[7] C. Xu, Z. Guan, W. Zhao, H. Wu, Y. Niu, and B. Ling, "Adversarial incomplete multi-view clustering." in *IJCAI*, 2019, pp. 3933–3939.
[8] Z. Huang, P. Hu, J. T. Zhou, J. Lv, and X. Peng, "Partially view-aligned clustering," *NeurIPS*, vol. 33, 2020.
[9] H. W. Kuhn, "The hungarian method for the assignment problem," *Naval research logistics quarterly*, vol. 2, no. 1-2, pp. 83–97, 1955.
[10] M. Yang, Y. Li, Z. Huang, Z. Liu, P. Hu, and X. Peng, "Partially view-aligned representation learning with noise-robust contrastive loss," in *CVPR*, 2021.
[11] F. R. Bach and M. I. Jordan, "Kernel independent component analysis," *Journal of machine learning research*, vol. 3, no. Jul, pp. 1–48, 2002.
[12] A. Vinokourov, N. Cristianini, and J. Shawe-Taylor, "Inferring a semantic representation of text via cross-language correlation analysis," in *NeurIPS*, 2003, pp. 1497–1504.
[13] Z. Yang, Q. Xu, W. Zhang, X. Cao, and Q. Huang, "Split multiplicative multi-view subspace clustering," *IEEE Transactions on Image Processing*, pp. 5147–5160, 2019.
[14] Z. Zhang, L. Liu, F. Shen, H. T. Shen, and L. Shao, "Binary multi-view clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 7, pp. 1774–1782, 2018.
[15] C. Xu, D. Tao, and C. Xu, "Multi-view learning with incomplete views," *IEEE Transactions on Image Processing*, vol. 24, no. 12, pp. 5812–5825, 2015.
[16] M. Hu and S. Chen, "Doubly aligned incomplete multi-view clustering," *arXiv:1903.02785*, 2019.
[17] X. Liu, M. Li, C. Tang, J. Xia, J. Xiong, L. Liu, M. Kloft, and E. Zhu, "Efficient and effective regularized incomplete multi-view clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.
[18] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality reduction by learning an invariant mapping," in *CVPR*, vol. 2, 2006, pp. 1735–1742.
[19] A. v. d. Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *arXiv:1807.03748*, 2018.
[20] M. Caron, I. Misra, J. Mairal, P. Goyal, P. Bojanowski, and A. Joulin, "Unsupervised learning of visual features by contrasting cluster assignments," *arXiv:2006.09882*, 2020.
[21] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *CVPR*, 2015, pp. 815–823.
[22] M. Gutmann and A. Hyvärinen, "Noise-contrastive estimation: A new estimation principle for unnormalized statistical models," in *AISTATS*, 2010, pp. 297–304.
[23] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," *arXiv:2002.05709*, 2020.
[24] J.-B. Grill, F. Strub, F. Altché, C. Tallec, P. H. Richemond, E. Buchatskaya, C. Doersch, B. A. Pires, Z. D. Guo, M. G. Azar *et al.*, "Bootstrap your own latent: A new approach to self-supervised learning," *arXiv:2006.07733*, 2020.
[25] T. Huynh, S. Kornblith, M. R. Walter, M. Maire, and M. Khademi, "Boosting contrastive self-supervised learning with false negative cancellation," *arXiv:2011.11765*, 2020.

[26] T.-S. Chen, W.-C. Hung, H.-Y. Tseng, S.-Y. Chien, and M.-H. Yang, "Incremental false negative detection for contrastive learning," *arXiv:2106.03719*, 2021.
[27] B. Han, Q. Yao, X. Yu, G. Niu, M. Xu, W. Hu, I. Tsang, and M. Sugiyama, "Co-teaching: Robust training of deep neural networks with extremely noisy labels," *arXiv:1804.06872*, 2018.
[28] B. Han, J. Yao, G. Niu, M. Zhou, I. Tsang, Y. Zhang, and M. Sugiyama, "Masking: A new perspective of noisy supervision," *arXiv:1805.08193*, 2018.
[29] X. Wang, S. Wang, J. Wang, H. Shi, and T. Mei, "Co-mining: Deep face recognition with noisy labels," in *CVPR*, 2019, pp. 9358–9367.
[30] X. Ma, H. Huang, Y. Wang, S. Romano, S. Erfani, and J. Bailey, "Normalized loss functions for deep learning with noisy labels," in *ICML*, 2020, pp. 6543–6553.
[31] H. Song, M. Kim, D. Park, and J.-G. Lee, "Learning from noisy labels with deep neural networks: A survey," *arXiv:2007.08199*, 2020.
[32] Y. Kim, J. Yun, H. Shon, and J. Kim, "Joint negative and positive learning for noisy labels," *arXiv:2104.06574*, 2021.
[33] T. Xiao, T. Xia, Y. Yang, C. Huang, and X. Wang, "Learning from massive noisy labeled data for image classification," in *CVPR*, 2015, pp. 2691–2699.
[34] D. Hendrycks, M. Mazeika, D. Wilson, and K. Gimpel, "Using trusted data to train deep networks on labels corrupted by severe noise," *arXiv:1802.05300*, 2018.
[35] K.-H. Lee, X. He, L. Zhang, and L. Yang, "Cleannet: Transfer learning for scalable image classifier training with label noise," in *CVPR*, 2018, pp. 5447–5456.
[36] J. Han, P. Luo, and X. Wang, "Deep self-learning from noisy labels," in *CVPR*, 2019, pp. 5138–5147.
[37] J. Li, R. Socher, and S. C. Hoi, "Dividemix: Learning with noisy labels as semi-supervised learning," *arXiv:2002.07394*, 2020.
[38] P. Khosla, P. Teterwak, C. Wang, A. Sarna, Y. Tian, P. Isola, A. Maschinot, C. Liu, and D. Krishnan, "Supervised contrastive learning," *arXiv:2004.11362*, 2020.
[39] D. Arpit, S. K. Jastrzebski, N. Ballas, D. Krueger, E. Bengio, M. S. Kanwal, T. Maharaj, A. Fischer, A. C. Courville, Y. Bengio *et al.*, "A closer look at memorization in deep networks," in *ICML*, 2017, pp. 233–242.
[40] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *arXiv:1912.01703*, 2019.
[41] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv:1412.6980*, 2014.
[42] L. Fei-Fei and P. Perona, "A bayesian hierarchical model for learning natural scene categories," in *CVPR*, vol. 2, 2005, pp. 524–531.
[43] D. Dai and L. Van Gool, "Ensemble projection for semi-supervised image classification," in *ICCV*, 2013, pp. 2072–2079.
[44] F. Li, M. Andreetto, M. Ranzato, and P. Perona, "Caltech101," *Computational Vision Group, California Institute of Technology*, 2003.
[45] M.-R. Amini, N. Usunier, and C. Goutte, "Learning from multiple partially observed views - an application to multilingual text categorization," in *NeruIPS*, 2009, pp. 28–36.
[46] Z. Huang, J. T. Zhou, X. Peng, C. Zhang, H. Zhu, and J. Lv, "Multi-view spectral clustering network." in *IJCAI*, 2019, pp. 2563–2569.
[47] Z. Han, C. Zhang, H. Fu, and J. T. Zhou, "Trusted multi-view classification," in *ICLR*, 2021.
[48] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," vol. 25, pp. 1097–1105, 2012.
[49] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv:1409.1556*, 2014.
[50] X. Peng, Z. Huang, J. Lv, H. Zhu, and J. T. Zhou, "Comic: Multi-view clustering without parameter selection," in *ICML*, 2019, pp. 5092–5101.
[51] S.-Y. Li, Y. Jiang, and Z.-H. Zhou, "Partial multi-view clustering," in *AAAI*, 2014.
[52] C. Zhang, Q. Hu, H. Fu, P. Zhu, and X. Cao, "Latent multi-view subspace clustering," in *CVPR*, 2017, pp. 4279–4287.
[53] H. Zhao and Z. Ding, "Multi-view clustering via deep matrix factorization," in *AAAI*, 2017, pp. 2921–2927.
[54] F. Nie, J. Li, X. Li *et al.*, "Self-weighted multiview clustering with multiple graphs." in *IJCAI*, 2017, pp. 2564–2570.
[55] Y. Jiang, Z. Yang, Q. Xu, X. Cao, and Q. Huang, "When to learn what: Deep cognitive subspace clustering," in *ACM MM*, 2018, pp. 718–726.
[56] Y. Jiang, Q. Xu, Z. Yang, X. Cao, and Q. Huang, "Duet robust deep subspace clustering," in *ACM MM*, 2019, pp. 1596–1604.
[57] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright *et al.*, "Scipy 1.0: fundamental algorithms for scientific computing in python," *Nature methods*, vol. 17, no. 3, pp. 261–272, 2020.
[58] J. Wen, Z. Zhang, Y. Xu, B. Zhang, L. Fei, and H. Liu, "Unified embedding alignment with missing views inferring for incomplete multi-view clustering," in *AAAI*, 2019, pp. 5393–5400.
[59] Y. Jiang, Q. Xu, K. Ma, Z. Yang, X. Cao, and Q. Huang, "What to select: Pursuing consistent motion segmentation from multiple geometric models," in *AAAI*, 2021, pp. 1708–1716.

Mouxing Yang received the bachelor's degree in computer science and technology from Sichuan University, Chengdu, China, in 2020, where he is currently pursuing the master's degree in computer science with the College of Computer Science. His research interest includes multi-modal representation learning.



Yunfan Li received the bachelor's degree in computer science and technology from Sichuan University, Chengdu, China, in 2020, where he is currently pursuing the Ph.D. degree in computer science with the College of Computer Science. His research interest includes unsupervised learning.

Peng Hu received the Ph.D. degree in computer science and technology from Sichuan University, China, in 2019. He is currently an associate research professor at the College of Computer Science, Sichuan University. His research interests mainly focus on multi-view learning, cross-modal retrieval, and network compression. On these areas, he has authored more than 20 articles in the top-tier conferences and journals.

Jinfeng Bai received the Ph.D. degree from the University of Chinese Academy of Sciences, Beijing, China, in 2015. He is currently working in Tomorrow Advancing Life Education Group (TAL) and is in charge of the AI laboratory. His research interests cover speech recognition, speech synthesis, image text recognition, sequence learning, natural language processing, computer vision and other fields.

Jiancheng Lv (Member, IEEE) received the Ph.D. degree in computer science and engineering from the University of Electronic Science and Technology of China, Chengdu, China, in 2006. He is currently a Professor with the Data Intelligence and Computing Art Laboratory, College of Computer Science, Sichuan University, Chengdu. His research interests include neural networks, machine learning, and big data.

Xi Peng is currently a full professor at College of Computer Science, Sichuan University. His current interests mainly focus on machine learning and multi-media analysis. On these areas, he has authored more than 70 articles published in JMLR, TPAMI, ICML, NeurIPS, and so on. Dr. Peng has served as an Associate Editor for four journals such as " IEEE Trans on SMC: Systems", a Guest Editor for four journals such as "IEEE Trans. on Neural Network and Learning Systems".