CONTRASTIVE FINE-GRAINED CLASS CLUSTERING VIA GENERATIVE ADVERSARIAL NETWORKS

Anonymous authors
Paper under double-blind review

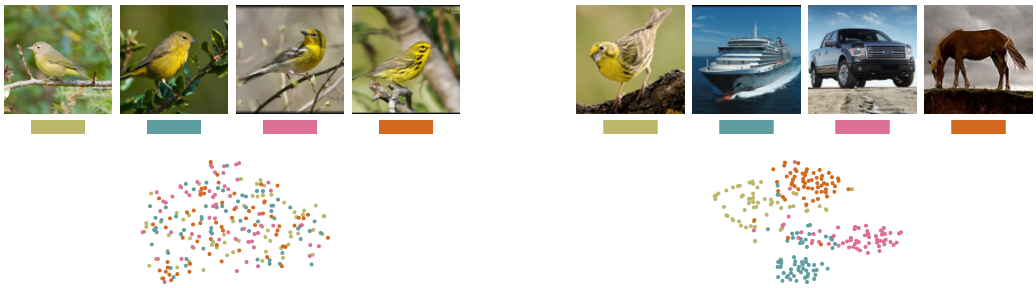
ABSTRACT

Unsupervised fine-grained class clustering is practical yet challenging due to the difficulty of feature representations learning of subtle object details. We introduce C3-GAN, a method that leverages the categorical inference power of InfoGAN with contrastive learning. We aim to learn feature representations that not only maximize the mutual information between the latent code and its observation, but also encourage the data to form distinct cluster boundaries in the embedding space. Our approach is to train the discriminator optimizing the contrastive loss, where the image-latent pairs that maximize the mutual information are considered as positive pairs and the rest as negative pairs, then we utilize it for clustering images. For this formulation, we map the latent code, which is the input of the generator, to the embedding space of the discriminator and let them act as a cluster centroid. In this way, C3-GAN achieved to learn a clustering-friendly embedding space where each cluster is distinctively separated. Experiments show that C3-GAN achieved state-of-the-art clustering performances on four fine-grained benchmark datasets, while also alleviating the mode collapse phenomenon.

1 INTRODUCTION

Unsupervised fine-grained class clustering is to group images of very similar objects (Benny & Wolf, 2020). While a line of works based on multiview-based self-supervised learning (SSL) methods (He et al., 2020; Chen et al., 2020; Grill et al., 2020) shows promising results in the conventional coarse-grained class clustering (Van Gansbeke et al., 2020) task, it is difficult for a fine-grained clustering task to benefit from these methods for two reasons. First, though it is more challenging and ambiguous for finding the distinction between fine-grained classes, datasets of the fine-grained object images are difficult to be large-scale. Figure 1 visually represents that finding distinctions between these classes are more difficult compared to the ones of the coarse-grained object classes. Second, the augmentation process of these methods consider the subtle changes in color or shape, that actually play an important role in sorting classes, as noisy factors. Thus, it is required to find an another approach for the fine-grained class clustering task.

Generative adversarial networks (GAN) (Goodfellow et al., 2014) can be a solution as it is required to learn fine details to generate realistic images. The possible starting point can be InfoGAN (Chen et al., 2016), which has achieved unsupervised categorical inference on MNIST dataset (LeCun et al., 2010) by maximizing the mutual information between the latent code and its observation. The only prior knowledge employed was the number of classes and the fact that the data is uniformly distributed over the classes. FineGAN (Singh et al., 2019) extends InfoGAN by integrating the scene decomposition learning method into the framework, and learns three latent codes that all follow the uniform categorical distribution. Each code has a hierarchical relationship with each other, and is sequentially injected to the generator for a background, a super-class object, and a sub-class object image syntheses. FineGAN demonstrates that the learnt codes for foreground image generation can also be used for clustering real images into their fine-grained classes and outperforms conventional coarse-grained class clustering methods. This result implies that extracting only the foreground part is very helpful for the given task. However, FineGAN requires bounding box annotations and additional training of classifiers, which hinders its applicability. Also, the method lacks the ability of learning the distribution of real images due to the mode collapse phenomenon (Higgins et al., 2017).



(a) Fine-grained Class Clustering

(b) Coarse-grained Class Clustering

Figure 1: **Fine-grained vs. Coarse-grained.** We compare two sets of 4 classes, each selected from (a) CUB (Wah et al., 2011) and (b) STL (Coates et al., 2011) datasets. The diagram visualizes image features extracted from ResNet-50 that is pre-trained on ImageNet-1k via the method of SimCLR (Chen et al., 2020). The color of each dot corresponds to its ground-truth class. Both in the image-level and the feature-level, the clusters in (a) are way more indistinguishable than those of (b).

In this paper, we propose **C**onstrastive fine-grained **C**lass **C**lustering **GAN** (C3-GAN) that deals with the aforementioned problems. We first remove the reliance on human-annotated labels by adopting the method inspired by PerturbGAN (Bielski & Favaro, 2019) for unsupervised scene decomposition learning. And, with the intuition that optimal clustering results would be achieved if clusters of image features are uniformly distributed and can be linearly separable from each other (Wang & Isola, 2020), we train the discriminator to optimize the contrastive loss (He et al., 2020), where the image-latent feature pairs that maximize the mutual information are set as positive pairs. In specific, we map the latent code, which is the input of the generator, onto the embedding space of the discriminator and let it act as a cluster centroid that pulls the feature of the image that has generated with this specific value. By setting the relation with other latent codes as negative pairs, the cluster centroids would be pushing each others so that the embedding space to have the characteristic described earlier. The induced embedding space of the discriminator is also helpful for image generation learning, as the generator is required to have the ability to synthesize certain number of object classes with their distinct characteristics. We have conducted experiments on four fine-grained datasets including CUB, Stanford Cars, Stanford Dogs and Oxford Flower, and show that our C3-GAN achieved state-of-the-art clustering performance on all the datasets while also alleviating the mode collapse issue.

Our main contributions are summarized as three folds:

- We propose a novel form of information-theoretic regularization using the contrastive loss. With the proposed formulation, we directly learn the cluster centroids that would lead the data to form distinct cluster boundaries in the embedding space without falling into a degenerated solution.
- By extracting foreground features via scene decomposition learning, our method achieves the state-of-the-art performance on the fine-grained class clustering while not requiring any human-annotated label.
- Our model is capable of generating images with the controllability over the object classes, and also good at alleviating the mode collapse issue of GAN.

2 Related Work

Unsupervised Clustering. Unsupervised clustering methods can be mainly categorized into information theory-based and Expectation–Maximization (EM)-based approaches. The first approach aims to maximize mutual information between groups of original images and augmented images in an end-to-end fashion (Ji et al., 2019; Zhong et al., 2020). To enhance the performance, IIC (Ji et al., 2019) additionally trains the auxiliary branch using an unlabeled large dataset, and DRC (Zhong et al., 2020) optimizes the contrastive loss on logit features for reducing the intra-class feature variation. EM-based methods decouple the cluster assigning process (E-step) and the feature representations

learning process (M-step) to learn more robust feature representations. Specifically, the feature representations are learned by not only fitting the proto clustering results estimated in the expectation step but also optimizing particular pretext tasks. The clusters are inferred by k-means clustering (Xie et al., 2016; Caron et al., 2018; Liu et al., 2020; Li et al., 2021) or training of an additional encoder (Dizaji et al., 2017). However, these methods suffer from the uneven allocations issue of k-means clustering, which could result in a degenerate solution. For this reason, SCAN (Van Gansbeke et al., 2020) proposes a novel objective function that doesn't require k-means clustering. Based on the observation that K nearest neighbors of each feature belong to the same ground-truth clusters with the high probability, it trains a classifier that assigns identical clusters to all nearest neighbors and significantly improved the clustering performance. Unlike the prior methods where the cluster centroids are not learned in an end-to-end manner or not the subject of interest, C3-GAN jointly learns the cluster centroids with other parts of the model with clear regularizations on their distribution.

Unsupervised Scene Decomposition. Heavy cost of annotating segmentation masks have drew active research efforts on developing unsupervised methods. Some works address the task in the information-theoretic perspective (Ji et al., 2019; Savarese et al., 2021), but a majority of the works are based on GAN. ReDO (Chen et al. (2019)) learns to discern object masks by redrawing a scene, and PerturbGAN (Bielski & Favaro (2019)), that has separate background and foreground generators, triggers scene decomposition learning by randomly perturbing foreground images. Meanwhile, a line of works utilizes pre-trained high quality models such as StyleGAN (Karras et al., 2019; 2020) and BigGAN (Brock et al., 2019). Labels4Free (Abdal et al., 2021) trains a segmentation network on top of the pre-trained StyleGAN utilizing the fact that inputs of each layer of StyleGAN have different degrees of contribution to foreground synthesis. Voynov & Babenko (2020) and Melas-Kyriazi et al. (2021) explore the latent space of pre-trained GANs to find the perturbing directions that can be used to induce segmentation masks. While all these methods aim to infer foreground masks by training a mask predictor with a synthesized dataset, or projecting real images into the latent space of the pre-trained GAN, C3-GAN is more focusing on the semantic representations learning of the foreground object.

Fine-grained Feature Learning. For hierarchical image generation, some works (Singh et al., 2019; Li et al., 2020; Benny & Wolf, 2020) extend InfoGAN with multiple pairs of adversarial networks, and integrate the scene decomposition learning method into the framework. FineGAN (Singh et al., 2019) finds multiple latent codes using information theory, and each code is used for sequential generation of a background, a super-class object, and a sub-class object image. MixNMatch (Li et al., 2020) and OneGAN (Benny & Wolf, 2020) extend FineGAN with multiple encoders to directly infer these latent codes from real images. Since this autoencoder-based structure could evoke degenerate solutions where only one generator is responsible for synthesizing the entire image, MixNMatch conducts adversarial learning on the joint distribution of the latent code and image (Donahue et al., 2017), and OneGAN trains the model in two stages, learning the generators first and the encoders next. All of these works succeeded in generating images in a hierarchical manner, and further suggest that the learned latent codes can be used for clustering real images into their fine-grained classes. However, these works cannot be applied to datasets unlabeled with bounding box annotations, and require a training of additional networks (classifiers). The proposed model in this paper, C3-GAN, can learn the latent code of foreground region in a completely unsupervised way, and utilize the discriminator for inferring the clusters of the given dataset.

3 METHOD

Given a dataset $X = \{x_i\}_{i=1}^{N}$ consisting of single object images, we aim to distribute the set X into Y semantically fine-grained clusters in an unsupervised way. Our method, C3-GAN, is built from the GAN, and the clusters are found in the feature space of the discriminator D, $\mathcal{H} \in \mathbb{R}^{d^h}$. The cluster index for each image x_i is inferred from the cosine distance between embedded image feature $h_i \in \mathbb{R}^{d^h}$ and the set of cluster centroids $l \in \mathbb{R}^{Y \times d^h}$, which are the latent codes c mapped onto the space. Specifically, the index of the closest centroid will be assigned to the query image x_i. For the robust fine-grained feature learning, C3-GAN decomposes the scene into the background and foreground part, and associate the latent code only with the foreground region. Moreover, it induces the discriminator D to learn the embedding space where each cluster is uniformly distributed with distinctive boundaries, by formulating the information regularization based on the contrastive loss.

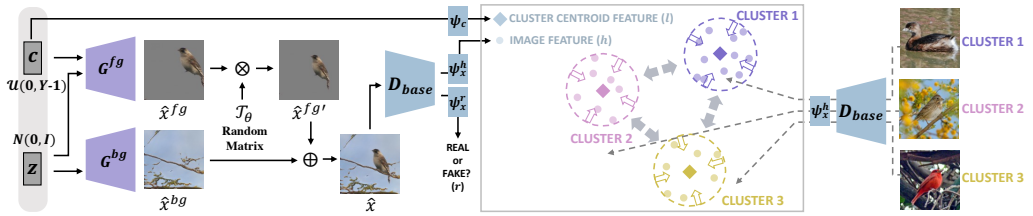


Figure 2: Overview of C3-GAN. The background generator G^{bg} and foreground generator G^{fg} synthesize a background image $\hat{x}^{bg}$ and a foreground image $\hat{x}^{fg}$ respectively. To trigger the decomposed image generation, we perturb the foreground image with the random affine transformation matrix $\mathcal{T}_\theta$ right before the composition. The association between the latent code c and the image $\hat{x}$ is learned by optimizing the contrastive loss of their feature representations in the embedding space of the discriminator D. The dotted line represents the points where the real image features h (•) land, and the cluster assignment is made based on the distance between these and the cluster centroids l (♦), which are the embedded latent codes c.

3.1 Preliminaries

The proposed C3-GAN is built from FineGAN (Singh et al., 2019), which is in turn based on InfoGAN (Chen et al., 2016). InfoGAN learns to associate a random latent code c with its observation $\hat{x}$ by optimizing the model to maximize the mutual information of the two, $\mathrm{I}(c, \hat{x})$. The latent code c can take various forms depending on the prior knowledge of the factor to be inferred, and be defined as the uniform categorical distribution when the purpose is to make the categorical inference of the data. FineGAN learns three such latent codes for hierarchical image generation, and each code is used for a background, a super-class object, and a sub-class object image synthesis respectively. To facilitate this decomposition, FineGAN employs multiple pairs of generator and discriminator and trains an auxiliary background classifier using object bounding box labels. It is demonstrated that the latent codes for the foreground image generation can also be utilized for clustering images according to their fine-grained features. The inference is made by training additional classifiers with synthesized images that are pseudo-labeled with their corresponding latent code values.

Our method differs from FineGAN in that, i) it employs the discriminator D for the clustering without requiring an additional training, and ii) it learns only one latent code c that represents the fine-grained class of an foreground object. Specifically, our generator G synthesizes background image $\hat{x}^{bg} \in \mathbb{R}^{3\times H\times W}$, foreground mask $\hat{m} \in \mathbb{R}^{1\times H\times W}$ and foreground appearance $\hat{t} \in \mathbb{R}^{3\times H\times W}$ with two independent random variables, the noise value z and the latent code c. The noise $z \in \mathbb{R}^{d^z}$ is sampled from the normal distribution $\mathcal{N}(0, I)$. The latent code $c \in \mathbb{R}^{Y}$ is represented as 1-hot encoding where the index k such that $c_k = 1$ is sampled from the uniform categorical distribution $\mathcal{U}(0, Y-1)$. The noise value z is used for generating all image components making the layout of the scene, while the latent code c is used only for the foreground synthesis. To add some randomness, we augmented the 1-hot latent code c to a latent variable $c' \in \mathbb{R}^{d^c}$ which is sampled from a gaussian distribution $N(\mu_c, \sigma_c)$ where values of the mean μ_c and the diagonal covariance matrix σ_c are conditioned on the original code c (Zhang et al., 2017). The final image $\hat{x} \in \mathbb{R}^{3\times H\times W}$ is composited by applying the hadamard product to the generated components, as described in Fig. 2.

To achieve fully unsupervised scene decomposition by removing the dependence on the bounding box annotations, we leverage the random affine transformation method inspired by PerturbGAN (Bielski & Favaro, 2019). In specific, scene decomposition is triggered by perturbing foreground components $\hat{m}$ and $\hat{t}$ with the affine transformation matrix $\mathcal{T}_\theta$ right before the final image composition. The parameters θ of the transformation matrix $\mathcal{T}_\theta$ include rotation angle, scaling factor and translation distance, and they are all randomly sampled from the uniform distributions with predefined value ranges. We also employ the regularization functions that enforce foreground mask $\hat{m}$ to be more like a hard mask and to take up the reasonable proportions of the scene,

$$\begin{aligned}\mathcal{L}_G^{mask} = &\ \mathbb{E}_{hw}[-\hat{m}\ \log(\hat{m}) - (1-\hat{m})\ \log(1-\hat{m})] \\ &+ \max(0, 0.1 - \mathbb{E}_{hw}[\hat{m}]) + \max(0, \mathbb{E}_{hw}[\hat{m}] - 0.9).\end{aligned} \tag{1}$$

In summation, the generation process is described as:

$$\begin{aligned} c' \sim N(\mu_c, \sigma_c), \quad \hat{x}^{bg}, \hat{m}, \hat{t} = G(z, c'), \\ \hat{m}', \hat{t}' = \mathcal{T}_\theta(\hat{m}, \hat{t}), \quad \hat{x} = \hat{x}^{bg} \odot (1 - \hat{m}') + \hat{t}' \odot \hat{m}'. \end{aligned} \tag{2}$$

3.2 Contrastive Fine-grained Class Clustering

We assume that the dataset X would be effectively clustered when i) it explicitly forms discernable cluster boundaries in the embedding space $\mathcal{H}$ and ii) each cluster centroid $l \in \mathbb{R}^{d^h}$ exhibits distinct semantic features. This is exactly the space that the discriminator of InfoGAN aims to approach when the latent code c is sampled from the uniform categorical distribution $\mathcal{U}(0, Y-1)$ (Chen et al., 2016). However, while jointly optimizing an adversarial loss, the model has a possibility of falling into the mode collapse where a generator G covers only few confident modes (classes) to easily deceive the discriminator D. This is why InfoGAN cannot infer the classes on real images. To handle this problem, we propose a new form of the auxiliary probability $Q(c|x)$ that represents the mutual information between the latent code c and image x.

Let us describe the mathematical definition in details. The discriminator D aims to learn adversarial features $r \in \mathbb{R}$ for image authenticity discrimination, and image features $h \in \mathbb{R}^{d^h}$ for optimizing the information-theoretic regularization. The features are encoded from separate branches ψ_x^r and ψ_x^h, which has split at the end of the base encoder of the discriminator D_{base}. The adversarial features r are learned with the hinge loss as shown in the equation below:

$$\begin{aligned} r = \psi_x^r(D_{base}(x)), \qquad \hat{r} = \psi_x^r(D_{base}(\hat{x})) \\ \mathcal{L}_D^{adv} = \min(0, 1-r) + \min(0, 1+\hat{r}), \qquad \mathcal{L}_G^{adv} = -\hat{r}. \end{aligned} \tag{3}$$

Meanwhile, the image features h can be considered as logits in a classification model. In a supervised learning setting where the ground-truth class labels are given, the data mapped into the embedding space gradually forms the cluster boundaries as the training proceeds, like the one in Fig. 1 (b). However, it is difficult to learn such space in the given problem setting simply by maximizing the mutual information between the latent code c and its observation $\hat{x}$, $\mathrm{I}(c; \hat{x})$, for the aforementioned reasons. Our solution is to formulate the auxiliary probability $Q(c|x)$ based on the contrastive loss. Specifically, we map the latent code c onto the embedding space of the discriminator with simple linear layer ψ_c, and let it act as a cluster centroid l that pulls the feature of the image h that has generated with this specific value. By setting the relation with other latent codes as negative pairs, the cluster centroids would be pushing each others so that they are distributed with distinct boundaries. The equation of the proposed formulation is as below:

$$\begin{aligned} Q(c_k = 1|\hat{x}) = q_k = \frac{\exp(\mathrm{sim}(\hat{h}, l_k)/\tau)}{\sum_{y=0}^{Y-1} \exp(\mathrm{sim}(\hat{h}, l_y)/\tau)} \\ s.t. \qquad \hat{h} = \psi_x^h(D_{base}(\hat{x})), \quad l = \psi_c(I_Y), \end{aligned} \tag{4}$$

where $\mathrm{sim}(a, b)$ is the cosine distance between vectors a and b, y is the index of the cluster, and τ is a temperature of the softmax function. I_Y denotes the identity matrix with size Y.

With this function, we expect that the image feature h would be mapped closest to k-th cluster, where k is the index of its corresponding latent code c such that $c_k = 1$, while distances to other centroids l_y would be far enough so that the value of q_k is close to 1. We additionally maximize the mutual information between the latent code c and the masked foreground images $\hat{x}^{fg}$ to further assist the scene decomposition learning. So the information-theoretic objectives are defined as:

$$\mathcal{L}^{info} = -\log q_k, \qquad \mathcal{L}^{info^{fg}} = -\log q_k^{fg}. \tag{5}$$

Following the prior works (Ji et al., 2019; Van Gansbeke et al., 2020), we adopted overclustering strategy so that the model can learn more expressive features. We also regularize the prediction on real images $p \in \mathbb{R}^Y$ to further reduce the gap between real and fake images. Specifically, we minimize the entropy of p, $H(p)$, to promote each data to be mapped to one cluster with high confidence and the KL divergence between the averaged prediction $\bar{\mathbf{p}}$ and the uniform distribution $\mathbf{u}$ to avoid a

degenerated solution where few clusters are overly allocated. The regularizations are as below:

$$\mathcal{L}_D^{entropy} = \mathbb{E}[H(p)] + D_{\mathrm{KL}}(\bar{\mathbf{p}} \| \mathbf{u})$$
$$s.t. \qquad h = \psi_x^h(D_{base}(x)) \,, \quad p_k = \frac{\exp(\mathrm{sim}(h, l_k)/\tau)}{\sum_{y=0}^{Y-1} \exp(\mathrm{sim}(h, l_y)/\tau)} \,, \quad \bar{\mathbf{p}} = \mathbb{E}[p]. \tag{6}$$

In summary, our C3-GAN is trained optimizing the following objective functions iteratively:

$$\min_{D,G} \; \mathcal{L}_D^{adv} + \mathcal{L}_G^{adv} + \lambda_0 \mathcal{L}_D^{entropy} + \lambda_1 \mathcal{L}_G^{mask} + \lambda_2 \mathcal{L}^{info} + \lambda_3 \mathcal{L}^{info^{fg}}. \tag{7}$$

4 Experiments

4.1 Experimental Setting

Datasets. We tested our method on 4 datasets which contain single object images. **i) CUB** (Wah et al., 2011): 5,994 training and 5,794 test images of 200 bird species. **ii) Oxford Flower** (Nilsback & Zisserman, 2008): 2,040 training and 6,149 test images of 102 flower categories. **iii) Stanford Cars** (Krause et al., 2013): 8,144 training and 8,041 test images of 196 car models. **iv) Stanford Dogs** (Khosla et al., 2011): 12,000 training and 8,580 test images of 120 dog species. Since the CUB and Oxford Flower datasets have a small number of training images, all the experiments were conducted using the entire data as in prior works (Singh et al., 2019; Li et al., 2020; Benny & Wolf, 2020).

Implementation Details. The weights of loss terms $(\lambda_0, \lambda_1, \lambda_2, \lambda_3)$ are set as (0.1, 1, 5, 1), and the temperature τ is set as 0.1. The models are trained with Adam optimizer with learning rate of 0.0002 and momentum coefficients of (0.5, 0.999). We described the network specification in Appendix A.1, and the value ranges of parameters for the random affine transformation and the number of clusters Y in Appendix A.2.

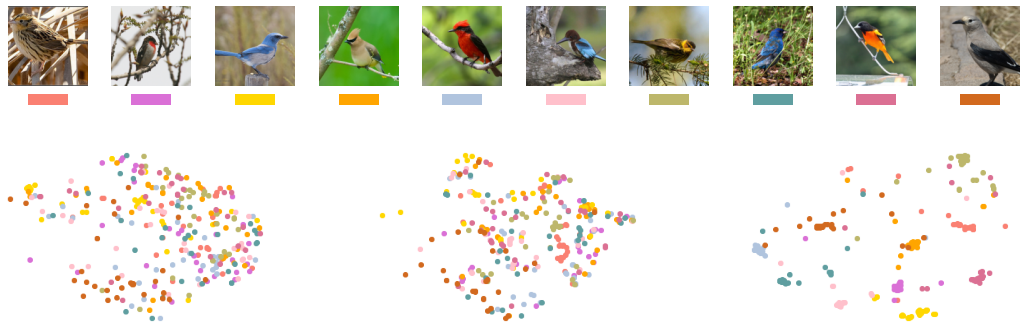
Baselines. We first compare C3-GAN to **IIC** (Ji et al., 2019) in order to emphasize the difference between the coarse-grained and fine-grained class clustering tasks since it achieved decent clustering results with sobel filtered images where color and texture information is discarded. To claim that optimizing the instance discrimination task is not enough for fine-grained feature representations learning, we also compare with the simple baseline of **SimCLR+K-Means** (Chen et al., 2020). The work that shows remarkable result on the coarse-grained class clustering, **SCAN** (Van Gansbeke et al., 2020), are also included to evaluate their method that induces consistent predictions between nearest neighbors on the fine-grained class clustering task. In addition to **InfoGAN** (Chen et al., 2016), the reported results of **FineGAN** (Singh et al., 2019), **MixNMatch** (Li et al., 2020) and **OneGAN** (Benny & Wolf, 2020) were compared to claim the efficacy of our proposed formulation of the auxiliary probability based on the contrastive loss. For the fair comparison, we mainly compared with the results of the unsupervised version of FineGAN and MixNMatch, which are trained by pseudo-labeling the edges of real images as background regions.

4.2 Fine-grained Class Clustering Results

Quantitative results. We evaluate the clustering performance with two metrics: Accuracy (Acc) and Normalized Mutual Information (NMI). The accuracy is calculated based on the optimal mapping between the cluster indices and the real classes inferred from the Hungarian algorithm (Kuhn & Yaw, 1955). The results are presented in Table 1. As shown in Table 1, our C3-GAN outperforms other methods with remarkable margins in terms of Acc on all datasets. For NMI, our method presents better or comparable performances as well. The lowest results of IIC underlines that understanding only structural characteristic is not enough for the fine-grained class clustering task. It is difficult to decide which of the GAN-based methods or the SSL-based methods are better, since both approaches have similar range of scores on every metric and display globally entangled feature spaces as shown in Fig. 3 (a) and (b). From the results of SCAN (2nd rank) and MixNMatch (3rd rank), we conjecture that circumventing the problem by requiring local disentanglement of the embedding space and extracting foreground features via scene decomposition are helpful for improving the fine-grained

	Acc ↑				NMI ↑			
	Bird	Car	Dog	Flower	Bird	Car	Dog	Flower
FineGAN† (Singh et al., 2019)	*12.6*	*7.8*	*7.9*	-	*0.40*	*0.35*	*0.23*	-
MixNMatch† (Li et al., 2020)	*13.6*	*7.9*	*8.9*	-	*0.42*	*0.36*	*0.32*	-
OneGAN† (Benny & Wolf, 2020)	*10.1*	*6.0*	*7.3*	-	*0.39*	*0.27*	*0.21*	-
Fully Unsupervised Setting								
IIC (Ji et al., 2019)	7.4	4.9	5.0	8.7	0.36	0.27	0.18	0.24
SimCLR (Chen et al., 2020) +k-Means	8.4	6.7	6.8	12.5	0.40	0.33	0.19	0.29
InfoGAN (Chen et al., 2016)	8.6	6.5	6.4	23.2	0.39	0.31	0.21	0.44
FineGAN w/o labels	6.9	6.8	6.0	8.10	0.37	0.33	0.22	0.24
MixNMatch w/o labels	10.2	7.3	10.3	39.0	0.41	0.34	0.30	0.57
SCAN (Van Gansbeke et al., 2020)	11.9	8.8	12.3	56.5	0.45	0.38	0.35	**0.77**
C3-GAN (Ours)	**27.6**	**14.1**	**17.9**	**67.8**	**0.53**	**0.41**	**0.36**	0.67

Table 1: Quantitative evaluation of the clustering performance. † denotes the results reported in the original papers, and the rest is reproduced by our implementation. Note that FineGAN, MixNMatch, and OneGAN utilize supervision for training unlike our C3-GAN.



(a) MixNMatch (Li et al., 2020) (b) SCAN (Van Gansbeke et al., 2020) (c) C3-GAN (Ours)

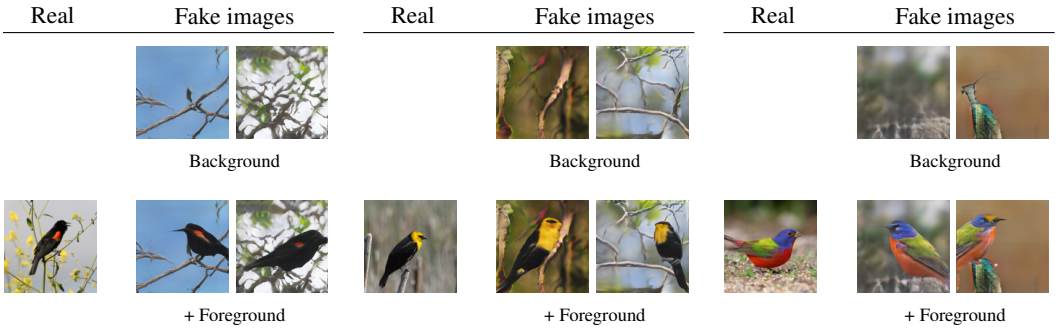
Figure 3: Visualization of the embedding space of MixNMatch, SCAN and C3-GAN trained on CUB dataset. Plotted points are image features of 10 classes that are reduced into 2-dimensional space via t-SNE. The colors represent the ground truth classes of each point. Only C3-GAN has a embedding space where the cluster distinction is clear.

clustering performance. The state-of-the-art scores of C3-GAN are not only benefited from the scene decomposition learning, but also boosted from the proposed formulation of $Q(c|x)$ that enforces the clusters to be distinctively distributed in the embedding space as visualized in Fig. 3 (c). Furthermore, we present multiple qualitative results on fine-grained clustering tasks in the Appendix A.3, which show the superiority of our method compared to SCAN and MixNMatch.

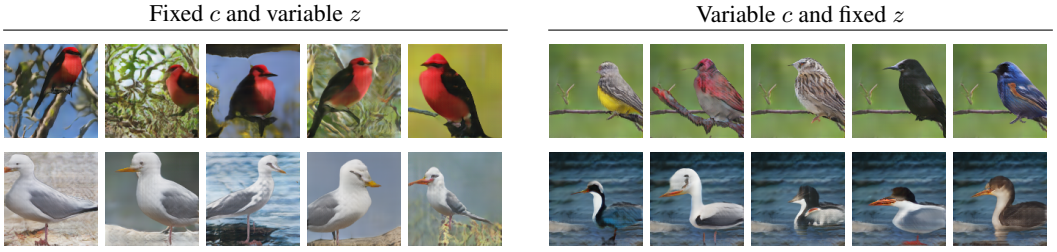
Ablation Study. We performed ablation study regarding three factors: i) Overclustering, ii) Perturbing foreground images via random affine transformation, and iii) Our formulation of the auxiliary probability $Q(c|x)$. We investigate the clustering performance on CUB dataset by removing each factor and the results are shown in Table 2. The largest performance degradation is induced by changing the formulation of the auxiliary probability $Q(c|x)$ to a trivial softmax function. This result implies that the high degree of cluster separation, which achieved with our proposed method, is strongly correlated with the clustering performance (Wang & Isola, 2020). We observe that C3-GAN fails to decompose background and foreground parts when the random perturbation $\mathcal{T}_\theta$ is not applied. The significant performance degradation suggests that the method of extracting only the foreground features has a non-negligible amount of benefit in the clustering performance. Overclustering is also helpful for boosting the performance by allowing a larger capacity for the feature representations learning. We added more ablation results on overclustering in the Appendix A.2.

		Acc ↑	NMI ↑
C3-GAN (Ours)		**27.6**	**0.53**
i)	– Overclustering	22.7	0.50
ii)	– Perturbing foreground images via random affine transformation	16.6	0.47
iii)	– Contrastive loss based auxiliary probability $Q(c\|x)$	14.5	0.45

Table 2: Effects of each component of C3-GAN on clustering performance. The largest performance drop is induced by changing the formulation of the auxiliary probability $Q(c|x)$ to a trivial softmax function.



(a) Images synthesized with the predicted cluster index of a given real image.



(b) Images synthesized by controlling latent code c and noise value z.

Figure 4: Qualitative evaluation of image generation performance of C3-GAN trained on CUB dataset. We present (a) the synthesized images with the predicted cluster indices from real images and (b) the randomly synthesized images with sampled input variables z and c.

4.3 Image Generation Results

Qualitative Results. Fig. 4 presents images synthesized by C3-GAN trained on CUB dataset. First, we plot the synthesized novel scenes with the predicted cluster indices of real images in Fig. 4(a). Considering the identical characteristics of synthesized foreground objects, C3-GAN achieved to learn cluster centroids with distinctive fine-grained features set so that it can also clearly mapped with real images. To more investigate the characteristics of two input variables, c and z, we plot the results synthesized by fixing one of the two while diversifying the other one in Fig. 4(b). When c is fixed, all the generated images contain the same bird species while the pose of the bird and backgrounds vary due to the variation of z value. On the contrary, for the variable c and fixed z setting, we could observe images of various bird species with identical pose and background features. We also plot the results of other datasets in the appendix A.4.

We observe that C3-GAN also effectively alleviates the mode collapse issue from Fig. 5. The images are synthesized with the cluster index corresponding to the class of real image. While C3-GAN presents *intra*-class color, shape, and layout variation for a given flower class, the results of other baselines differ only in layout. We conjecture that this is because the learnt embedding space of the discriminator represents the real data distribution so well, that the generator gets a stronger signal for learning to synthesize realistic images in addition to the adversarial loss. Through the qualitative

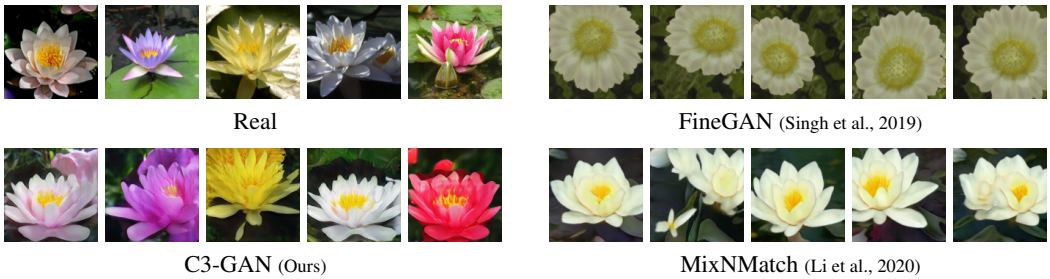


Figure 5: Qualitative image generation results of C3-GAN trained on Oxford flower dataset. The fake images are synthesized with cluster indices predicted by each model's own method. Only C3-GAN generates images that reflect the *intra*-class color, shape, and layout variation of the real images.

	FID ↓				IS ↑				Reverse KL ↓			
	Bird	Car	Dog	Flower	Bird	Car	Dog	Flower	Bird	Car	Dog	Flower
InfoGAN (Chen et al., 2016)	35.71	70.91	59.07	78.37	8.20	3.25	6.91	13.39	0.56	0.73	0.40	0.42
FineGAN w/o labels	33.87	87.66	49.12	35.72	10.07	**4.20**	10.67	10.42	0.67	0.58	0.24	0.52
MixNMatch w/o labels	31.59	78.36	48.11	**32.03**	9.76	4.11	**10.70**	**16.48**	0.52	**0.57**	0.22	0.34
C3-GAN (Ours)	**19.37**	**67.36**	**45.40**	64.19	**14.52**	3.62	9.33	13.75	**0.29**	**0.57**	**0.18**	**0.25**

Table 3: Quantitative evaluation of the image synthesis performance.

investigation of synthesized images, we also found that C3-GAN was the only one that succeeded in scene decomposition learning in an unsupervised way. FineGAN and MixNMatch trained without bounding box annotations fell into a degenerate solution where the entire image is synthesized from the background generator.

Quantitative Results. We evaluate image synthesis performance based on the scores of Frechet Inception Distance (FID) and Inception Score (IS), in addition to the score of reverse KL which measures the distance between the predicted class distribution of real and synthesized images. We compared C3-GAN with the GAN-based models trained in a fully unsupervised setting. The scores of IS and reverse KL were measured with Inception networks fine-tuned on each fine-grained dataset using the method of Cui et al. (2018)[1], and the results are displayed in Table 3. C3-GAN presents state-of-the-art or comparable performance in terms of FID and Reverse KL. With the qualitative results together, these results imply our C3-GAN has competitive capability for fine-grained object image generation.

5 Conclusion

In this study, we proposed a new GAN-based method, C3-GAN, for unsupervised fine-grained class clustering which is more challenging and less explored than coarse-grained clustering tasks. To improve fine-grained class clustering performance, we formulate the information-theoretic regularization based on the contrastive loss. Also, a scene decomposition-based approach is incorporated into the framework to enforce the model to learn representation focusing on the details of foreground images. We further enhance the performance by adopting the overclustering approach. Extensive experiments show our C3-GAN not only outperforms previous fine-grained clustering methods but also synthesizes fine-grained object images with comparable quality while reducing mode collapse that previous state-of-the-art GAN methods suffer from. Since our main goal is unsupervised fine-grained clustering rather than image generation, C3-GAN still has room to be improved despite its comparable results.

[1]The accuracy of Inception Networks fine-tuned on the CUB/Oxford Flower/Stanford Dogs/Stanford Cars datasets are 0.87/0.94/0.86/0.47 respectively.

ETHICS STATEMENT

Remarkable advancement of generative models has played a crucial role as AI tools for text (Brown et al., 2020; Kim et al., 2021), image (Karras et al., 2020; Choi et al., 2020b), audio (Sisman et al., 2020), and multimodal content generation (Tian et al., 2021; Yan et al., 2021). However, as its side effect, many malicious applications have been reported as well, thus leading to severe societal problems, such as deepfake (Dolhansky et al., 2020), fake news (Lazer et al., 2018), and generation biased by training data (Gupta et al., 2021). Our work might be an extension of the harmful effects of generative models. On the contrary, generative models can be a solution to alleviate these side effects via data augmentation (Yang et al., 2019; Choi et al., 2020a). In particular, our method can contribute to alleviating data bias and fine-grained class imbalance. Therefore, many endless efforts are required to make these powerful generative models benefit humans.

REPRODUCIBILITY STATEMENT

We described the datasets and baselines compared with our method in detail in 4.1. The details of our model architecture and hyperparameter setup are also explained in A.1 and A.2. And we plan to open the source code of C3-GAN for reproducibility.

REFERENCES

Rameen Abdal, Peihao Zhu, Niloy J. Mitra, and Peter Wonka. Labels4free: Unsupervised segmentation using stylegan. In *ICCV*, 2021.

Yaniv Benny and Lior Wolf. Onegan: Simultaneous unsupervised learning of conditional image generation, foreground segmentation, and fine-grained clustering. In *ECCV*, 2020.

Adam Bielski and Paolo Favaro. Emergence of object segmentation in perturbed generative models. In *NeurIPS*, 2019.

Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale GAN training for high fidelity natural image synthesis. In *ICLR*, 2019.

Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. In *NeurIPS*, 2020.

Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *ECCV*, 2018.

Mickaël Chen, Thierry Artières, and Ludovic Denoyer. Unsupervised object segmentation by redrawing. In *NeurIPS*, 2019.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020.

Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *NeurIPS*, 2016.

Kristy Choi, Aditya Grover, Trisha Singh, Rui Shu, and Stefano Ermon. Fair generative modeling via weak supervision. In *ICML*, 2020a.

Yunjey Choi, Youngjung Uh, Jaejun Yoo, and Jung-Woo Ha. Stargan v2: Diverse image synthesis for multiple domains. In *CVPR*, 2020b.

Adam Coates, Andrew Ng, and Honglak Lee. An analysis of single layer networks in unsupervised feature learning. In *AISTATS*, 2011.

Yin Cui, Yang Song, Chen Sun, Andrew Howard, and Belongie Serge. Large scale fine-grained categorization and domain-specific transfer learning. In *CVPR*, 2018.

Kamran Ghasedi Dizaji, Amirhossein Herandi, and Heng Huang. Deep clustering via joint convolutional autoencoder embedding and relative entropy minimization. In *NeurIPS*, 2017.

Brian Dolhansky, Joanna Bitton, Ben Pflaum, Jikuo Lu, Russ Howes, Menglin Wang, and Cristian Canton Ferrer. The deepfake detection challenge (dfdc) dataset. In *arXiv*, 2020.

Jeff Donahue, Philipp Krähenbühl, and Trevor Darrell. Adversarial feature learning. In *ICLR*, 2017.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NeurIPS*, 2014.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent: A new approach to self-supervised learning. In *NeurIPS*, 2020.

Aman Gupta, Deepak Bhatt, and Anubha Pandey. Transitioning from real to synthetic data: Quantifying the bias in model. In *Synthetic Data Generation Workshop at ICLR 2021*, 2021.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020.

Irina Higgins, Loïc Matthey, Arka Pal, Christopher P. Burgess, Xavier Glorot, Matthew M. Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. In *ICLR*, 2017.

Xu Ji, João F Henriques, and Andrea Vedaldi. Invariant information clustering for unsupervised image classification and segmentation. In *ICCV*, 2019.

Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *CVPR*, 2019.

Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *CVPR*, 2020.

Aditya Khosla, Nityananda Jayadevaprakash, Bangpeng Yao, and Li Fei-Fei. Novel dataset for fine-grained image categorization. In *First Workshop on Fine-Grained Visual Categorization*, 2011.

Boseop Kim, HyoungSeok Kim, Sang-Woo Lee, Gichang Lee, Donghyun Kwak, Dong Hyeon Jeon, Sunghyun Park, Sungju Kim, Seonhoon Kim, Dongpil Seo, et al. What changes can large-scale language models bring? intensive study on hyperclova: Billions-scale korean generative pretrained transformers. In *EMNLP*, 2021.

Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *4th International IEEE Workshop on 3D Representation and Recognition*, 2013.

H. W. Kuhn and Bryn Yaw. The hungarian method for the assignment problem. In *Naval Research Logistics Quarterly*, 1955.

David MJ Lazer, Matthew A Baum, Yochai Benkler, Adam J Berinsky, Kelly M Greenhill, Filippo Menczer, Miriam J Metzger, Brendan Nyhan, Gordon Pennycook, David Rothschild, et al. The science of fake news. In *AAAS*, 2018.

Yann LeCun, Corinna Cortes, and CJ Burges. Mnist handwritten digit database. In *ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist*, 2010.

Junnan Li, Pan Zhou, Caiming Xiong, and Steven C.H. Hoi. Prototypical contrastive learning of unsupervised representations. In *ICLR*, 2021.

Yuheng Li, Krishna Kumar Singh, Utkarsh Ojha, and Yong Jae Lee. Mixnmatch: Multifactor disentanglement and encoding for conditional image generation. In *CVPR*, 2020.

Steven Liu, Tongzhou Wang, David Bau, Jun-Yan Zhu, and Antonio Torralba. Diverse image generation via self-conditioned gans. In *CVPR*, 2020.

Luke Melas-Kyriazi, Christian Rupprecht, Iro Laina, and Andrea Vedaldi. Finding an unsupervised image segmenter in each of your deep generative models. In *arXiv*, 2021.

Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *ICCVGIP*, 2008.

Pedro Savarese, Sunnie S. Y. Kim, Michael Maire, Greg Shakhnarovich, and David McAllester. Information-theoretic segmentation by inpainting error maximization. In *CVPR*, 2021.

Krishna Kumar Singh, Utkarsh Ojha, and Yong Jae Lee. Finegan: Unsupervised hierarchical disentanglement for fine-grained object generation and discovery. In *CVPR*, 2019.

Berrak Sisman, Junichi Yamagishi, Simon King, and Haizhou Li. An overview of voice conversion and its challenges: From statistical modeling to deep learning. In *TASLP*, 2020.

Yu Tian, Jian Ren, Menglei Chai, Kyle Olszewski, Xi Peng, Dimitris N Metaxas, and Sergey Tulyakov. A good image generator is what you need for high-resolution video synthesis. In *ICLR*, 2021.

Wouter Van Gansbeke, Simon Vandenhende, Stamatios Georgoulis, Marc Proesmans, and Luc Van Gool. Scan: Learning to classify images without labels. In *ECCV*, 2020.

Andrey Voynov and Artem Babenko. Unsupervised discovery of interpretable directions in the gan latent space. In *ICML*, 2020.

Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. In *California Institute of Technology*, 2011.

Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *ICML*, 2020.

Junyuan Xie, Ross B. Girshick, and Ali Farhadi. Unsupervised deep embedding for clustering analysis. In *ICML*, 2016.

Wilson Yan, Yunzhi Zhang, Pieter Abbeel, and Aravind Srinivas. Videogpt: Video generation using vq-vae and transformers. In *arXiv*, 2021.

Shuo Yang, Kai Shu, Suhang Wang, Renjie Gu, Fan Wu, and Huan Liu. Unsupervised fake news detection on social media: A generative approach. In *AAAI*, 2019.

Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitris Metaxas. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In *ICCV*, 2017.

Huasong Zhong, Chong Chen, Zhongming Jin, and Xian-Sheng Hua. Deep robust clustering by contrastive learning. In *arXiv*, 2020.

A APPENDIX

A.1 NETWORK ARCHITECTURE

We present the detailed description of architecture of the generator and the discriminator for our C3-GAN in Table 4 and Table 5, respectively. The colored dots (●, ●, ●) indicate that outputs of the earlier layers are used as inputs for the later layers. The training is done with 2 NVIDIA-V100 GPUs, and we optimized the model until the convergence of the FID score.

Module	Layers	Input size	Output size
G_{bg}	Linear, BN, GLU, UnSqueeze	64 ($\mathbf{z}$)	512×4×4
	[Up, Conv(3,1), BN, GLU]×5	512×4×4	64×128×128
	[Conv(3,1), BN, GLU, Conv(3,1), BN]×3	64×128×128	64×128×128
	Conv(3,1), Tanh	64×128×128	3×128×128 ($\hat{\mathbf{x}}^{\mathbf{bg}}$)
$N(\mu_c, \sigma_c)$	Linear, BN, GLU	Y ($\mathbf{c}$)	8 ●
G_{fg}^{base}	Linear, BN, GLU, Reshape	64 ($\mathbf{z}$)	512×4×4
	[Up, Conv(3,1), BN, GLU]×5	(512+8 ●)×4×4	16×128×128
	[Conv(3,1), BN, GLU, Conv(3,1), BN]×3	16×128×128	16×128×128 ●
G_{fg}^{m}	Conv(3,1), BN, GLU	16×128×128 ●	64×128×128
	Conv(3,1), Sigm	64×128×128	1×128×128 ($\hat{\mathbf{m}}$)
G_{fg}^{t}	Conv(3,1), BN, GLU	(16● + Y ($\mathbf{c}$))×128×128	16×128×128
	[Conv(3,1), BN, GLU, Conv(3,1), BN]×2	16×128×128	16×128×128
	Conv(3,1), BN, GLU	16×128×128	16×128×128
	Conv(3,1), Tanh	16×128×128	3×128×128 ($\hat{\mathbf{t}}$)

Table 4: Architectural description of the generator of C3-GAN.

Module	Layers	Input size	Output size
D_{base}	Conv(4,2), LReLU	3×128×128 ($\mathbf{x}$)	64×64×64
	[Conv(4,2), BN, LReLU]×4	64×64×64	512×4×4
	Conv(3,1), BN, LReLU	512×4×4	512×4×4 ●
ψ_x^r	Conv(4,4), Squeeze	512×4×4 ●	1 ($\mathbf{r}$)
ψ_x^h	Conv(3,1), BN, LReLU	512×4×4 ●	512×4×4
	Conv(4,4), Squeeze	512×4×4	512 ($\mathbf{h}$)
ψ_c	Linear	Y ($\mathbf{c}$)	512 ($\mathbf{l}$)

Table 5: Architecture description of the discriminator of C3-GAN.

- Conv(a,b) : 2D Convolution layer with kernel size a and stride b
- GLU : Gated Linear Units

A.2 Additional Ablation Study

Random affine transformation. To validate the effects of random affine transformation, we experimented with two types of purtubing policies such as weak and strong perturbation. Their detailed value ranges are presented in Table 6. The type of perturbation to use for each dataset was determined by manually checking the results of randomly perturbed real images. The strong perturbation was used for CUB dataset, and the rest datasets were trained with the weak perturbation.

Criteria	Scale	Rotation	Translation	Dataset
Weak perturbation	(0.9, 1.1)	(-2, 2)	(-0.08, 0,08)	Stanford Cars, Stanford Dogs, Oxford Flower
Strong perturbation	(0.8, 1.5)	(-15, 15)	(-0.15, 0,15)	CUB

Table 6: Details of random affine transformation

Cluster numbers for each dataset. We use four datasets for evaluation, including CUB, Oxford Flower, Standford Cars, and Stanford Dogs. As depicted in 3.2, we employ the overclustering approaches to improve the performances, which is to set the cluster number to be multiple times as the actual number of classes of the dataset. For finding optimal setting, we compared the results of C3-GAN by setting the number of clusters as 1, 2, and 3 times the actual number of classes. We found that performance tends to improve as the number of clusters increases. This implies that overclustering is indeed helpful for expressive feature learning. The cluster numbers for each dataset were determined based on the results in Table 7. To further investigate the results when we are not aware of the actual number of classes, we additionally conducted all experiments by setting the number of clusters as 100 (underclustering) or 500 (overclustering). The results are represented in Table 8. For the underclustering setting, we only report the scores of NMI, since some classes have no matching cluster indices. Despite the less promising results of the underclustering, this implies that our method can be applied to datasets whose cluster size is not available if we set a large enough number of clusters.

		Acc ↑	NMI ↑	FID ↓
CUB	× 1	22.7	0.50	18.75
	× 2	**27.6**	**0.53**	19.37
	× 3	26.3	0.51	**17.13**
Stanford Cars	× 1	8.3	0.33	**64.55**
	× 2	11.5	0.39	66.65
	× 3	**14.1**	**0.41**	67.36
Stanford Dogs	× 1	11.8	0.30	51.37
	× 2	15.8	0.35	54.82
	× 3	**17.9**	**0.36**	**45.40**
Oxford Flower	× 1	55.6	**0.72**	75.12
	× 2	61.7	0.67	74.59
	× 3	**67.8**	0.67	**64.19**

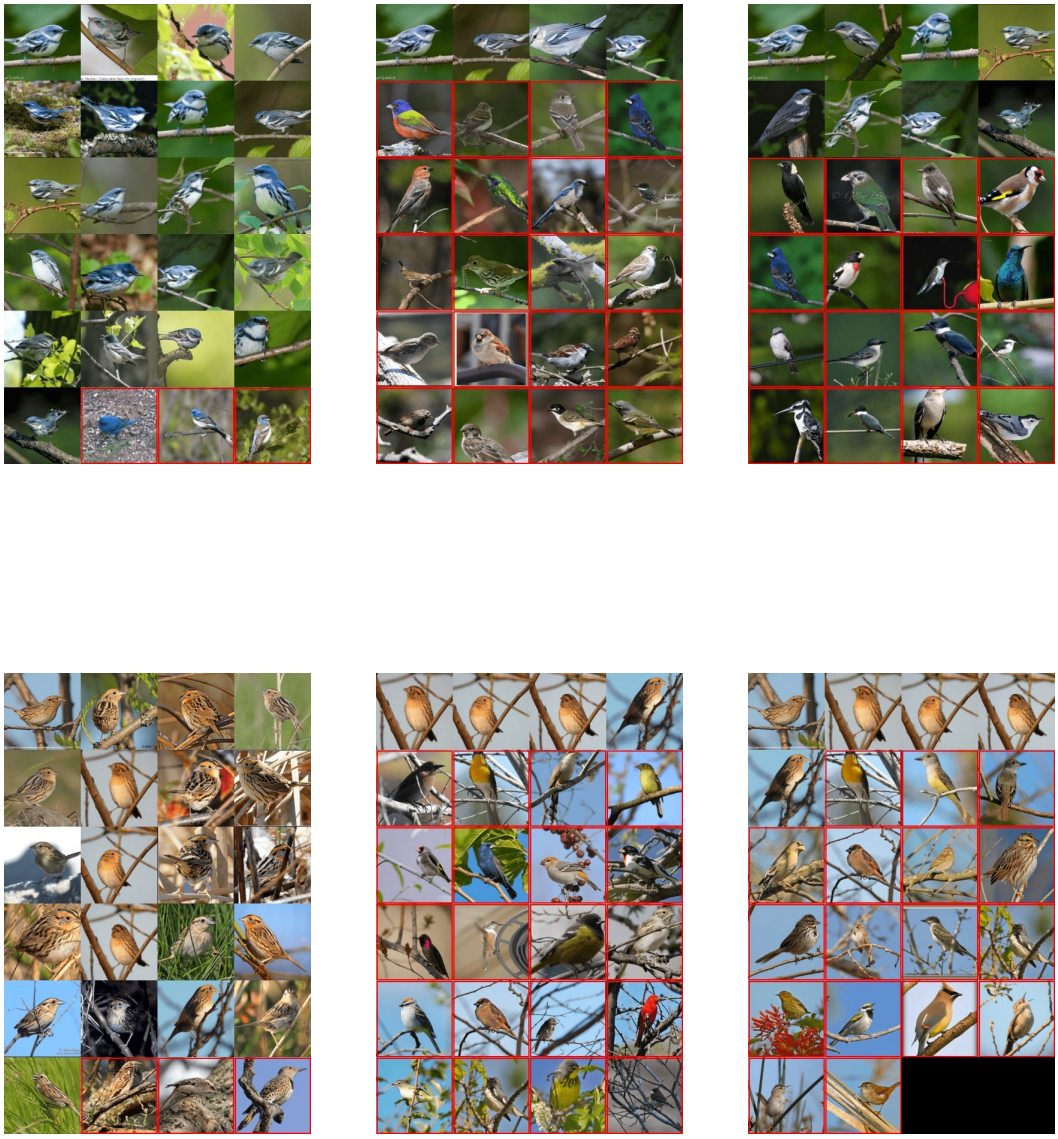
Table 7: Results of hyperparameter search for overclustering.

	Acc ↑				NMI ↑			
# of clusters	Bird	Car	Dog	Flower	Bird	Car	Dog	Flower
Actual number of classes × 3	26.3	14.1	17.9	67.8	0.51	0.41	0.36	0.67
100 (Underclustering)	-	-	-	-	0.38	0.25	0.32	0.75
500 (Overclustering)	23.7	12.1	20.5	70.8	0.51	0.40	0.39	0.65

Table 8: Arbitrary under and overclustering results when the number of clusters are unknown.

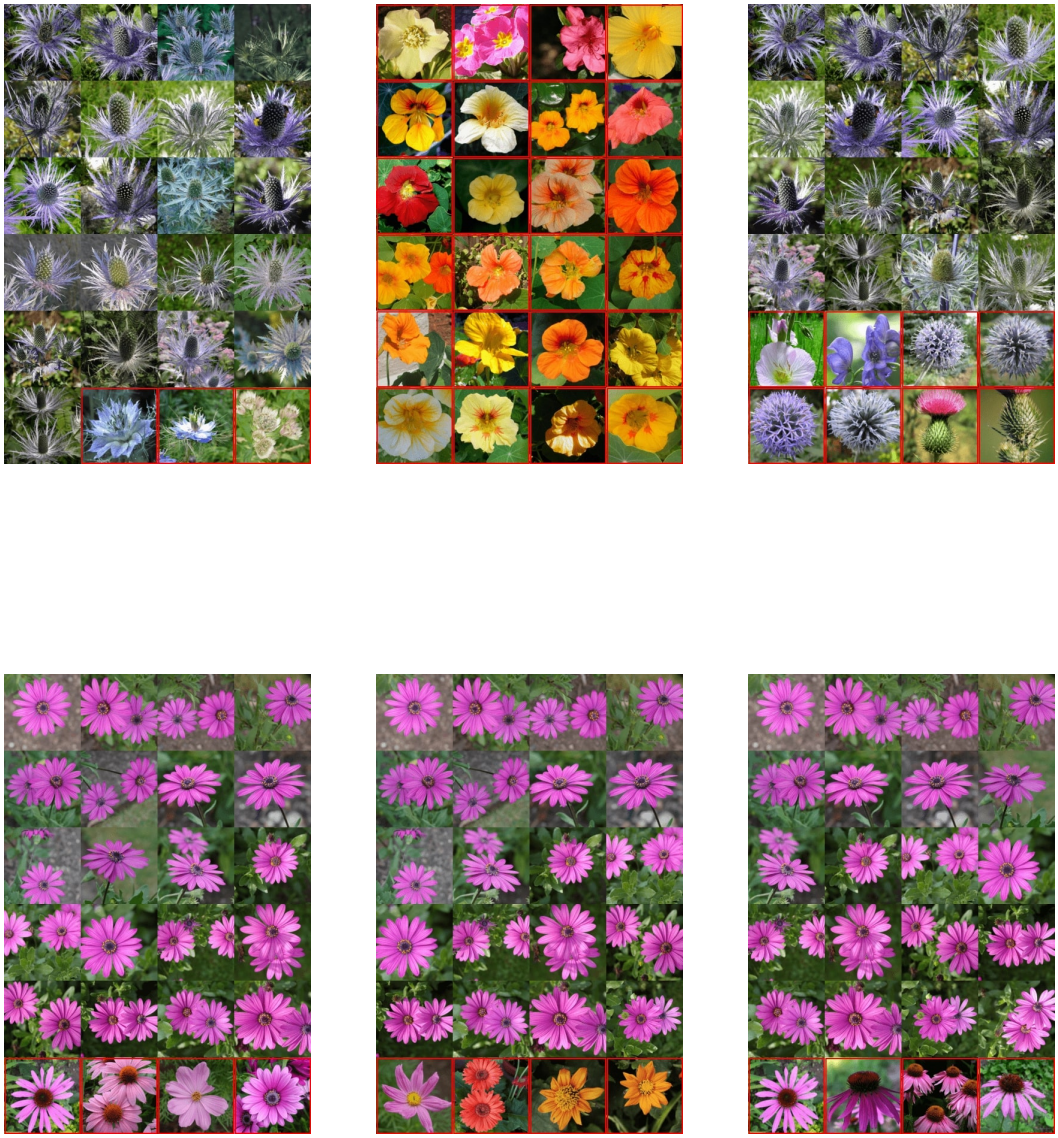
A.3 Qualitative Analysis of Clustering Results

We present the clustered images by our C3-GAN on two classes for CUB and Oxford Flowers datasets, comparing the results of MixNMatch and SCAN in Figs. 6 and 7. From Fig. 6, our method presents better clustered results outperforming two comparable methods. It is worth noting that even the objects in images incorrectly clustered by C3-GAN look very similar to given object class while the cluster results of other two methods contain images of objects that seldom resemble the given class object. We can confirm the consistent results on the Flower dataset as shown in Fig. 7.



(a) C3-GAN (Ours) (b) SCAN (Van Gansbeke et al., 2020) (c) MixNMatch (Li et al., 2020)

Figure 6: Examples of clustering results on CUB dataset. The cluster indices of the first and the second rows are the ones mapped from real classes *Cerulean Warbler* and *le Conte Sparrow* via Hungarian Algorithm. Red boxes denote incorrectly clustered images.

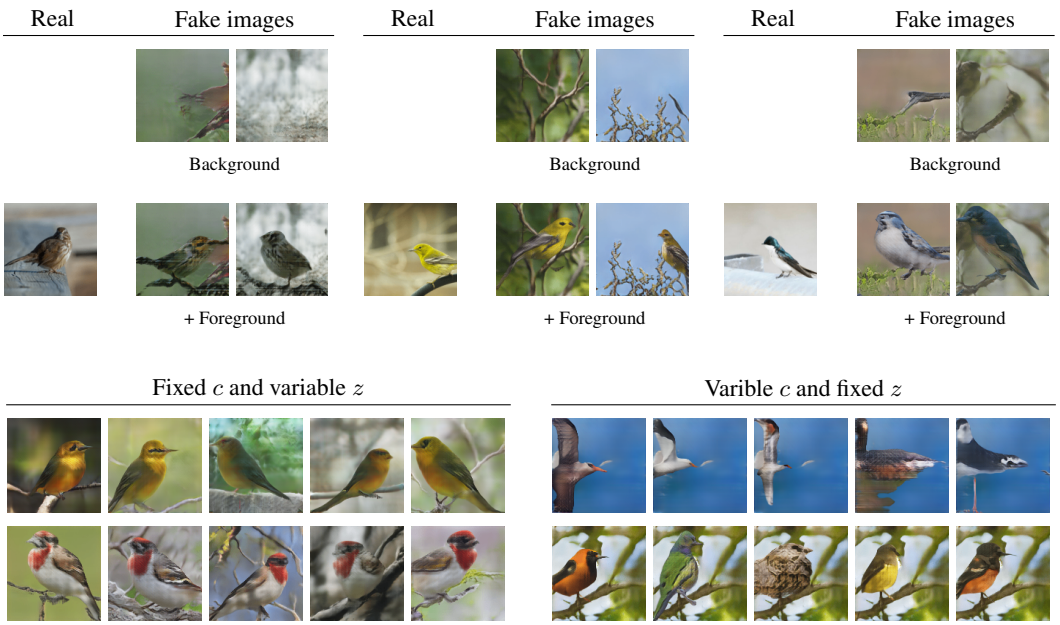


(a) C3-GAN (Ours) (b) SCAN (Van Gansbeke et al., 2020) (c) MixNMatch (Li et al., 2020)

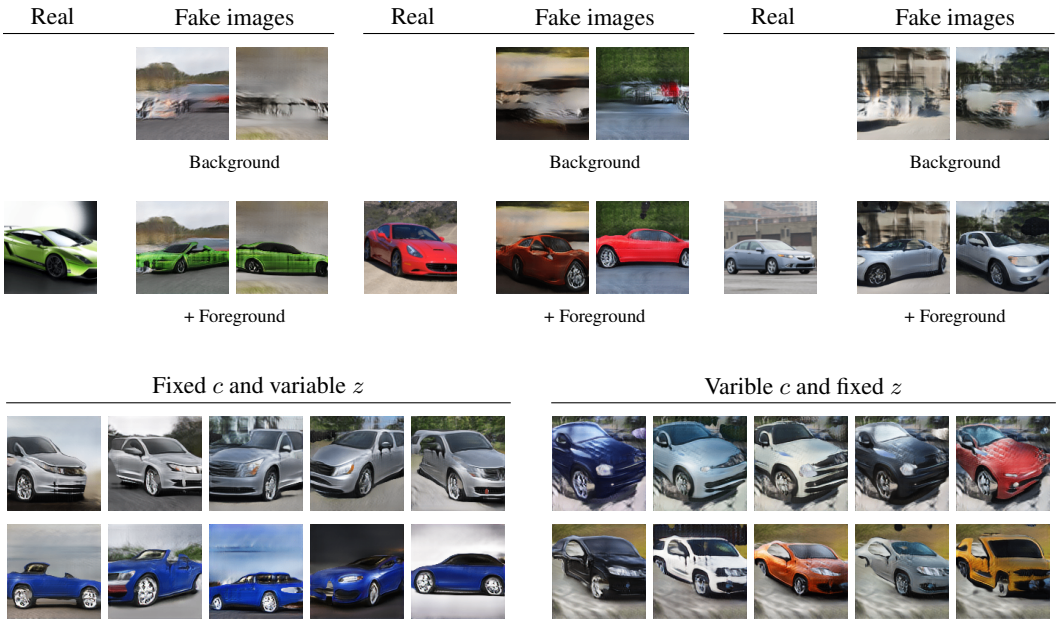
Figure 7: Examples of clustering results on Flower dataset. The cluster indices of the first and the second rows are the ones mapped from real classes *35* and *66* via Hungarian Algorithm. Red boxes denote incorrectly clustered images.

A.4 ADDITIONAL IMAGE GENERATION RESULTS

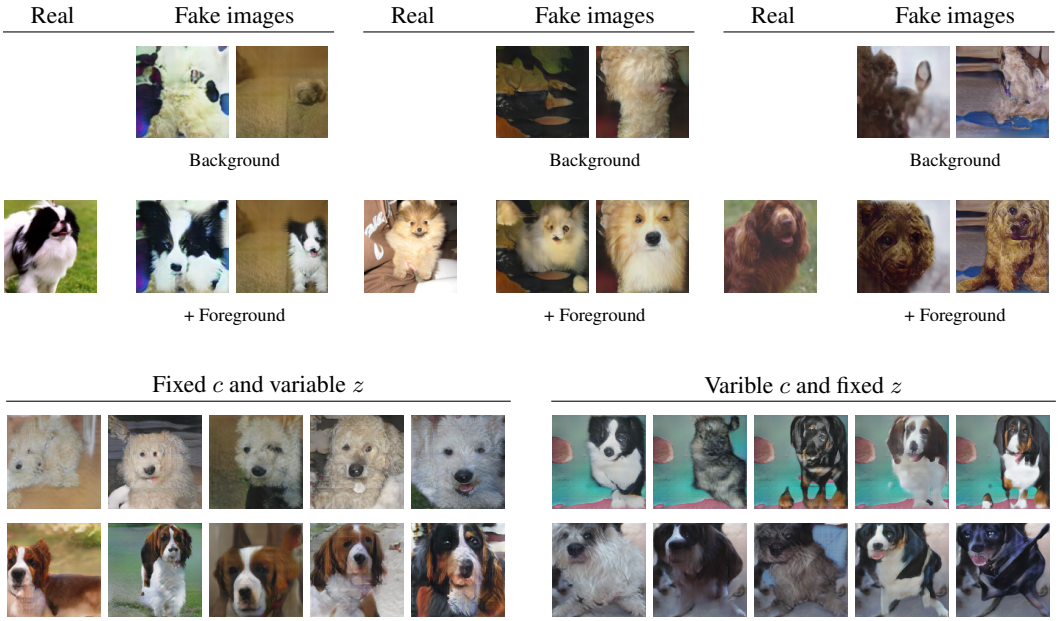
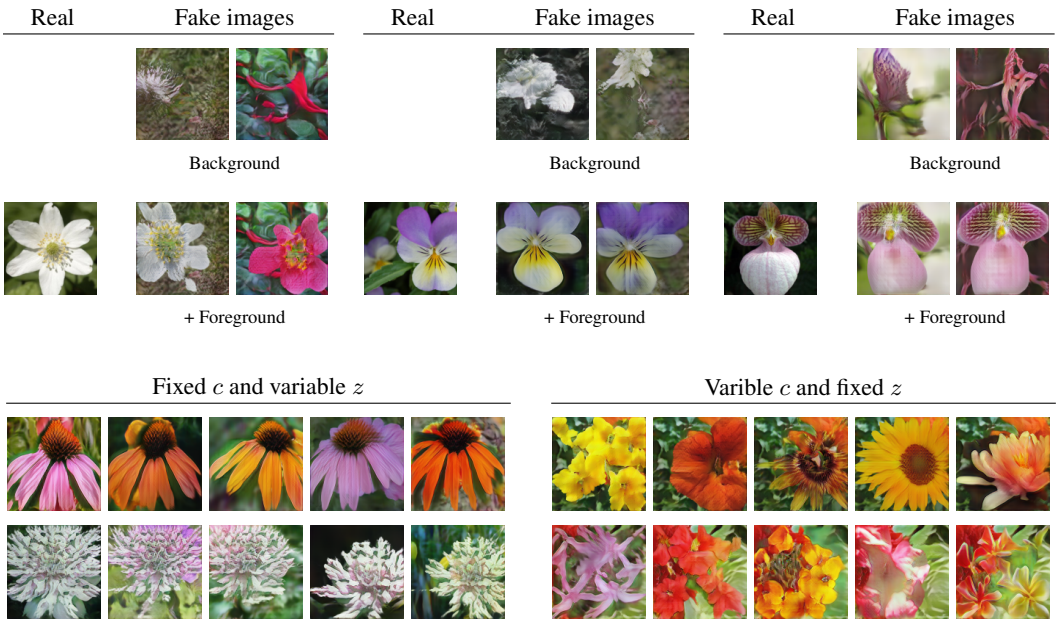
A.4.1 CUB



A.4.2 STANFORD CARS



A.4.3 STANFORD DOGS



A.4.4 OXFORD FLOWER