Deep Image Clustering with Category-Style Representation

Junjie Zhao[1]*, Donghuan Lu[2]*, Kai Ma[2], Yu Zhang[1]†, and Yefeng Zheng[2]†

[1] School of Computer Science and Engineering, Southeast University, Nanjing, China
{kamij.zjj,zhang_yu}@seu.edu.cn
[2] Tencent Jarvis Lab, Shenzhen, China
{caleblu,kylekma,yefengzheng}@tencent.com

Abstract. Deep clustering which adopts deep neural networks to obtain optimal representations for clustering has been widely studied recently. In this paper, we propose a novel deep image clustering framework to learn a category-style latent representation in which the category information is disentangled from image style and can be directly used as the cluster assignment. To achieve this goal, mutual information maximization is applied to embed relevant information in the latent representation. Moreover, augmentation-invariant loss is employed to disentangle the representation into category part and style part. Last but not least, a prior distribution is imposed on the latent representation to ensure the elements of the category vector can be used as the probabilities over clusters. Comprehensive experiments demonstrate that the proposed approach outperforms state-of-the-art methods significantly on five public datasets.‡

Keywords: Image clustering, Deep learning, Unsupervised learning

1 Introduction

Clustering is a widely used technique in many fields, such as machine learning, data mining and statistical analysis. It aims to group objects 'similar' to each other into the same set and 'dissimilar' ones into different sets. Unlike supervised learning methods, clustering approaches should be oblivious to ground truth labels. Conventional methods, such as K-means [23] and spectral clustering [25], require feature extraction to convert data to a more discriminative form. Domain knowledge could be useful to determine more appropriate feature extraction strategies in some cases. But for many high-dimensional problems (*e.g.* images), manually designed feature extraction methods can easily lead to inferior performance.

*Equal contribution and the work was done at Tencent Jarvis Lab.

†Y. Zhang and Y. Zheng are the corresponding authors.

‡Project address: https://github.com/sKamiJ/DCCS.

Because of the powerful capability of deep neural networks to learn non-linear mapping, a lot of deep learning based clustering methods have been proposed recently. Many studies attempt to combine deep neural networks with various kinds of clustering losses [33,13,9] to learn more discriminative yet low-dimensional latent representations. To avoid trivially learning some arbitrary representations, most of those methods also minimize a reconstruction [13] or generative [24] loss as an additional regularization. However, there is no substantial connection between the discriminative ability and the generative ability of the latent representation. The aforementioned regularization turns out to be less relevant to clustering and forces the latent representation to contain unnecessary generative information, which makes the network hard to train and could also affect the clustering performance.

In this paper, instead of using a decoder/generator to minimize the reconstruction/generative loss, we use a discriminator to maximize the mutual information [14] between input images and their latent representations in order to retain discriminative information. To further reduce the effect of irrelevant information, the latent representation is divided into two parts, *i.e.*, the category (or cluster) part and the style part, where the former one contains the distinct identities of images (inter-class difference) while the latter one represents style information (intra-class difference). Specifically, we propose to use data augmentation to disentangle the category representation from style information, based on the observation [31,16] that appropriate augmentation should not change the image category.

Moreover, many deep clustering methods require additional operations [33,29] to group the latent representation into different categories. But their distance metrics are usually predefined and may not be optimal. In this paper, we impose a prior distribution [24] on the latent representation to make the category part closer to the form of a one-hot vector, which can be directly used to represent the probability distribution of the clusters.

In summary, we propose a novel approach, **D**eep **C**lustering with **C**ategory-**S**tyle representation (DCCS) for unsupervised image clustering. The main contributions of this study are four folds:

- We propose a novel end-to-end deep clustering framework to learn a latent category-style representation whose values can be used directly for the cluster assignment.
- We show that maximizing the mutual information is enough to prevent the network from learning arbitrary representations in clustering.
- We propose to use data augmentation to disentangle the category representation (inter-class difference) from style information (intra-class difference).
- Comprehensive experiments demonstrate that the proposed DCCS approach outperforms state-of-the-art methods on five commonly used datasets, and the effectiveness of each part of the proposed method is evaluated and discussed in thorough ablation studies.

2 Related Work

In recent years, many deep learning based clustering methods have been proposed. Most approaches [13,9,36] combined autoencoder [2] with traditional clustering methods by minimizing reconstruction loss as well as clustering loss. For example, Jiang *et al.* [17] combined a variational autoencoder (VAE) [19] for representation learning with a Gaussian mixture model for clustering. Yang *et al.* [35] also adopted the Gaussian mixture model as the prior in VAE, and incorporated a stochastic graph embedding to handle data with complex spread. Although the usage of the reconstruction loss can embed the sample information into the latent space, the encoded latent representation may not be optimal for clustering.

Other than autoencoder, Generative Adversarial Network (GAN) [10] has also been employed for clustering [5]. In ClusterGAN [24], Mukherjee *et al.* also imposed a prior distribution on the latent representation, which was a mixture of one-hot encoded variables and continuous latent variables. Although their representations share a similar form to ours, their one-hot variables cannot be used as the cluster assignment directly due to the lack of proper disentanglement. Additionally, ClusterGAN consisted of a generator (or a decoder) to map the random variables from latent space to image space, a discriminator to ensure the generated samples close to real images and an encoder to map the images back to the latent space to match the random variables. Such a GAN model is known to be hard to train and brings irrelevant generative information to the latent representations. To reduce the complexity of the network and avoid unnecessary generative information, we directly train an encoder by matching the aggregated posterior distribution of the latent representation to the prior distribution.

To avoid the usage of additional clustering, some methods directly encoded images into latent representations whose elements can be treated as the probabilities over clusters. For example, Xu *et al.* [16] maximized pair-wise mutual information of the latent representations extracted from an image and its augmented version. This method achieved good performance on both image clustering and segmentation, but its batch size must be large enough (more than 700 in their experiments) so that samples from different clusters were almost equally distributed in each batch. Wu *et al.* [31] proposed to learn one-hot representations by exploring various correlations of the unlabeled data, such as local robustness and triple mutual information. However, their computation of mutual information required pseudo-graph to determine whether images belonged to the same category, which may not be accurate due to the unsupervised nature of clustering. To avoid this issue, we maximized the mutual information between images and their own representations instead of representations encoded from images with the same predicted category.

3 Method

As stated in the introduction, the proposed DCCS approach aims to find an appropriate encoder Q to convert the input image X into a latent representation

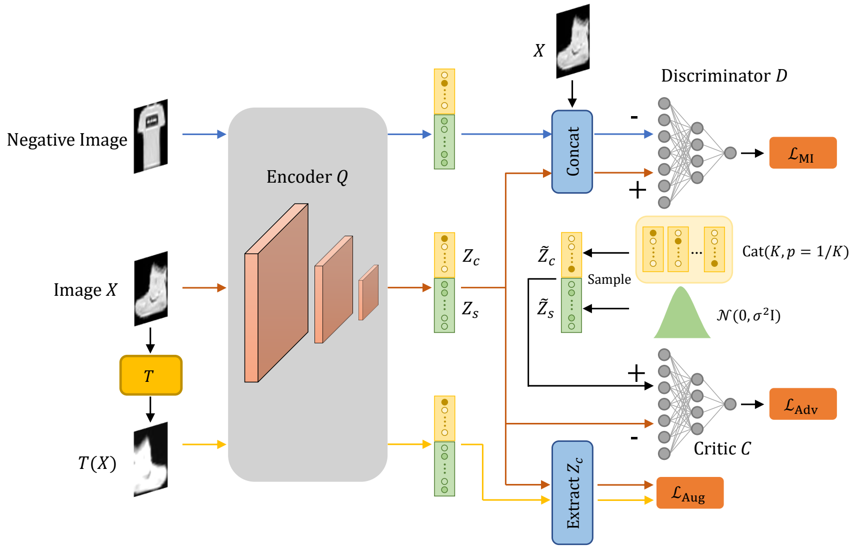

X
Discriminator D
Negative Image
Encoder Q
Concat
-
+
$\mathcal{L}_{\mathrm{MI}}$
Image X
Z_c
Z_s
$\tilde{Z}_c$
$\tilde{Z}_s$
Sample
$\mathrm{Cat}(K, p = 1/K)$
$\mathcal{N}(0, \sigma^2 \mathrm{I})$
T
+
-
$\mathcal{L}_{\mathrm{Adv}}$
Critic C
$T(X)$
Extract Z_c
$\mathcal{L}_{\mathrm{Aug}}$

Fig. 1: The overall framework of the proposed DCCS method. The encoder Q converts the image X into a latent representation $Z = (Z_c, Z_s)$. The discriminator D maximizes the mutual information between X and Z, while the critic C imposes a prior distribution on Z to make Z_c closer to the form of a one-hot vector and constrain the range of Z_s. Z_c is also regularized to be invariant to data augmentation T

Z, which consists of disentangled category and style information. To be more precise, the encoded latent representation Z consists of a softmax-activated vector Z_c and a linear-activated vector Z_s, *i.e.*, $Z = (Z_c, Z_s)$, where Z_c represents the probabilities of X belonging to each class and Z_s represents the intra-class style information. To achieve this, three regularization strategies are applied to constrain the latent representation as detailed in the following three sections, and the framework is shown in Fig. 1. To clarify notations, we use upper case letters (*e.g.* X) for random variables, lower case letters (*e.g.* x) for their values and calligraphic letters (*e.g.* $\mathcal{X}$) for sets. The probability distributions are denoted with upper case letters (*e.g.* $P(X)$), and the corresponding densities are denoted with lower case letters (*e.g.* $p(x)$).

3.1 Maximize Mutual Information

Because of the powerful capability to fit training data with complex non-linear transformations, the encoder of deep neural networks can easily map input images to arbitrary representations if without proper constraints thus losing relevant information for proceeding the target clustering task. To retain the essential information of each image and learn better discriminative latent representations, a discriminator D is introduced to maximize the mutual information $I(X, Z)$

between the input image X and its encoded latent representation Z. Based on information theory, $I(X,Z)$ takes the following form:

$$I(X,Z) = \iint q(z|x)p_X(x)\log\frac{q(z|x)p_X(x)}{q_Z(z)p_X(x)}dxdz \tag{1}$$

$$= \mathrm{KL}(Q(Z|X)P_X(X)\|Q_Z(Z)P_X(X)) \tag{2}$$

where $Q(Z|X)$ is the encoding distribution, P_X is the prior distribution of the images, $Q_Z = \mathbb{E}_{P_X}[Q(Z|X)]$ is the aggregated posterior distribution of the latent representation and $\mathrm{KL}(\cdot\|\cdot)$ is the KL-divergence. In this original formulation, however, KL-divergence is unbounded and maximizing it may lead to an unstable training result. Following [14], we replace KL-divergence with JS-divergence to estimate the mutual information:

$$I^{(\mathrm{JSD})}(X,Z) = \mathrm{JS}(Q(Z|X)P_X(X), Q_Z(Z)P_X(X)). \tag{3}$$

According to [27,10], JS-divergence between two arbitrary distributions $P(X)$ and $Q(X)$ can be estimated by a discriminator D:

$$\begin{aligned}\mathrm{JS}(P(X),Q(X)) = \frac{1}{2}\max_D\{&\mathbb{E}_{X\sim P(X)}[\log S(D(X))]\\ &+\mathbb{E}_{X\sim Q(X)}[\log(1-S(D(X)))]\}+\log 2\end{aligned} \tag{4}$$

where S is the sigmoid function. Replacing $P(X)$ and $Q(X)$ with $Q(Z|X)P_X(X)$ and $Q_Z(Z)P_X(X)$, the mutual information can be maximized by:

$$\begin{aligned}\frac{1}{2}\max_{Q,D}\{&\mathbb{E}_{(X,Z)\sim Q(Z|X)P_X(X)}[\log S(D(X,Z))]\\ &+\mathbb{E}_{(X,Z)\sim Q_Z(Z)P_X(X)}[\log(1-S(D(X,Z)))]\}+\log 2.\end{aligned} \tag{5}$$

Accordingly, the mutual information loss function can be defined as:

$$\begin{aligned}\mathcal{L}_{\mathrm{MI}} = -(&\mathbb{E}_{(X,Z)\sim Q(Z|X)P_X(X)}[\log S(D(X,Z))]\\ &+\mathbb{E}_{(X,Z)\sim Q_Z(Z)P_X(X)}[\log(1-S(D(X,Z)))])\end{aligned} \tag{6}$$

where Q and D are jointly optimized.

With the concatenation of X and Z as input, minimizing $\mathcal{L}_{\mathrm{MI}}$ can be interpreted as to determine whether X and Z are correlated. For discriminator D, an image X along with its representation is a positive sample while X along with a representation encoded from another image is a negative sample. As aforementioned, many deep clustering methods use the reconstruction loss or generative loss to avoid arbitrary representations. However, it allows the encoded representation to contain unnecessary generative information and makes the network, especially GAN, hard to train. The mutual information maximization only instills necessary discriminative information into the latent space and experiments in Section 4 confirm that it leads to better performance.

3.2 Disentangle Category-Style Information

As previously stated, we expect the latent category representation Z_c only contains the categorical cluster information while all the style information is represented by Z_s. To achieve such a disentanglement, an augmentation-invariant regularization term is introduced based on the observation that certain augmentation should not change the category of images.

Specifically, given an augmentation function T which usually includes geometric transformations (*e.g.* scaling and flipping) and photometric transformations (*e.g.* changing brightness or contrast), Z_c and Z'_c encoded from X and $T(X)$ should be identical while all the appearance differences should be represented by the style variables. Because the elements of Z_c represent the probabilities over clusters, the KL-divergence is adopted to measure the difference between $Q(Z_c|X)$ and $Q(Z_c|T(X))$. The augmentation-invariant loss function for the encoder Q can be defined as:

$$\mathcal{L}_{\text{Aug}} = \text{KL}(Q(Z_c|X) \| Q(Z_c|T(X))). \tag{7}$$

3.3 Match to Prior Distribution

There are two potential issues with the aforementioned regularization terms: the first one is that the category representation cannot be directly used as the cluster assignment, therefore additional operations are still required to determine the clustering categories; the second one is that the augmentation-invariant loss may lead to a degenerate solution, *i.e.*, assigning all images into a few clusters, or even the same cluster. In order to resolve these issues, a prior distribution P_Z is imposed on the latent representation Z.

Following [24], a categorical distribution $\tilde{Z}_c \sim \text{Cat}(K, p = 1/K)$ is imposed on Z_c, where $\tilde{Z}_c$ is a one-hot vector and K is the number of categories that the images should be clustered into. A Gaussian distribution $\tilde{Z}_s \sim \mathcal{N}(0, \sigma^2\mathbf{I})$ (typically $\sigma = 0.1$) is imposed on Z_s to constrain the range of style variables.

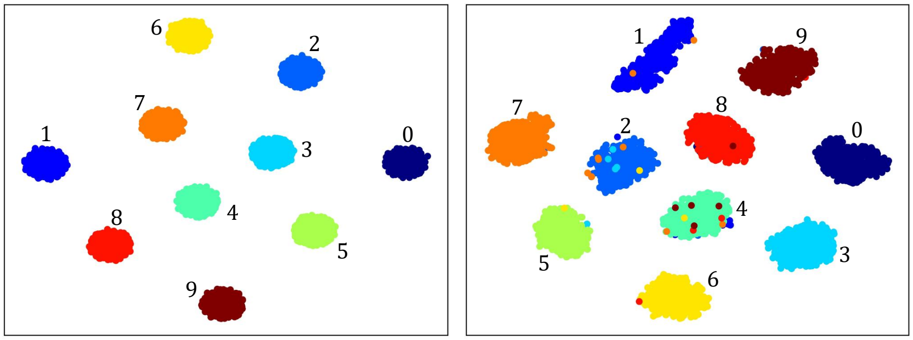
As aforementioned, ClusterGAN [24] uses the prior distribution to generate random variables, applies a GAN framework to train a proper decoder and then learns an encoder to match the decoder. To reduce the complexity of the network and avoid unnecessary generative information, we directly train the encoder by matching the aggregated posterior distribution $Q_Z = \mathbb{E}_{P_X}[Q(Z|X)]$ to the prior distribution P_Z. Experiments demonstrate that such a strategy can lead to better clustering performance.

To impose the prior distribution P_Z on Z, we minimize the Wasserstein distance [1] $W(Q_Z, P_Z)$ between Q_Z and P_Z, which can be estimated by:

$$\max_{C \in \mathcal{C}} \{\mathbb{E}_{\tilde{Z} \sim P_Z}[C(\tilde{Z})] - \mathbb{E}_{Z \sim Q_Z}[C(Z)]\} \tag{8}$$

where $\mathcal{C}$ is the set of 1-Lipschitz functions. Under the optimal critic C (denoted as *discriminator* in vanilla GAN), minimizing Eq. 8 with respect to the encoder parameters also minimizes $W(Q_Z, P_Z)$:

$$\min_{Q} \max_{C \in \mathcal{C}} \{\mathbb{E}_{\tilde{Z} \sim P_Z}[C(\tilde{Z})] - \mathbb{E}_{Z \sim Q_Z}[C(Z)]\}. \tag{9}$$



(a)

(b)

Fig. 2: The t-SNE visualization of the latent representations of MNIST dataset. The dimensions of Z_c and Z_s are set as 10 and 50, respectively. Here, (a) shows the prior representation $\tilde{Z}$ sampled from P_Z, the numbers 0-9 represent different categories, while (b) demonstrates the encoded representation Z. Each point represents a latent representation and the color refers to its ground truth label

For optimization, the gradient penalty [12] is introduced to enforce the Lipschitz constraint on the critic. The adversarial loss functions for the encoder Q and the critic C can be defined as:

$$\mathcal{L}_{\text{Adv}}^{Q} = -\mathbb{E}_{Z\sim Q_Z}[C(Z)] \tag{10}$$

$$\mathcal{L}_{\text{Adv}}^{C} = \mathbb{E}_{Z\sim Q_Z}[C(Z)] - \mathbb{E}_{\tilde{Z}\sim P_Z}[C(\tilde{Z})] + \lambda\mathbb{E}_{\hat{Z}\sim P_{\hat{Z}}}[(\|\nabla_{\hat{Z}}C(\hat{Z})\|_2 - 1)^2] \tag{11}$$

where λ is the gradient penalty coefficient, $\hat{Z}$ is the latent representation sampled uniformly along the straight lines between pairs of latent representations sampled from Q_Z and P_Z, and $(\|\nabla_{\hat{Z}}C(\hat{Z})\|_2 - 1)^2$ is the one-centered gradient penalty. Q and C are optimized alternatively.

Note that the reason why we use Wasserstein distance instead of f-divergence is that Wasserstein distance is continuous everywhere and differentiable almost everywhere. Such a critic is able to provide meaningful gradients for the encoder even with an input containing discrete variables. On the other hand, the loss of the critic can be viewed as an estimation of $W(Q_Z, P_Z)$ to determine whether the clustering progress has converged or not, as shown in Section 4.2.

Fig. 2a shows the t-SNE [22] visualization of the prior representation $\tilde{Z} = (\tilde{Z}_c, \tilde{Z}_s)$ with points being colored based on $\tilde{Z}_c$. It shows that the representations sampled from the prior distribution can be well clustered based on $\tilde{Z}_c$ while $\tilde{Z}_s$ represents the intra-class difference. After imposing the prior distribution on Z as displayed in Fig. 2b, the encoded latent representations show a similar pattern as the prior representations, therefore the cluster assignment can be easily achieved by using argmax over Z_c.

Algorithm 1 Deep Clustering with Category-Style Representation

```
Input: Dataset $\mathcal{X} = \{x^i\}_{i=1}^N$, $\theta_Q, \theta_D, \theta_C$ initial parameters of encoder $Q$, discriminator
    $D$ and critic $C$, the dimensions of $Z_c$ and $Z_s$, hyper-parameters $\sigma$, $\lambda$, $\beta_{\text{MI}}$, $\beta_{\text{Aug}}$,
    $\beta_{\text{Adv}}$, augmentation function $T$, the number of critic iterations per encoder iteration
    $n_{\text{critic}}$, batch size $m$.
while $\mathcal{L}^C$ not converged do
    for $t = 1, ..., n_{\text{critic}}$ do
        Sample $\{x^i\}_{i=1}^m$ from $\mathcal{X}$, $\{\tilde{z}^i\}_{i=1}^m$ from $P_Z$, $\{\epsilon^i\}_{i=1}^m$ from $U[0,1]$;
        Sample $z^i$ from $Q(Z|X = x^i)$ for $i = 1, ..., m$;
        Compute $\hat{z}^i = \epsilon^i z^i + (1 - \epsilon^i)\tilde{z}^i$ for $i = 1, ..., m$;
        Update $\theta_C$ by minimizing $\mathcal{L}^C$ (Eq. 14);
    end for
    Sample $\{x^i\}_{i=1}^m$ from $\mathcal{X}$;
    Sample $z'^i = (z'^i_c, z'^i_s)$ from $Q(Z|X = T(x^i))$ for $i = 1, ..., m$;
    Sample $z^i = (z^i_c, z^i_s)$ from $Q(Z|X = x^i)$ for $i = 1, ..., m$;
    Sample $z^j$ from $\{z^i\}_{i=1}^m$ for each $x^i$ to form negative paris;
    Update $\theta_Q$ and $\theta_D$ by minimizing $\mathcal{L}^Q$ (Eq. 12) and $\mathcal{L}^D$ (Eq. 13);
end while
for $i = 1, ..., N$ do
    Sample $z^i = (z^i_c, z^i_s)$ from $Q(Z|X = x^i)$;
    Compute cluster assignment $l^i = \text{argmax}(z^i_c)$;
end for
Output: Cluster assignment $\{l^i\}_{i=1}^N$.
```

3.4 The Unified Model

As shown in Fig. 1, the network of DCCS consists of three parts: the encoder Q to convert images into latent representations, the discriminator D to maximize the mutual information and the critic C to impose the prior distribution. The objective functions for encoder Q, discriminator D and critic C are defined as:

$$\mathcal{L}^Q = \beta_{\text{MI}}\mathcal{L}_{\text{MI}} + \beta_{\text{Aug}}\mathcal{L}_{\text{Aug}} + \beta_{\text{Adv}}\mathcal{L}^Q_{\text{Adv}} \tag{12}$$

$$\mathcal{L}^D = \beta_{\text{MI}}\mathcal{L}_{\text{MI}} \tag{13}$$

$$\mathcal{L}^C = \beta_{\text{Adv}}\mathcal{L}^C_{\text{Adv}} \tag{14}$$

where β_{MI}, β_{Aug} and β_{Adv} are the weights used to balance each term.

As described in Algorithm 1, the parameters of Q and D are jointly updated while the parameters of C are trained separately. Note that because Q is a deterministic encoder, *i.e.*, $Q(Z|X = x) = \delta_{\mu(x)}$, where δ denotes Dirac-delta and $\mu(x)$ is a deterministic mapping function, sampling z^i from $Q(Z|X = x^i)$ is equivalent to assign z^i with $\mu(x^i)$.

4 Experiments

4.1 Experimental Settings

Datasets. We evaluate the proposed DCCS on five commonly used datasets, including MNIST [21], Fashion-MNIST [32], CIFAR-10 [20], STL-10 [6] and

Table 1: Statistics of the datasets

Dataset	Images	Clusters	Image size
MNIST [21]	70000	10	28×28
Fashion-MNIST [32]	70000	10	28×28
CIFAR-10 [20]	60000	10	$32 \times 32 \times 3$
STL-10 [6]	13000	10	$96 \times 96 \times 3$
ImageNet-10 [4]	13000	10	$96 \times 96 \times 3$

ImageNet-10 [4]. The statistics of these datasets are described in Table 1. As a widely adopted setting [33,4,31], the training and test sets of these datasets are jointly utilized. For STL-10, the unlabelled subset is not used. For ImageNet-10, images are selected from the ILSVRC2012 1K dataset [7] the same as in [4] and resized to 96 × 96 pixels. Similar to the IIC approach [16], color images are converted to grayscale to discourage clustering based on trivial color cues.

Evaluation metrics. Three widely used metrics are applied to evaluate the performance of the clustering methods, including unsupervised clustering accuracy (ACC), normalized mutual information (NMI), and adjusted rand index (ARI) [31]. For these metrics, a higher score implies better performance.

Implementation details. The architectures of encoders are similar to [24,12] with a different number of layers and units being used for different sizes of images. The critic and discriminator are multi-layer perceptions. All the parameters are randomly initialized without pretraining. The Adam [18] optimizer with a learning rate of 10^{-4} and $\beta_1 = 0.5$, $\beta_2 = 0.9$ is used for optimization. The dimension of Z_s is set to 50, and the dimension of Z_c is set to the expected number of clusters. For other hyper-parameters, we set the standard deviation of prior Gaussian distribution $\sigma = 0.1$, the gradient penalty coefficient $\lambda = 10$, $\beta_{\mathrm{MI}} = 0.5$, $\beta_{\mathrm{Adv}} = 1$, the number of critic iterations per encoder iteration $n_{\mathrm{critic}} = 4$, and batch size $m = 64$ for all datasets. Because β_{Aug} is related to the datasets and generally the more complex the images are, the larger β_{Aug} should be. We come up with an applicable way to set β_{Aug} by visualizing the t-SNE figure of the encoded representation Z, *i.e.*, β_{Aug} is gradually increased until the clusters visualized by t-SNE start to overlap. With this method, β_{Aug} is set to 2 for MNIST and Fashion-MNIST, and set to 4 for other datasets. The data augmentation includes four commonly used approaches, *i.e.*, random cropping, random horizontal flipping, color jittering and channel shuffling (which is used on the color images before graying). For more details about network architectures, data augmentation and hyper-parameters, please refer to the supplementary materials.

4.2 Main Result

Quantitative comparison. We first compare the proposed DCCS with several baseline methods as well as other state-of-the-art clustering approaches based

Table 2: Comparison with baseline and state-of-the-art methods on MNIST and Fashion-MNIST. The best three results of each metric are highlighted in **bold**. $\star$: Re-implemented results with the released code

Method	MNIST			Fashion-MNIST		
	ACC	NMI	ARI	ACC	NMI	ARI
K-means [30]	0.572	0.500	0.365	0.474	0.512	0.348
SC [37]	0.696	0.663	0.521	0.508	0.575	-
AC [11]	0.695	0.609	0.481	0.500	0.564	0.371
NMF [3]	0.545	0.608	0.430	0.434	0.425	-
DEC [33]	0.843	0.772	0.741	0.590*	0.601*	0.446*
JULE [34]	0.964	0.913	0.927	0.563	0.608	-
VaDE [17]	0.945	0.876	-	0.578	0.630	-
DEPICT [9]	0.965	0.917	-	0.392	0.392	-
IMSAT [15]	**0.984**	**0.956***	**0.965***	**0.736***	**0.696***	**0.609***
DAC [4]	0.978	0.935	0.949	0.615*	0.632*	0.502*
SpectralNet [29]	0.971	0.924	0.936*	0.533*	0.552*	-
ClusterGAN [24]	0.950	0.890	0.890	0.630	0.640	0.500
DLS-Clustering [8]	0.975	0.936	-	0.693	0.669	-
DualAE [36]	0.978	0.941	-	0.662	0.645	-
RTM [26]	0.968	0.933	0.932	0.710	0.685	0.578
NCSC [38]	0.941	0.861	0.875	**0.721**	**0.686**	**0.592**
IIC [16]	**0.992**	**0.978***	**0.983***	0.657*	0.637*	0.523*
DCCS (Proposed)	**0.989**	**0.970**	**0.976**	**0.756**	**0.704**	**0.623**

on deep learning, as shown in Table 2 and Table 3. DCCS outperforms all the other methods by large margins on Fashion-MNIST, CIFAR-10, STL-10 and ImageNet-10. For the ACC metric, DCCS is 2.0%, 3.3%, 3.7% and 2.7% higher than the second best methods on these four datasets, respectively. Although for MNIST, the clustering accuracy of DCCS is slightly lower (*i.e.*, 0.3%) than IIC [16], DCCS significantly surpasses IIC on CIFAR-10 and STL-10.

Training progress. The training progress of the proposed DCCS is monitored by minimizing the Wasserstein distance $W(Q_Z, P_Z)$, which can be estimated by the negative critic loss $-\mathcal{L}^C$. As plotted in Fig. 3, the critic loss stably converges and it correlates well with the clustering accuracy, demonstrating a robust training progress. The visualizations of the latent representations with t-SNE at three different stages are also displayed in Fig. 3. From stage A to C, the latent representations gradually cluster together while the critic loss decreases steadily.

Qualitative analysis. Fig. 4 shows images with top 10 predicted probabilities from each cluster in MNIST and ImageNet-10. Each row corresponds to a cluster and the images from left to right are sorted in a descending order based on their probabilities. In each row of Fig. 4a, the same digits are written in different ways, indicating that Z_c contains well disentangled category information. For

Table 3: Comparison with baseline and state-of-the-art methods on CIFAR-10, STL-10 and ImageNet-10. The best three results of each metric are highlighted in **bold**. $\star$: Re-implemented results with the released code. †: The results are evaluated on STL-10 without using the unlabelled data subset

Method	CIFAR-10			STL-10			ImageNet-10		
	ACC	NMI	ARI	ACC	NMI	ARI	ACC	NMI	ARI
K-means [30]	0.229	0.087	0.049	0.192	0.125	0.061	0.241	0.119	0.057
SC [37]	0.247	0.103	0.085	0.159	0.098	0.048	0.274	0.151	0.076
AC [11]	0.228	0.105	0.065	0.332	0.239	0.140	0.242	0.138	0.067
NMF [3]	0.190	0.081	0.034	0.180	0.096	0.046	0.230	0.132	0.065
AE [2]	0.314	0.239	0.169	0.303	0.250	0.161	0.317	0.210	0.152
GAN [28]	0.315	0.265	0.176	0.298	0.210	0.139	0.346	0.225	0.157
VAE [19]	0.291	0.245	0.167	0.282	0.200	0.146	0.334	0.193	0.168
DEC [33]	0.301	0.257	0.161	0.359	0.276	0.186	0.381	0.282	0.203
JULE [34]	0.272	0.192	0.138	0.277	0.182	0.164	0.300	0.175	0.138
DAC [4]	0.522	0.396	0.306	0.470	0.366	0.257	**0.527**	**0.394**	**0.302**
IIC [16]	**0.617**	**0.513***	**0.411***	**0.499**†	**0.431***†	**0.295***†	-	-	-
DCCM [31]	**0.623**	**0.496**	**0.408**	**0.482**	**0.376**	**0.262**	**0.710**	**0.608**	**0.555**
DCCS (Proposed)	**0.656**	**0.569**	**0.469**	**0.536**	**0.490**	**0.362**	**0.737**	**0.640**	**0.560**

Table 4: Evaluation of different ways for the cluster assignment

Method	MNIST			Fashion-MNIST			CIFAR-10			STL-10		
	ACC	NMI	ARI	ACC	NMI	ARI	ACC	NMI	ARI	ACC	NMI	ARI
Argmax over Z_c	**0.9891**	**0.9696**	**0.9758**	**0.7564**	0.7042	**0.6225**	**0.6556**	**0.5692**	0.4685	**0.5357**	**0.4898**	**0.3617**
K-means on Z	**0.9891**	0.9694	0.9757	**0.7564**	**0.7043**	0.6224	0.6513	0.5588	**0.4721**	0.5337	0.4888	0.3599
K-means on Z_c	**0.9891**	**0.9696**	0.9757	0.7563	0.7042	0.6223	0.6513	0.5587	**0.4721**	0.5340	0.4889	0.3602
K-means on Z_s	0.5164	0.4722	0.3571	0.4981	0.4946	0.3460	0.2940	0.1192	0.0713	0.4422	0.4241	0.2658

ImageNet-10 in Fig. 4b, most objects are well clustered and the major confusion is for the airships and airplanes in the sky due to their similar shapes and backgrounds (Row 8). A possible solution is overclustering, *i.e.*, more number of clusters than expected, which requires investigation in future work.

4.3 Ablation Study

Cluster assignment w/o K-means. As stated in Section 3.3, by imposing the prior distribution, the latent category representation Z_c can be directly used as the cluster assignment. Table 4 compares the results of several ways to obtain the cluster assignment with the same encoder. We can see that using Z_c with or without K-means has similar performance, indicating that Z_c is discriminative enough to be used as the cluster assignment directly. Additional experiments on each part of Z show that applying K-means on Z_c can yield similar performance as on Z, while the performance of applying K-means on Z_s is much worse. It demonstrates that the categorical cluster information and the style information are well disentangled as expected.

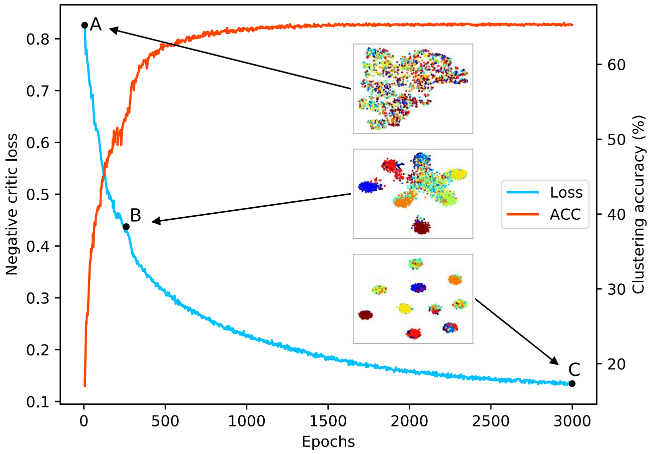


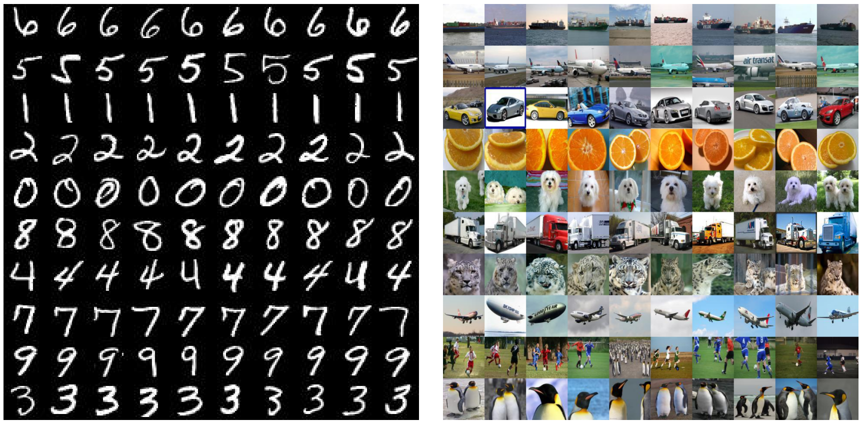
Fig. 3: Training curves of the negative critic loss and the clustering accuracy on CIFAR-10. The t-SNE visualizations of the latent representations Z for different stages are also displayed. The color of the points in the t-SNE visualizations refers to the ground truth category

Table 5: Ablation study of DCCS on Fashion-MNIST and CIFAR-10

Method	Loss			Fashion-MNIST			CIFAR-10		
	$\mathcal{L}_{\mathrm{Adv}}$	$\mathcal{L}_{\mathrm{MI}}$	$\mathcal{L}_{\mathrm{Aug}}$	ACC	NMI	ARI	ACC	NMI	ARI
M1	✓			0.618	0.551	0.435	0.213	0.076	0.040
M2	✓	✓		0.692	0.621	0.532	0.225	0.085	0.048
M3	✓		✓	0.725	0.694	0.605	0.645	0.557	0.463
M4	✓	✓	✓	**0.756**	**0.704**	**0.623**	**0.656**	**0.569**	**0.469**

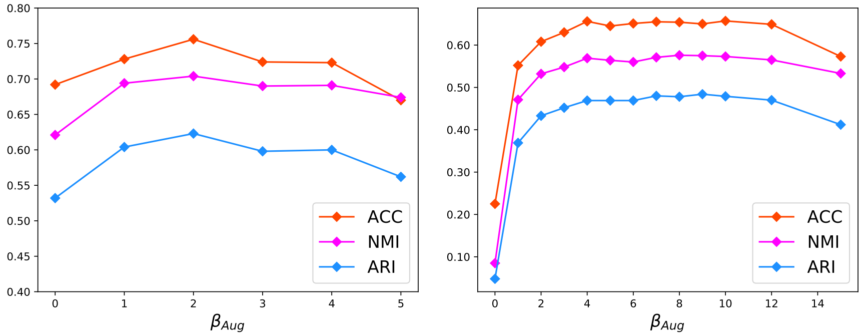
Ablation study on the losses. The effectiveness of the losses is evaluated in Table 5. M1 is the baseline method, *i.e.*, the only constraint applied to the network is the prior distribution. This constraint is always necessary to ensure that the category representation can be directly used as the cluster assignment. By adding the mutual information maximization in M2 or the category-style information disentanglement in M3, the clustering performance achieves significant gains. The results of M4 demonstrate that combining all three losses can further improve the clustering performance. Note that large improvement with data augmentation for CIFAR-10 is due to that the images in CIFAR-10 have considerable intra-class variability, therefore disentangling the category-style information can improve the clustering performance by a large margin.

Impact of $\boldsymbol{\beta}_{\mathbf{Aug}}$. The clustering performance with different β_{Aug}, which is the weight of the data augmentation loss in Eq. 12, is displayed in Fig. 5. For Fashion-



(a) (b)

Fig. 4: Clustering images from MNIST (a) and ImageNet-10 (b). Each row contains the images with the highest probability to belong to the respective cluster

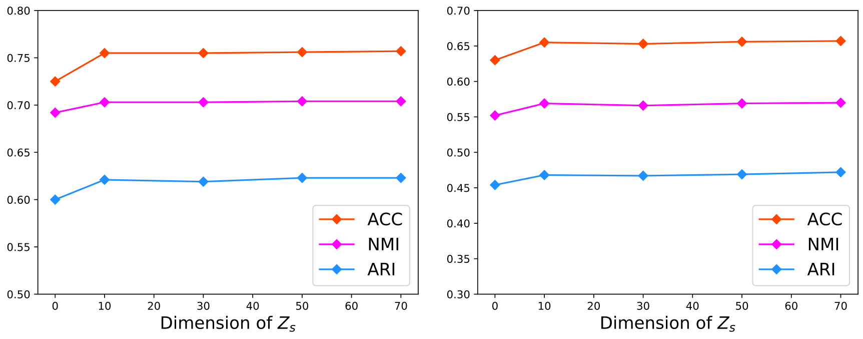


(a) (b)

Fig. 5: The impact of β_{Aug} on Fashion-MNIST (a) and CIFAR-10 (b)

MNIST, the performance drops when β_{Aug} is either too small or too large because a small β_{Aug} cannot disentangle the style information enough, and a large β_{Aug} may lead the clusters to overlap. For CIFAR-10, the clustering performance is relatively stable with large β_{Aug}. As previously stated, the biggest β_{Aug} without overlapping clusters in the t-SNE visualization of the encoded representation Z is selected (the visualization of t-SNE can be found in the supplementary materials).

Impact of Z_s. As shown in Fig. 6, varying the dimension of Z_s from 10 to 70 does not affect the clustering performance much. However, when the dimen-



(a) (b)

Fig. 6: The impact of Z_s on Fashion-MNIST (a) and CIFAR-10 (b)

sion of Z_s is 0, *i.e.*, the latent representation only contains the category part, the performance drops a lot, demonstrating the necessity of the style representation. The reason is that for the mutual information maximization, the category representation alone is not enough to describe the difference among images belonging to the same cluster.

5 Conclusions

In this work, we proposed a novel unsupervised deep image clustering framework with three regularization strategies. First, mutual information maximization was applied to retain essential information and avoid arbitrary representations. Furthermore, data augmentation was employed to disentangle the category representation from style information. Finally, a prior distribution was imposed to prevent degenerate solutions and avoid the usage of additional clustering so that the category representation could be used directly as the cluster assignment. Ablation studies demonstrated the effectiveness of each component and the extensive comparison experiments on five datasets showed that the proposed approach outperformed other state-of-the-art methods.

Acknowledgements

This work was supported by National Key Research and Development Program of China (No. 2018AAA0100100), National Natural Science Foundation of China (61702095), the Key Area Research and Development Program of Guangdong Province, China (No. 2018B010111001), National Key Research and Development Project (2018YFC2000702), Science and Technology Program of Shenzhen, China (No. ZDSYS201802021814180) and the Big Data Computing Center of Southeast University.

References

1. Arjovsky, M., Chintala, S., Bottou, L.: Wasserstein GAN. arXiv preprint arXiv:1701.07875 (2017)
2. Bengio, Y., Lamblin, P., Popovici, D., Larochelle, H.: Greedy layer-wise training of deep networks. In: Advances in Neural Information Processing Systems. pp. 153–160 (2007)
3. Cai, D., He, X., Wang, X., Bao, H., Han, J.: Locality preserving nonnegative matrix factorization. In: International Joint Conference on Artificial Intelligence (2009)
4. Chang, J., Wang, L., Meng, G., Xiang, S., Pan, C.: Deep adaptive image clustering. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 5879–5887 (2017)
5. Chen, X., Duan, Y., Houthooft, R., Schulman, J., Sutskever, I., Abbeel, P.: InfoGAN: Interpretable representation learning by information maximizing generative adversarial nets. In: Advances in Neural Information Processing Systems. pp. 2172–2180 (2016)
6. Coates, A., Ng, A., Lee, H.: An analysis of single-layer networks in unsupervised feature learning. In: Proceedings of the International Conference on Artificial Intelligence and Statistics. pp. 215–223 (2011)
7. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: ImageNet: A large-scale hierarchical image database. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 248–255. IEEE (2009)
8. Ding, F., Luo, F.: Clustering by directly disentangling latent space. arXiv preprint arXiv:1911.05210 (2019)
9. Ghasedi Dizaji, K., Herandi, A., Deng, C., Cai, W., Huang, H.: Deep clustering via joint convolutional autoencoder embedding and relative entropy minimization. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 5736–5745 (2017)
10. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: Advances in Neural Information Processing Systems. pp. 2672–2680 (2014)
11. Gowda, K.C., Krishna, G.: Agglomerative clustering using the concept of mutual nearest neighbourhood. Pattern Recognition **10**(2), 105–112 (1978)
12. Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., Courville, A.C.: Improved training of Wasserstein GANs. In: Advances in Neural Information Processing Systems. pp. 5767–5777 (2017)
13. Guo, X., Liu, X., Zhu, E., Yin, J.: Deep clustering with convolutional autoencoders. In: International Conference on Neural Information Processing. pp. 373–382. Springer (2017)
14. Hjelm, R.D., Fedorov, A., Lavoie-Marchildon, S., Grewal, K., Bachman, P., Trischler, A., Bengio, Y.: Learning deep representations by mutual information estimation and maximization. arXiv preprint arXiv:1808.06670 (2018)
15. Hu, W., Miyato, T., Tokui, S., Matsumoto, E., Sugiyama, M.: Learning discrete representations via information maximizing self-augmented training. In: Proceedings of the International Conference on Machine Learning. pp. 1558–1567 (2017)
16. Ji, X., Henriques, J.F., Vedaldi, A.: Invariant information clustering for unsupervised image classification and segmentation. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 9865–9874 (2019)
17. Jiang, Z., Zheng, Y., Tan, H., Tang, B., Zhou, H.: Variational deep embedding: An unsupervised and generative approach to clustering. arXiv preprint arXiv:1611.05148 (2016)

18. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
19. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114 (2013)
20. Krizhevsky, A., Hinton, G., et al.: Learning multiple layers of features from tiny images. Technical Report (2009)
21. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proceedings of the IEEE **86**(11), 2278–2324 (1998)
22. Maaten, L.v.d., Hinton, G.: Visualizing data using t-SNE. Journal of Machine Learning Research **9**(Nov), 2579–2605 (2008)
23. MacQueen, J., et al.: Some methods for classification and analysis of multivariate observations. In: Proceedings of the Berkeley Symposium on Mathematical Statistics and Probability. vol. 1, pp. 281–297. Oakland, CA, USA (1967)
24. Mukherjee, S., Asnani, H., Lin, E., Kannan, S.: ClusterGAN: Latent space clustering in generative adversarial networks. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 33, pp. 4610–4617 (2019)
25. Ng, A.Y., Jordan, M.I., Weiss, Y.: On spectral clustering: Analysis and an algorithm. In: Advances in Neural Information Processing Systems. pp. 849–856 (2002)
26. Nina, O., Moody, J., Milligan, C.: A decoder-free approach for unsupervised clustering and manifold learning with random triplet mining. In: Proceedings of the Geometry Meets Deep Learning Workshop in IEEE International Conference on Computer Vision (2019)
27. Nowozin, S., Cseke, B., Tomioka, R.: f-GAN: Training generative neural samplers using variational divergence minimization. In: Advances in Neural Information Processing Systems. pp. 271–279 (2016)
28. Radford, A., Metz, L., Chintala, S.: Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv preprint arXiv:1511.06434 (2015)
29. Shaham, U., Stanton, K., Li, H., Nadler, B., Basri, R., Kluger, Y.: SpectralNet: Spectral clustering using deep neural networks. arXiv preprint arXiv:1801.01587 (2018)
30. Wang, J., Wang, J., Song, J., Xu, X.S., Shen, H.T., Li, S.: Optimized Cartesian K-Means. IEEE Transactions on Knowledge and Data Engineering **27**(1), 180–192 (2014)
31. Wu, J., Long, K., Wang, F., Qian, C., Li, C., Lin, Z., Zha, H.: Deep comprehensive correlation mining for image clustering. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 8150–8159 (2019)
32. Xiao, H., Rasul, K., Vollgraf, R.: Fashion-MNIST: A novel image dataset for benchmarking machine learning algorithms. arXiv preprint arXiv:1708.07747 (2017)
33. Xie, J., Girshick, R., Farhadi, A.: Unsupervised deep embedding for clustering analysis. In: Proceedings of the International Conference on Machine Learning. pp. 478–487 (2016)
34. Yang, J., Parikh, D., Batra, D.: Joint unsupervised learning of deep representations and image clusters. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5147–5156 (2016)
35. Yang, L., Cheung, N.M., Li, J., Fang, J.: Deep clustering by gaussian mixture variational autoencoders with graph embedding. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 6440–6449 (2019)
36. Yang, X., Deng, C., Zheng, F., Yan, J., Liu, W.: Deep spectral clustering using dual autoencoder network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4066–4075 (2019)

37. Zelnik-Manor, L., Perona, P.: Self-tuning spectral clustering. In: Advances in Neural Information Processing Systems. pp. 1601–1608 (2005)
38. Zhang, T., Ji, P., Harandi, M., Huang, W., Li, H.: Neural collaborative subspace clustering. arXiv preprint arXiv:1904.10596 (2019)