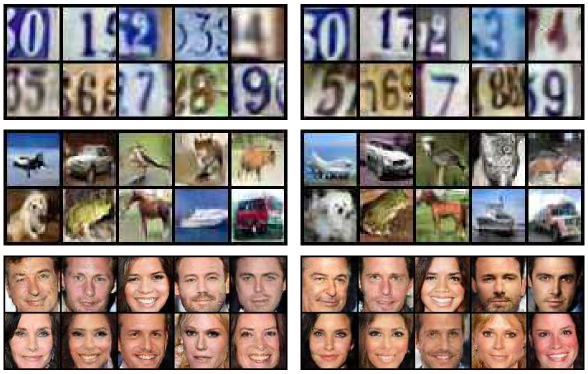
Regularizing Discriminative Capability of CGANs for Semi-Supervised Generative Learning

Yi Liu[1*], Guangchang Deng[1*], Xiangping Zeng[1], Si Wu[1,2†], Zhiwen Yu[1], and Hau-San Wong[2]
[1]School of Computer Science and Engineering, South China University of Technology
[2]Department of Computer Science, City University of Hong Kong
csyiliu@gmail.com, csgc@mail.scut.edu.cn, csxpzeng@gmail.com, cswusi@scut.edu.cn,
zhwyu@scut.edu.cn, cshswong@cityu.edu.hk

Abstract

Semi-supervised generative learning aims to learn the underlying class-conditional distribution of partially labeled data. Generative Adversarial Networks (GANs) have led to promising progress in this task. However, it still needs to further explore the issue of imbalance between real labeled data and fake data in the adversarial learning process. To address this issue, we propose a regularization technique based on Random Regional Replacement (R^3-regularization) to facilitate the generative learning process. Specifically, we construct two types of between-class instances: cross-category ones and real-fake ones. These instances could be closer to the decision boundaries and are important for regularizing the classification and discriminative networks in our class-conditional GANs, which we refer to as R^3-CGAN. Better guidance from these two networks makes the generative network produce instances with class-specific information and high fidelity. We experiment with multiple standard benchmarks, and demonstrate that the R^3-regularization can lead to significant improvement in both classification and class-conditional image synthesis.



(a) Baseline (ful. sup.) (b) R^3-CGAN (semi-sup.)

Figure 1. Synthesized images of a baseline model and R^3-CGAN on SVHN (*top*), CIFAR-10 (*middle*) and FaceScrub-100 (*bottom*). We adopt the SN-GAN [20] as our baseline, and perform fully supervised learning. For fair comparison, we adopt a similar network architecture for the generator as [16] [8] [40]. To highlight the effectiveness of the proposed model in semi-supervised generative learning, our R^3-CGAN is trained with 1k, 4k and 2k labels on the three benchmarks, respectively. The results suggest that R^3-CGAN is able to synthesize images having similar or even higher fidelity than the baseline with full supervision.

1. Introduction

Considerable progress has been recently made in synthesizing high-fidelity images. There are many deep generative models proposed to learn the underlying distribution of real data and synthesize new instances from random noise, such as Generative Adversarial Networks (GANs) [9] [6] [28] [20] [45] [4] [21]. Most existing GAN-based models were designed to perform unsupervised or fully supervised generative learning on a dataset.

In practice, one may also be interested in learning on partially labeled data, since this has a wide range of applications in which fully supervised data is difficult to acquire. How to leverage the large amount of unlabeled data is crucial for semi-supervised generative learning. Several models utilize a categorical discriminative network to simultaneously identify real instances and predict the corresponding class labels in the adversarial learning process, such as CatGAN [35], ImprovedGAN [32], and CT-GAN [39]. Considering that these two tasks of the discriminative network may be incompatible, another strategy is adopted by including an additional classification network into the minimax game, like Triple-GAN [16], Triangle-GAN [8] and EnhancedTGAN [40]. In those models, the generative and classification networks learn to produce two types of fake

[*] Joint first authors.
[†] Corresponding author.

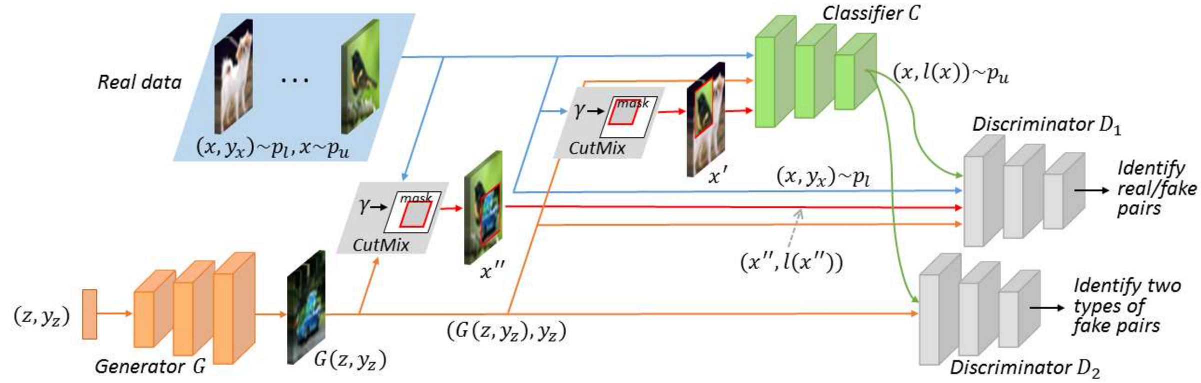


Figure 2. An overview of the proposed R^3-CGAN model for semi-supervised generative learning. There are four constituent networks: a generator G, a classifier C and two discriminators D_1 and D_2. To address the issue of insufficiency of labeled data, the random regional replacement strategy of CutMix is adopted to construct x' and x'', corresponding to cross-category instances and real-fake instances, respectively. Each pair of randomly selected images are combined by replacing a rectangular region of one image with that of the other. The replacement region is determined by a random variable γ having a beta distribution. We construct suitable training targets for x' and x'' to regularize the behaviors of C and D_1, thereby inducing G to synthesize instances with class-specific information and high fidelity.

instance-label pairs: real unlabeled instance with pseudo label, and synthesized instance with specified label. The discriminative network is trained to distinguish real instance-label pairs from these fake pairs. However, there are still two issues: (i) The limited supervision on class label prediction. The classification network often gives incorrect, yet confident predictions on unlabeled data. (ii) The imbalance between real and fake pairs. The discriminator tends to memorize the real labeled instances and reject unseen types of instances even from the distribution of true data. In this work, we address these two issues by proposing a regularization technique based on Random Regional Replacement (R^3-regularization) in the learning process of the classification and discriminative networks. Figure 1 indicates the possibility of semi-supervised class-conditional image synthesis comparable to the case of full supervision.

Specifically, we propose a Class-conditional Generative Adversarial Network with the R^3-regularization for improving semi-supervised generative learning, and our model is referred to as R^3-CGAN. An overview of the proposed approach is shown in Figure 2. We adopt the Triangle-GAN as our base model, and jointly train four players in the minimax game, including a generative network, a classification network, and two discriminative networks. We believe that the classification and discriminative networks play important roles in guiding the generative network to produce high-fidelity instances. To avoid the generalization capability of the classification and discriminative networks from being affected by the insufficiency of labeled data, we adopt the CutMix strategy [43] to construct two types of between-class instances: cross-category ones and real-fake ones. For both real labeled and unlabeled data, we perform replacement in a random rectangular region between different images. When the original images are from different classes, the constructed image can be complex and may be close to the decision boundaries. We further construct suitable training targets for the new instances to regularize the classification network, such that it has continuous and smooth predictions in-between the original instances. On the other hand, we also combine the real and synthesized instances in a similar way to regularize one of the discriminators. As a result, the generative network is able to receive better learning signals from both classification and discriminative networks. The extensive experiments confirm that the proposed R^3-CGAN is able to significantly outperform the competing methods in both image synthesis and classification. In summary, our main contribution is three-fold: (1) We analyze the issues caused by the insufficiency of labeled data in state-of-the-art semi-supervised generative models [16] [8] [40], and adopt an effective regularization strategy based on random regional replacement to address them. (2) Different from previous semi-supervised CGANs, we improve generative learning by regularizing the behaviors of the classification and discriminative networks on two types of constructed between-class instances and corresponding suitable training targets. (3) We carefully formulate the optimization problem of the constituent networks in our model to obtain an effective solution to our challenging generation tasks. We find that the proposed R^3-CGAN is able to achieve both state-of-the-art image synthesis and classification on multiple standard semi-supervised benchmarks.

2. Related Work

Convolutional Neural Networks (CNNs) have been successfully applied to many applications, in which the networks are often trained in a supervised fashion [14] [33] [30]. In the semi-supervised setting, there are a large amount of unlabeled data. To exploit these data, a number of existing works provide various strategies for constructing training targets for them [12] [31] [41] [17] [2]. The ladder network [29] is a well-known CNN architecture for semi-supervised learning. Similar to the Γ-model of the ladder network, Laine and Aila proposed the Π-model [15], in which each training sample was fed into a CNN twice, and the model is optimized by minimizing the prediction divergence. Adversarial perturbation in the inputs [23] [22] and model parameters [26] was used to maximally change the model's predictions, and penalizing the divergence can lead to better generalization performance. To make the training process more efficient, Laine and Aila also proposed a temporal ensembling model [15] to aggregate the prediction on each training sample over previous training epochs, and used the result as a training target to optimize the current model. In most cases, the aggregated result is more accurate than the current prediction, and the generalization performance can thus be significantly improved due to the better supervision. To avoid maintaining the aggregated results on the whole training set, Tarvainen and Valpola constructed a teacher network by aggregating the parameters of a classification network, and providing the teacher's predictions as the training targets for the original network [36]. Luo et al. utilized the teacher's predictions to construct a similarity graph of training samples for constraining representation learning [18]. Further, mutual learning [46] between two constituent networks was applied to semi-supervised learning [27] [42]. The networks are able to be mutually reinforced via providing training targets for each other.

Adopting GANs for class-conditional image synthesis is another direction of semi-supervised learning. To address the issue of insufficient labeled data, the GAN-based models aim at synthesizing high-fidelity instances, conditioned on the specified class labels. There are a number of conditional generative models, such as CGAN [19], CVAE [34] and CVAE-GAN [3]. For semi-supervised generative learning, Springenberg [35] proposed a Categorical Generative Adversarial Network (CatGAN). A categorical discriminative network was used to simultaneously distinguish real data from fake data, and predict the corresponding class labels. Salimans et al. [32] investigated several useful techniques for improving the training process, such as feature matching, historical averaging, etc. When incorporating them into model training, both semi-supervised image generation and classification performance can be improved. Further, Wei et al. [39] proposed a soft consistency term to enhance the Lipschitz continuity of the discriminative network in a Wasserstein GAN [1]. However, it may be incompatible for a single discriminative network to simultaneously distinguish real data from fake data and predict the corresponding class labels. In this situation, it would become more difficult to find a good solution in the two-player minimax game. Therefore, Li et al. [16] included an additional classification network in the adversarial learning process. In the resulting three-player game, a generative network along with the classification network synthesizes image-label pairs for fooling a discriminative network. Wu et al. [40] enhanced the generative network in learning class-conditional distributions by adopting feature-semantic matching between real and fake data. Gan et al. [8] further incorporated an additional discriminative network into the adversarial learning process to distinguish the two types of fake data pairs, such that unlabeled real data can be exploited to improve the generation performance.

This work focuses on semi-supervised generative learning. Triangle-GAN is used as our base model. However, the proposed approach is significantly different from the above GAN-based methods. We propose to improve class-conditional instance synthesis by regularizing the predictions of the classification and discriminative networks on constructed between-class instances. As a result, the generative network is able to receive better guidance from them. There are a few interpolation strategies proposed recently for constructing between-class instances, like MixUp [44] [37]. However, interpolating between real and fake images blurs the output image, which may confuse the discriminative network. Therefore, we consider that the CutMix strategy is more suitable for our purpose. Although CutMix was first proposed to combine labeled images, we extend it to adapt both real labeled (unlabeled) data and fake data. We include different regularization terms in the overall loss of the proposed model accordingly, and achieve significant improvement in both class-conditional synthesis and classification.

3. Proposed Approach

For semi-supervised instance synthesis, we adopt the Triangle-GAN as our base model. As shown in Figure 2, there are 4 players in the game-theoretic adversarial learning process: a generative network G, a classification network C, and two discriminative networks D_1 and D_2. The four constituent networks are parameterized by θ_G, θ_C, θ_{D_1} and θ_{D_2}, respectively. The generator G and classifier C are trained to synthesize two types of fake data in the form of instance-label pair: synthesized instances with specified class labels and real unlabeled instances with predicted class labels, respectively. D_1 learns to distinguish the pairs of real labeled instances and corresponding ground-truth labels from both types of fake pairs. Further, D_2 learns to identify the two types of fake data. We adopt a random regional replacement strategy to construct different between-class instances for enhancing C and D_1 separately. More accurate predictions on both object classes and real/fake classes in turn enhance the generation performance. In the following subsections, we will present the optimization formulation of the constituent networks in detail.

3.1. Semantics-guided Synthesis

The generator G learns to map a random vector z to a visually realistic image $G(z, y_z)$, conditioned on a specified class label y_z. The goal is to match the class-conditional distribution of synthesized data with that of real data. To produce instances as close to real instances as possible, an adversarial training term L^G_{adv} is defined as follows:

$$\begin{aligned} L^G_{adv} = \mathrm{E}_{z\sim p_z}\big[&\log\big(1 - D_1(G(z, y_z), y_z)\big) \\ &+ \log D_2(G(z, y_z), y_z)\big], \end{aligned} \tag{1}$$

where the random vector z is drawn from a prior p_z, and $D_1(\cdot,\cdot)$ ($D_2(\cdot,\cdot)$) denotes the predicted probability of an instance-label pair being from the real labeled data (fake data produced by the generator). Minimizing this term forces the generator to fool the two discriminators by synthesizing high-fidelity instances. However, the discriminators focus on the degree of similarity between real and synthesized instances within each class, while the discriminability of synthesized instances from different classes is often overlooked. To highlight the discrepancy between different classes, the generator should ensure that the classifier's predictions on the synthesized instances are consistent with the specified class labels. Therefore, we include a semantic regularization term into the optimization formulation of the generator G as follows:

$$\min_G \; L^G_{adv} + \mathrm{E}_{z\sim p_z}\big[\mathtt{CE}(y_z, C(G(z, y_z)))\big], \tag{2}$$

where $\mathtt{CE}$ denotes the cross entropy loss function, and $C(\cdot)$ represents the predicted class probability distribution for an input. Eq.(2) indicates that the generator can be guided by both the classifier and discriminators in the learning process. Therefore, we can improve the generation quality by enhancing these networks.

3.2. Improving the Classifier

A better classifier can capture more discriminative and robust information for identifying different classes, and thus guide the generator G to synthesize instances with class-specific information. To improve the classifier C in our framework, we propose to construct between-class instances. Toward this end, we apply the random regional replacement strategy of CutMix [43] on labeled data, and also extend CutMix to adapt unlabeled data, such that data augmentation can be performed without depending on the availability of ground-truth class labels.

Specifically, given two random instances x_a and x_b, we adopt CutMix to construct a new instance x' as follows:

$$\begin{aligned} x' &= \mathtt{CutMix}(x_a, x_b, \gamma) \\ &= M(\gamma) \odot x_a + \big(I - M(\gamma)\big) \odot x_b, \end{aligned} \tag{3}$$

where $I = 1^{W\times H}$, $M(\gamma) \in \{0,1\}^{W\times H}$ denotes a binary mask associated with a random variable $\gamma \sim \mathtt{Beta}(\alpha, \alpha)$, and $\odot$ represents element-wise multiplication. Both I and $M(\gamma)$ have the same resolution as the input images. To perform regional replacement, $M(\gamma)$ should indicate a rectangular region $B(\gamma)$ as follows:

$$M(\gamma)(u, v) = \begin{cases} 0, & \text{if } (u, v) \in B(\gamma), \\ 1, & \text{otherwise}, \end{cases} \tag{4}$$

where $B(\gamma)$ is determined by the top-left corner (u_0, v_0) and bottom-right corner $(u_0 + W\sqrt{1-\gamma}, v_0 + H\sqrt{1-\gamma})$. The coordinate (u_0, v_0) is randomly sampled on the image plane. When x_a and x_b are from different classes, x' is a between-class instance, and the corresponding training target is constructed as follows:

$$t(x') = \gamma t(x_a) + (1-\gamma) t(x_b), \tag{5}$$

where $t(\cdot)$ denotes the training target of an input. We specify different training targets for labeled and unlabeled instances as follows:

$$t(x) = \begin{cases} y_x, & \text{if } x \sim p_l, \\ f_{\bar{C}}(x), & \text{if } x \sim p_u, \end{cases} \tag{6}$$

where p_l (p_u) represents the distribution of real labeled (unlabeled) data, y_x denotes the ground-truth class label of x, $f_{\bar{C}}(\cdot)$ denotes the learnt representation for an input, and $\bar{C}$ denotes the ensemble of classifiers C over previous training epochs by performing exponential moving average on its parameters as follows:

$$\theta_{\bar{C}} \leftarrow \eta\theta_{\bar{C}} + (1-\eta)\theta_C, \tag{7}$$

where $\theta_{\bar{C}}$ denotes the parameters of the network $\bar{C}$, and η denotes the updating rate. We consider that the ensemble $\bar{C}$ is more stable and has better generalization performance. For unlabeled instances, instead of pseudo-labeling, we use the features associated with the last hidden layer as the training targets to mitigate the risk of error propagation.

The constructed instances, along with the original real instances and fake instances synthesized by the generator G, are used to train the classifier C, and the corresponding optimization formulation is presented as follows:

$$\begin{aligned} \min_C \; L^C_{adv} &+ \mathrm{E}_{z\sim p_z}\big[\mathtt{CE}(y_z, C(G(z, y_z)))\big] \\ &+ \mathrm{E}_{x'\sim p'_l}\big[\mathtt{CE}(t(x'), C(x'))\big] \\ &+ \mathrm{E}_{x'\sim p'_u}\big[\mathtt{MSE}(t(x'), f_C(x')\big], \end{aligned} \tag{8}$$

where $f_C(\cdot)$ represents the features associated with the last hidden layer of C, and the adversarial learning term L^C_{adv} is defined as follows:

$$\begin{aligned} L^C_{adv} = \mathrm{E}_{x\sim p_u}\big[&\mathtt{max}(C(x)) \log\big(1 - D_1(x, l(x))\big) \\ &+ \mathtt{max}(C(x)) \log(1 - D_2(x, l(x)))\big], \end{aligned} \tag{9}$$

and

$$l(x) = \texttt{one-hot}(C(x)), \text{ w.r.t } x \sim p_u. \quad (10)$$

In Eqs.(8-10), p'_l (p'_u) denotes the distribution of the constructed instances derived from real labeled (unlabeled) data, `MSE` is the mean square error function, and the function `one-hot` transforms a class probability distribution into a one-hot vector indicating the class that the corresponding sample should belong to. Training the classifier on the between-class instances forces it to have continuous and smooth predictions in between the classes. This helps to prevent the network from abruptly changing its output around the decision boundaries. In the 4-player game, the classifier C and generator G work cooperatively and compete with the discriminators D_1 and D_2.

3.3. Improving the Discriminators

To improve synthesis quality, a better discriminator is also required to provide guidance for the generator G. Considering the issue of imbalance between real and fake pairs, the R^3-regularization can also be applied to enhance the capability of the discriminator D_1. Different from Eq.(3), we construct new between-class instances by mixing real and synthesized instances as follows:

$$x'' = \texttt{CutMix}(x, G(z, y_z), \gamma). \quad (11)$$

In the case of $x \sim p_u$, the class label of x'' is determined as follows:

$$l(x'') = \texttt{one-hot}(\gamma l(x) + (1-\gamma) y_z). \quad (12)$$

For $x \sim p_l$, $l(x'')$ is computed by substituting the ground-truth label y_x for the pseudo label $l(x)$ in Eq.(12). The constructed pairs $(x'', l(x''))$ are fed into the discriminator D_1 to alleviate the imbalance of the original training pairs. To avoid confusing D_1 in the adversarial training process, we construct a binary training target for x'' as follows:

$$t(x'') = \begin{cases} 1, & \text{if } \gamma > 0.5, \\ 0, & \text{otherwise.} \end{cases} \quad (13)$$

Eq.(13) indicates that x'' is viewed as real if the region of real instance is greater than that of fake instance, and fake otherwise. After including an adversarial training term $L_{adv}^{D_1}$, we formulate the optimization problem of D_1 as follows:

$$\max_{D_1} L_{adv}^{D_1} + \mathrm{E}_{x'' \sim p''_s}\big[\texttt{CE}(t(x''), D_1(x'', l(x'')))\big], \quad (14)$$

where

$$\begin{aligned} L_{adv}^{D_1} = &\ \mathrm{E}_{x \sim p_l}\big[\log D_1(x, y_x)\big] \\ &+ \mathrm{E}_{x \sim p_u}\big[\log\big(1 - D_1(x, l(x))\big)\big] \\ &+ \mathrm{E}_{z \sim p_z}\big[\log\big(1 - D_1(G(z, y_z), y_z)\big)\big], \end{aligned} \quad (15)$$

p''_s represents the distribution of the constructed instances by using Eq.(11). In addition to distinguishing real data from the two types of fake data, D_1 is also encouraged to classify the constructed instances, which are more challenging than the original instances. On the other hand, the optimization formulation of D_2 is expressed as follows:

$$\begin{aligned} \max_{D_2} L_{adv}^{D_2} = &\ \mathrm{E}_{x \sim p_u}\big[\log\big(1 - D_2(x, l(x))\big)\big] \\ &+ \mathrm{E}_{z \sim p_z}\big[\log D_2(G(z, y_z), y_z)\big]. \end{aligned} \quad (16)$$

The discriminator D_2 is trained to identify the instance-label pairs produced by the generator and classifier. By competing with both discriminators D_1 and D_2, the generator G learns to synthesize instances matching the statistics of both real labeled and unlabeled instances.

4. Experiments

We present experimental results on the widely used standard benchmarks: SVHN [24] with 1k labels, CIFAR-10 [13] with 4k labels, CIFAR-100 [13] with 10k labels, and FaceScrub-100 [25] with 2k labels. We evaluate both semi-supervised image synthesis and classification on these datasets. Specifically, we first compare our R^3-CGAN with the main competing generative models including ImprovedGAN [32], Triple-GAN [16], Triangle-GAN [8] and EnhancedTGAN [40]. Note that the same network architecture is used in the generators of these models and our R^3-CGAN for fair comparison. To provide insights into what makes our model synthesize high-fidelity images, we conduct an investigation on the benefits of the proposed R^3-regularization. Furthermore, we perform extensive comparison with state-of-the-art methods in image classification.

4.1. Setting

In the experiments, the labeled instances are distributed equally across the classes in each dataset. We adopt the adversarial training process of Triangle-GAN in general, and train our model from scratch. The total number of training epochs is set to 400. Each mini-batch consists of 32 labeled samples, 128 unlabeled samples, and the same number of between-class samples constructed according to Eqs.(3-6). In addition, there are 128 samples synthesized by the generator from noise, and there are also 128 between-class samples constructed according to Eqs.(11-13). We adopt the Adam method [11] to optimize the constituent networks. We find that extensive tuning of the hyperparameters is not necessary to achieve state-of-the-art results on the test datasets. For the classifier, the learning rate μ_C is set to 0.1 during the first 350 epochs, and then is ramped down to 0 according to a Gaussian function. For the generator G and discriminators, both the learning rates μ_G and μ_D are set to 0.0003 during the first 350 epochs, respectively. After that, they are ramped down to 0 according to a linear function.

Table 1. Synthesis qualities of our R^3-CGAN and competing generative models on SVHN, CIFAR-10, CIFAR-100 and FaceScrub-100.

	SVHN (1k)		CIFAR-10 (4k)		CIFAR-100 (10k)		FaceScrub-100 (2k)	
Method	IS	FID	IS	FID	IS	FID	IS	FID
ImprovedGAN [32]	-	-	5.56±0.28	47.25	-	-	-	-
Triple-GAN [16]	-	-	5.77±0.14	47.08	-	-	-	-
Triangle-GAN [8]	2.75±0.02	36.56	6.56±0.07	35.31	-	-	-	-
EnhancedTGAN [40]	2.87±0.05	22.99	7.23±0.09	25.64	4.86±0.04	65.11	1.57±0.02	57.58
Baseline	2.66±0.02	45.03	6.57±0.06	37.21	4.29±0.06	72.39	1.66±0.03	31.21
R^3-CGAN	**2.99±0.02**	**10.87**	**7.42±0.05**	**20.34**	**7.49±0.01**	**26.29**	**1.73±0.02**	**25.28**

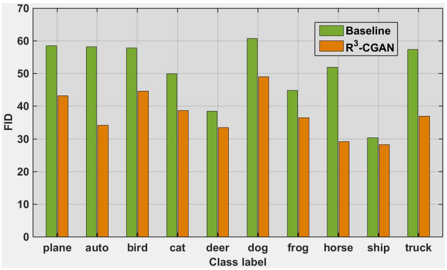


Figure 3. The FID score for each class on the synthesized CIFAR-10 data.

Table 2. Ablation experiment results of the proposed R^3-CGAN and variants on CIFAR-10 and CIFAR-100.

	CIFAR-10 (4k)		CIFAR-100 (10k)	
Method	IS	FID	IS	FID
Baseline	6.57±0.06	37.21	4.29±0.06	72.39
+ R^3-Reg. on D_1	**7.03±0.07**	**25.30**	**7.02±0.10**	**31.18**
Improvement	↑ 0.46	↓ 11.91	↑ 2.73	↓ 41.21
Baseline (ful. sup.)	7.07±0.08	26.49	7.11±0.06	32.39
+ R^3-Reg. on D_1	**7.78±0.07**	**17.98**	**7.83±0.14**	**23.45**
Improvement	↑ 0.71	↓ 8.51	↑ 0.72	↓ 8.94
R^3-CGAN	**7.42±0.05**	**20.34**	**7.49±0.01**	**26.29**
w/o R^3-Reg. on D_1	6.82±0.09	32.68	5.33±0.05	55.26
w/o R^3-Reg. on C	7.14±0.07	22.52	7.26±0.06	29.11

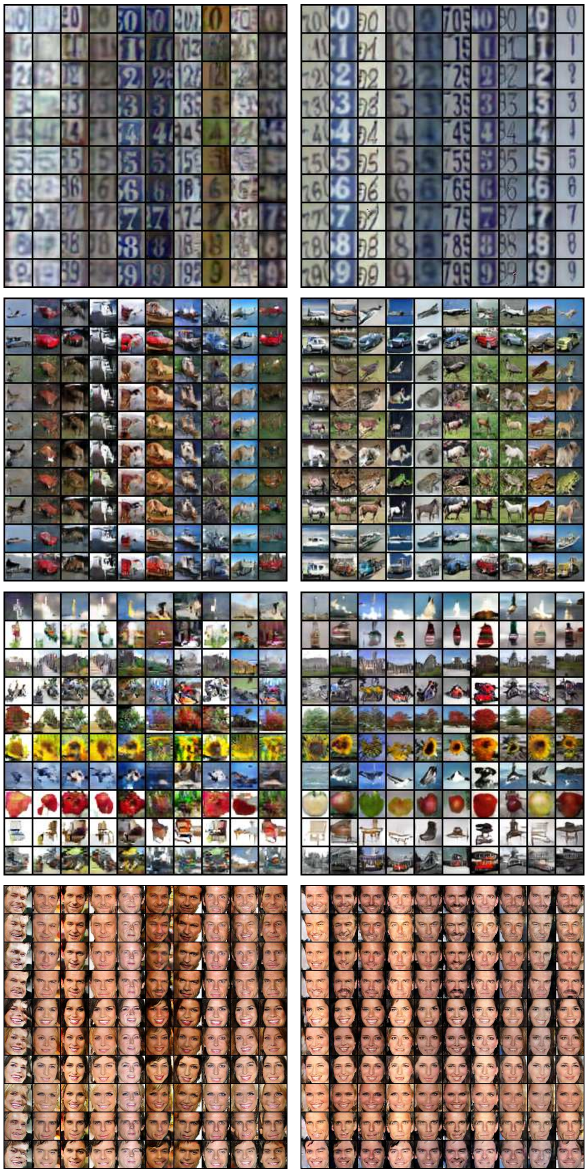
The momentum parameters β_1 and β_2 in Adam are fixed to 0.5 and 0.999, respectively. The hyperparameter α of CutMix is set to 0.2 for each task.

4.2. Comparison in Image Synthesis

We first conduct experiments on class-conditional image synthesis on the four benchmarks. The SVHN images contain digits with various backgrounds. The CIFAR-10 and CIFAR-100 images are from 10 and 100 natural object classes, respectively. In addition, the objects in FaceScrub-100 are human faces, and this dataset is constructed by selecting the 100 largest classes from FaceScrub [25]. We evaluate the synthesized images quantitatively via the Inception Score (IS) [32] and Fréchet Inception Distance (FID) [10]. These two criteria have been widely used to evaluate the diversity and realism of synthesized data. A larger IS and a lower FID indicates that a model can generate more diverse and realistic instances. Table 1 summarizes the results of our R^3-CGAN model and the competing generative models. To demonstrate the benefit of our R^3-regularization to semi-supervised generative learning, we build a baseline model by using the open-source code of the Triangle-GAN and train the model in the same configuration as ours. The results demonstrate that R^3-CGAN achieves significant improvement over the baseline model in synthesis quality. In particular, the gains reach 3.2 in IS and 46.1 in FID on CIFAR-100. In Figure 3, we show the improvement in FID per class on CIFAR-10. Our model also outperforms the competing methods on the four benchmarks. On CIFAR-100, R^3-CGAN also significantly outperforms EnhancedT-GAN, currently a state-of-the-art model, by 2.63 in IS and 38.82 in FID. On SVHN and FaceScrub-100, the FIDs of our model are 10.87 and 25.28 while those of EnhancedT-GAN are 22.99 and 57.58. We also visualize the images synthesized by our model and the baseline model in Figure 4. Our model is able to synthesize realistic images in specified classes, and the injected random vector encodes meaningful styles. In particular, we can observe better-resolved face images with reasonable structure and well preserved identities on FaceScrub-100 compared to the baseline, which confirms the outcome of Table 1.

4.3. Ablation Study

To highlight the effectiveness of our R^3-regularization in facilitating semi-supervised generative learning, we build a set of variants and test them on CIFAR-10 and CIFAR-100. Table 2 shows the evaluation results of the variants in synthesis quality. Compared to the baseline, including the R^3-regularization in the discriminative network D_1 indeed leads to significant improvement. On CIFAR-100, the IS increases from 4.29 to 7.02, and the corresponding FID decreases from 72.39 to 31.18. On CIFAR-10, the gain in FID is also very obvious. We also verify the effectiveness of the R^3-regularization in the case of full supervision. Since we can access the class labels of all training instances, the baseline model is equivalent to a SN-GAN [20]. We still observe that the R^3-regularization can lead to an improvement on each dataset in this case.



(a) Baseline (b) R^3-CGAN

Figure 4. Examples of the synthesized images produced by the baseline model and our R^3-CGAN on the benchmarks which from top to bottom are SVHN (1k), CIFAR-10 (4k), CIFAR-100 (10k) and FaceScrub-100 (2k). Each column shares the same random vector, and each row uses the same class label.

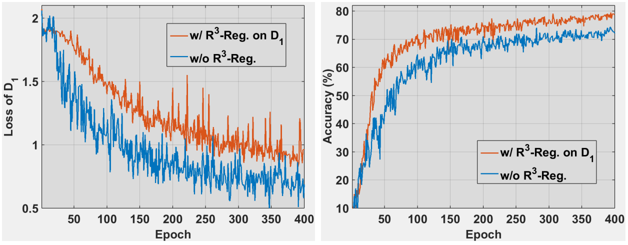


Figure 5. An experiment on CIFAR-10 (4k) to verify the effectiveness of the R^3-regularization on the discriminator D_1. The left subfigure shows that it is harder for D_1 to identify the complex instances constructed via CutMix. The right subfigure shows that more synthesized images are correctly classified by the classifier when applying this regularization, which indicates that the quality and discriminability of synthesized images are improved.

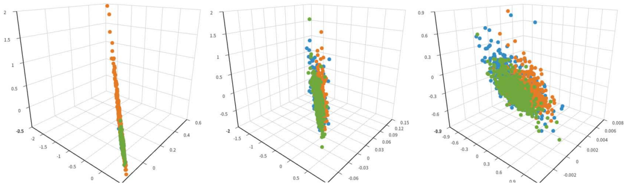


Figure 6. An experiment on CIFAR-10 (4k) to visualize the distributions of real data points (blue), synthesized data points (green) and CutMix-based constructed data points (orange). The embeddings of the 3 types of data points are plotted in epoch 50 (*left*), epoch 250 (*middle*) and epoch 350 (*right*).

On both CIFAR-10 and CIFAR-100, the results of our R^3-CGAN in the semi-supervised setting are better than those of the baseline with full supervision. To assess the relative contributions of the R^3-regularization to the final performance, we perform two ablation experiments by disabling the corresponding terms in the overall loss functions of the discriminative network D_1 and classification network C, respectively. A drop in performance can be observed in both cases, which indicates that enhancing C is also useful for improving synthesis quality as well as enhancing D_1.

4.4. Model Analysis

We perform further experiments on CIFAR-10 to obtain insight on how the R^3-regularization enhances the discriminative network in our framework. In Figure 5, we plot the loss values of D_1 for the cases with and without regularization in the training process. We can observe that the constructed complex instances lead to slow convergence of D_1. On the other hand, we also plot the accuracies of the classification network on the synthesized instances. Note that the classifier is optimized on the real data only in this experiment. Applying the R^3-regularization on D_1 provides the synthesized instances with more class-specific information, such that the classifier can correctly identify more of them.

In addition, we visualize the distributions of real instances, synthesized instances and CutMix-based constructed instances in the latent space associated with D_1 in Figure 6. Specifically, we include a fully-connected layer with 3 nodes between the last hidden layer and output without sacrificing its performance. We can observe that the distributions of real and synthesized data points can be matched. The constructed data points typically deviate from them, and induce D_1 to learn an embedding space in which the distributions become more dispersed.

In our R^3-regularization, the random variable γ is sam-

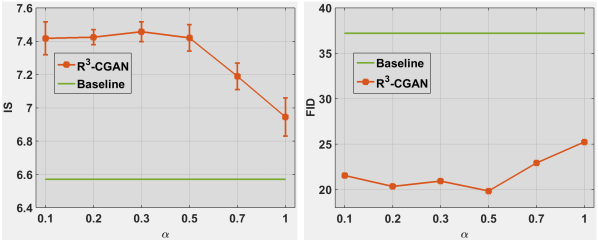


Figure 7. An experiment to investigate the impact of the hyperparameter α on the IS (*left*) and FID (*right*) on the synthesized CIFAR-10 data.

pled from a beta distribution $\mathtt{Beta}(\alpha, \alpha)$. The hyperparameter α controls the combination strength. We investigate the impact of different values of α from 0.1 to 1 on the synthesis quality in Figure 7. The results suggest that this regularization can lead to performance gains in all the cases, but the improvement is relatively stable when $\alpha \leq 0.5$.

4.5. Comparison in Image Classification

In addition to image synthesis, we also evaluate the proposed model in image classification. We compare our model with both state-of-the-art generative models and non-generative models. Note that we adopt a network architecture of CNN-13 for the classifier of our model. This architecture has also been used in most of the state-of-the-art works. The test error rates of our R^3-CGAN and the competing methods are shown in Tables 3-4. We can make the following observations: our R^3-CGAN significantly outperforms the baseline model by 2.50, 6.82, 3.29 and 17.07 percentage points on SVHN, CIFAR-10, CIFAR-100 and FaceScrub-100, respectively. These performance gains over the baseline model demonstrate the effectiveness of the R^3-regularization in enhancing the classification network. On all the benchmarks, R^3-CGAN achieves state-of-the-art results consistently. In particular, the proposed model substantially outperforms the second best model EnhancedTGAN, which demonstrates the superiority of our model in semi-supervised image classification on complex datasets.

Table 3. Test error rates (%) of the proposed R^3-CGAN and previous state-of-the-art methods on SVHN and CIFAR-10.

Method	SVHN (1k)	CIFAR-10 (4k)
Ladder Network [29]	-	20.40±0.47
SPCTN [41]	7.37±0.30	14.17±0.27
Π-model [15]	4.82±0.17	12.36±0.31
Temporal Ensemb. [15]	4.42±0.16	12.16±0.24
Mean Teacher [36]	3.95±0.19	12.31±0.28
VAT [23]	3.74±0.09	11.96±0.10
VAdD [26]	4.16±0.08	11.68±0.19
SNTG+Π-model [18]	3.82±0.25	11.00±0.13
Deep Co-Train [27]	3.61±0.15	9.03±0.18
CCN [42]	3.36±0.18	8.80±0.24
ICT [38]	3.89±0.04	7.29±0.02
CatGAN [35]	-	19.58±0.58
ImprovedGAN [32]	8.11±1.30	18.63±2.32
ALI [7]	7.42±0.65	17.99±1.62
Triple-GAN [16]	5.77±0.17	16.99±0.36
Triangle-GAN [8]	-	16.80±0.42
GoodBadGAN [5]	4.25±0.03	14.41±0.03
CT-GAN [39]	-	9.98±0.21
EnhancedTGAN [40]	2.97±0.09	9.42±0.22
Baseline	5.47±0.43	13.51±0.58
R^3-CGAN	**2.97±0.05**	**6.69±0.28**

Table 4. Test error rates (%) of the proposed R^3-CGAN and previous state-of-the-art methods on CIFAR-100 and FaceScrub-100.

Method	CIFAR-100 (10k)	FaceScrub-100 (2k)
Π-model [15]	39.19±0.36	23.72±0.19
Temporal Ensemb. [15]	38.65±0.51	22.38±0.16
SNTG+Π-model [18]	37.97±0.29	-
Deep Co-Train [27]	34.63±0.14	-
CCN [42]	35.28±0.23	-
EnhancedTGAN [40]	36.18±0.37	16.08±0.24
Baseline	35.95±0.30	24.03±0.55
R^3-CGAN	**32.66±0.21**	**6.96±0.43**

5. Conclusion

We present a class-conditional GAN-based model for improving semi-supervised generative learning. To address the issue of imbalance between real labeled data and fake data in the adversarial learning process, we adopt the strategy of random regional replacement to construct two types of between-class instances, cross-category ones and real-fake ones, respectively. Training our model on the extended data can effectively regularize the behaviors of the classification and discriminative networks, thereby inducing the generative network to synthesize instances with more class-specific information and high fidelity. The experimental results demonstrate the effectiveness of our R^3-regularization and superior performance of the R^3-CGAN model in both class-conditional image synthesis and classification.

The training process of CGANs can be sensitive to many aspects. To generate high-resolution images, more advanced model architectures and training techniques are needed. In our future work, we would consider how to apply the proposed R^3-regularization to other CGANs for more complex tasks.

Acknowledgments

This work was supported in part by the Research Grants Council of the Hong Kong Special Administration Region (Project No. CityU 11300715), in part by the National Natural Science Foundation of China (Project No. U1611461, 61722205, 61751205, 61572199), in part by City University of Hong Kong (Project No. 7005055), in part by the Natural Science Foundation of Guangdong Province (Project No. 2016A030310422, 2016A030308013), in part by Fundamental Research Funds for the Central Universities (Project No. 2018ZD33), and in part by National Undergraduate Innovative and Entrepreneurial Training Program (Project No. 201910561075).

References

[1] M. Arjovsky, S. Chintala, and L. Bottou. Wasserstein generative adversarial networks. In *Proc. International Conference on Machine Learning*, 2017.

[2] B. Athiwaratkun, M. Finzi, P. Izmailov, and A. Wilson. There are many consistent explanations of unlabeled data: why you should average. In *Proc. International Conference on Learning Representation*, 2019.

[3] J. Bao, D. Chen, F. Wen, H. Li, and G. Hua. CVAE-GAN: fine-grained image generation through asymmetric training. In *Proc. International Conference on Computer Vision*, 2017.

[4] A. Brock, J. Donahue, and K. Simonyan. Large scale GAN training for high fidelity natural image synthesis. In *Proc. International Conference on Learning Representation*, 2018.

[5] Z. Dai, Z. Yang, F. Yang, W. Cohen, and R. Salakhutdinov. Good semi-supervised learning that requires a bad GAN. In *Proc. Neural Information Processing Systmes*, 2017.

[6] E. Denton, S. Chintala, and R. Fergus. Deep generative image models using a Laplacian pyramid of adversarial networks. In *Proc. Neural Information Processing Systmes*, 2015.

[7] V. Dumoulin, I. Belghazi, B. Poole, O. Mastropietro, A. Lamb, M. Arjovsky, and A. Courville. Adversarially learned inference. In *Proc. International Conference on Learning Representation*, 2017.

[8] Z. Gan, L. Chen, W. Wang, Y. Pu, Y. Zhang, H. Liu, C. Li, and L. Carin. Triangle generative adversarial networks. In *Proc. Neural Information Processing Systmes*, 2017.

[9] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Proc. Neural Information Processing Systmes*, 2014.

[10] M. Heusel, H. Ramsauer, T. Unterthiner, and B. Nessler. GANs trained by a two time-scale update rule converge to a local Nash equilibrium. In *Proc. Neural Information Processing Systmes*, 2017.

[11] D. Kingma and J. Ba. Adam: a method for stochastic optimization. In *Proc. International Conference on Learning Representation*, 2015.

[12] D. Kingma, S. Mohamed, D. Rezende, and M. Welling. Semi-supervised learning with deep generative models. In *Proc. Neural Information Processing Systmes*, pages 3581 – 3589, 2017.

[13] A. Krizhevsky and G. Hinton. Learning multiple layers of features from tiny images. In *Tech. Rep., Univ. Toronto, Toronto, ON, Canada*, 2009.

[14] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Proc. Neural Information Processing Systmes*, 2014.

[15] S. Laine and T. Aila. Temporal ensembling for semi-supervised learning. In *Proc. International Conference on Learning Representation*, 2017.

[16] C. Li, K. Xu, J. Zhu, and B. Zhang. Triple generative adversarial nets. In *Proc. Neural Information Processing Systmes*, 2017.

[17] C. Li, J. Zhu, and B. Zhang. Max-margin deep generative models for (semi-)supervised learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(11):2762 – 2775, 2018.

[18] Y. Luo, J. Zhu, M. Li, Y. Ren, and B. Zhang. Smooth neighbors on teacher graphs for semi-supervised learning. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[19] M. Mirza and S. Osindero. Conditional generative adversarial nets. In *arXiv:1411.1784*, 2014.

[20] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. In *Proc. International Conference on Learning Representation*, 2018.

[21] T. Miyato and M. Koyama. cGANs with projection discriminator. In *Proc. International Conference on Learning Representation*, 2018.

[22] T. Miyato, S. Maeda, S. Ishii, and M. Koyama. Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(8):1979 – 1993, 2018.

[23] T. Miyato, S. Maeda, M. Koyama, K. Nakae, and S. Ishii. Distributional smoothing with virtual adversarial training. In *Proc. International Conference on Learning Representation*, 2016.

[24] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Ng. Reading digits in natural images with unsupervised feature learning. In *Proc. Neural Information Processing Systmes Workshop*, 2011.

[25] H. Ng and S. Winkler. A data-driven approach to cleaning large face datasets. In *Proc. IEEE International Conference on Image Processing*, 2014.

[26] S. Park, J. Park, S. Shin, and I. Moon. Adversarial dropout for supervised and semi-supervised learning. In *Proc. AAAI Conference on Artificial Intelligence*, 2018.

[27] S. Qiao, W. Shen, Z. Zhang, B. Wang, and A. Yuille. Deep co-training for semi-supervised image recognition. In *Proc. European Conference on Computer Vision*, 2018.

[28] A. Radford, L. Metz, and S. Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. In *Proc. International Conference on Learning Representation*, 2016.

[29] A. Rasmus, M. Berglund, M. Honkala, H. Valpola, and T. Raiko. Semi-supervised learning with ladder networks. In *Proc. Neural Information Processing Systmes*, 2015.

[30] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6):1137 – 1149, 2017.

[31] M. Sajjadi, M. Javanmardi, and T. Tasdizen. Regularization with stochastic transformations and perturbations for deep semi-supervised learning. In *Proc. Neural Information Processing Systmes*, pages 1163 – 1171, 2016.

[32] T. Salimans, I. Goodfellow, W. Zaremba, and V. Cheung. Improved techniques for training GANs. In *Proc. Neural Information Processing Systmes*, 2016.

[33] E. Shelhamer, J. Long, and T. Darrell. Fully convolutional networks for semantic segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(4):640 – 651, 2017.

[34] K. Sohn, X. Yan, and H. Lee. Learning structured output represenation using deep conditional generative models. In *Proc. Neural Information Processing Systmes*, 2015.

[35] J. Springenberg. Unsupervised and semi-supervised learning with categorical generative adversarial networks. In *Proc. International Conference on Learning Representation*, 2016.

[36] A. Tarvainen and H. Valpola. Mean teachers are better role models: weight-averaged consistency targets improve semi-supervised deep learning results. In *Proc. Neural Information Processing Systmes*, 2017.

[37] V. Verma, A. Lamb, C. Beckham, A. Najafi, L. Mitliagkas, A. Courville, D. Lopez-Paz, and Y. Bengio. Manifold mixup: better representations by interpolating hidden states. In *Proc. International Conference on Machine Learning*, 2019.

[38] V. Verma, A. Lamb, J. Kannala, Y. Bengio, and D. Lopez-Paz. Interpolation consistency training for semi-supervised learning. In *Proc. International Joint Conference on Artificial Intelligence*, 2019.

[39] X. Wei, B. Gong, Z. Liu, W. Lu, and L. Wang. Improving the improved training of Wasserstein GANs: a consistency term and its dual effect. In *Proc. International Conference on Learning Representation*, 2018.

[40] S. Wu, G. Deng, J. Li, R. Li, Z. Yu, and H. Wong. Enhancing TripleGAN for semi-supervised conditional instance synthesis and classification. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[41] S. Wu, Q. Ji, S. Wang, H. Wong, Z. Yu, and Y. Xu. Semi-supervised image classification with self-paced cross-task networks. *IEEE Transactions on Multimedia*, 20(4):851–865, 2018.

[42] S. Wu, J. Li, C. Liu, Z. Yu, and H. Wong. Mutual learning of complementary networks via residual correction for improving semi-supervised classification. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[43] S. Yun, D. Han, S. Oh, S. Chun, J. Choe, and Y. Yoo. CutMix: regularization strategy to train strong classifiers with localizable features. In *Proc. International Conference on Computer Vision*, 2019.

[44] H. Zhang, M. Cisse, Y. Dauphin, and D. Lopez-Paz. mixup: Beyond empirical risk minimization. In *Proc. International Conference on Learning Representation*, 2018.

[45] H. Zhang, I. Goodfellow, D. Metaxas, and A. Odena. Self-attention generative adversarial networks. In *arXiv:1705.05512*, 2018.

[46] Y. Zhang, T. Xiang, T. Hospedales, and H. Lu. Deep mutual learning. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2018.