OSSGAN: Open-Set Semi-Supervised Image Generation

Kai Katsumata   Duc Minh Vo   Hideki Nakayama

The University of Tokyo, Japan

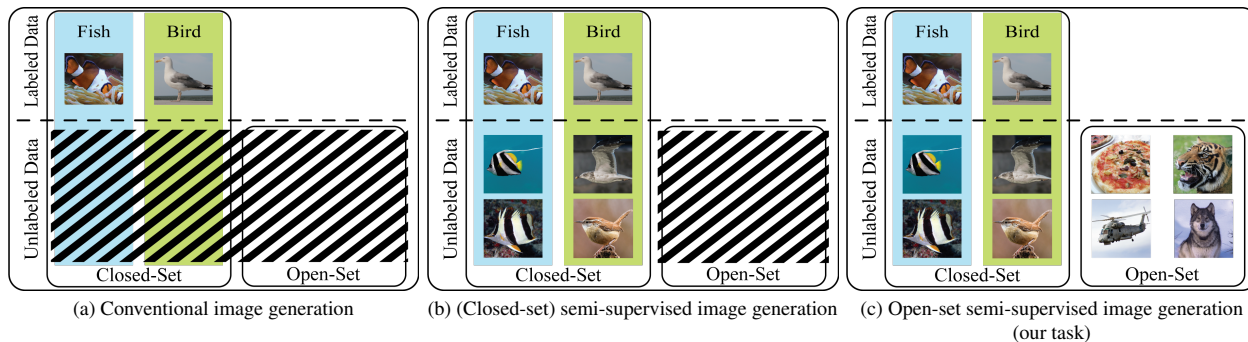{katsumata,vmduc,nakayama}@nlab.ci.i.u-tokyo.ac.jp

Figure 1. We explore the conditional image generation in which we relax the assumption about training data. (a) Conventional supervised image generation [1, 12, 23] principally uses labeled data for training. (b) (Closed-set) semi-supervised image generation [10] generalizes the supervised image generation, allowing the presence of unlabeled data with samples inside classes of interest, which are closed-set samples. (c) Our task of open-set semi-supervised image generation further generalizes semi-supervised image generation. The unlabeled data contain closed-set samples and open-set samples, which do not belong to any of the classes of interest.

Abstract

We introduce a challenging training scheme of conditional GANs, called open-set semi-supervised image generation, where the training dataset consists of two parts: (i) labeled data and (ii) unlabeled data with samples belonging to one of the labeled data classes, namely, a closed-set, and samples not belonging to any of the labeled data classes, namely, an open-set. Unlike the existing semi-supervised image generation task, where unlabeled data only contain closed-set samples, our task is more general and lowers the data collection cost in practice by allowing open-set samples to appear. Thanks to entropy regularization, the classifier that is trained on labeled data is able to quantify sample-wise importance to the training of cGAN as confidence, allowing us to use all samples in unlabeled data. We design OSSGAN, which provides decision clues to the discriminator on the basis of whether an unlabeled image belongs to one or none of the classes of interest, smoothly integrating labeled and unlabeled data during training. The results of experiments on Tiny ImageNet and ImageNet show notable improvements over supervised Big-GAN and semi-supervised methods. Our code is available

at *https://github.com/raven38/OSSGAN*.

1. Introduction

The outstanding performance of the SoTA conditional generative adversarial networks (cGANs) [1,12,23] is heavily reliant on having access to a vast amount of labeled data during training (Fig. 1a). This dependence necessitates significant efforts to label the data and limits the applications of cGANs in real-world scenarios. Reducing the reliance on labeled data in training cGANs is thus deemed necessary.

Semi-supervised image generation [4, 9, 10, 19, 20] allows the appearance of both labeled and unlabeled data during training, with the unlabeled data primarily containing within classes of interest (closed-set samples) (Fig. 1b). Despite the advances, the unlabeled data assumption is at odds with the fact that the majority of unlabeled data is outside of classes of interest (open-set samples), and ensuring that unlabeled data do not contain open-set samples is often costly and prone to error. In fact, in [4, 9, 10, 19, 20], even open-set samples are classified into classes that appear in labeled data, resulting in cGAN performance deterioration.

We go beyond semi-supervised image generation by al-

lowing the use of unlabeled data gathered miscellaneously to reduce the effort of labeling and introduce a novel task of open-set semi-supervised image generation (Fig. 1c). Unlabeled data contain open-set samples, and the conditional generator should produce images that are indistinguishable from real ones even when trained on both labeled and unlabeled data. Unlike the conventional semi-supervised fashion, the task allows unlabeled data to have the category set mismatched with labeled data, reducing the required labeling effort. Our task is a significant step towards real-world data, which contains labeled data and unlabeled data (both closed-set and open-set samples), lowering the data construction cost and expanding the range of real-world applications of cGANs.

To address our new task, we design **O**pen-**S**et **S**emi-supervised GAN (OSSGAN). We simultaneously train cGAN and a classifier that assigns labels to unlabeled data. By incorporating entropy regularization into the cross-entropy loss, the classifier quantifies the confidence of the prediction to enable the discriminator to use unlabeled data, including open-set samples, smoothly. Consequently, OSSGAN allows the natural integration of unlabeled data into cGANs without explicitly eliminating open-set samples.

The results of empirical experiments demonstrate that OSSGAN effectively utilizes unlabeled data, including open-set samples. More importantly, we achieve better performance in terms of FID and other metrics against strong supervised and semi-supervised baselines. Notably, our method achieves a performance comparable to that of Big-GAN [1], which has up to five times as many labeled samples as OSSGAN. Furthermore, the experiments with different degrees of an open-set sample ratio show that the proposed method is robust to miscellaneous data. Qualitative experimental results also reveal the superiority of our method. Our contributions are summarized as follows.

- We propose a novel open-set semi-supervised image generation task, which is based on a relaxed assumption in the case of building a dataset at a reasonable cost.

- We design OSSGAN, thanks to entropy regularization, smoothly using closed- and open-set samples in unlabeled data in cGAN training.

- We demonstrate the superiority of the proposed method over baselines on several benchmarks with limited labeled data in terms of quantitative metrics such as FID. Our qualitative experiments also show that our method achieves better generation quality.

2. Related work

CGANs [12] are a GANs extension, which learns a conditional generative distribution. CGANs can deal with many types of conditions such as class label [12], text description [24], or another image [7]. For well-constructed datasets, [1, 13, 23] are proposed to improve quality, fidelity, and training stability. Among them, self-attention GAN [23] and BigGAN [1] outperform other GANs with hundreds of classes. The progress in network architectures, optimization algorithms, and the quality and quantity of datasets support high-fidelity image generation. We aim to achieve high-fidelity image generation without a well-constructed dataset.

Image generation with data constraints is aimed at improving the generation quality without using enormous amounts of data. Collecting a large labeled dataset requires a tremendous annotation cost. To achieve better performance within limited resources (*i.e.*, time and money), several studies [6, 8, 18, 25] contribute to the data-efficient aspect of cGANs. Another line of studies [2, 10] focusing on the fact that unlabeled images are easier to collect than labeled images employs semi-supervised or unsupervised fashion. For semi-supervised image generation, [19, 20] employ an unconditional discriminator for unlabeled data, and [4, 9] take pseudo labels by employing a classifier. Unlabeled data have been efficiently utilized in self-supervised learning [10]. A different aspect in the study of semi-supervised learning and GAN concerns semi-supervised recognition tasks that employ GAN for generating pseudo samples [3, 17]. Unsupervised image generation frees us from tedious annotation labor. However, unsupervised methods do not control generated outcomes as semi-supervised methods do. In this study, we utilize unlabeled data gathered miscellaneously for training cGANs to reduce the data construction cost, which will further broaden their range of application.

Open-set semi-supervised recognition has the same objective as our method but addresses a totally different task. The goal of the recognition task is to build a model distinguishing open-set samples using a dataset consisting of labeled data with only closed-set classes and unlabeled data with both closed- and open-set classes. The joint optimization of classification and open-set sample detection models [22] and the consistent regularization with data augmentation [11] are used to tackle the problem. In contrast to recognition paradigms aimed at separating explicitly open- and closed-set samples, our generation task does not necessarily require explicit separation of these samples. Instead of applying a method for detecting open-set samples, we investigate a method of utilizing open-set samples.

3. Open-set semi-supervised image generation

3.1. Task definition

While labeling data is tremendously costly, we can collect unlabeled data at a relatively low cost. However, as

the semi-supervised image generation task does not allow the unlabeled data to contain any open-set samples, filtering out such open-set samples is mandatory. As a step towards reducing the manpower expended in the filtering process for building closed-set semi-supervised data, we consider a learning framework without the assumption that the sets of classes are shared between labeled and unlabeled data, meaning that open-set samples are freely applicable. While the aim of our task is to generate the images with the classes of interest (labeled data) similarly to supervised and semi-supervised image generation [1,10,13,23], we assume the unlabeled data contain closed-set and open-set samples (Fig. 1).

Let $\boldsymbol{x}, \boldsymbol{u} \in \mathbb{R}^d$ be real labeled and unlabeled samples, respectively, where $d = H \times W \times 3$ is the dimension of a sample where H and W are the height and width of the image. Let $y \in \mathcal{Y} = \{1, 2, \ldots, K\}$ be the class where K is the number of known classes. For the sake of simplicity, we will use a probability vector $\boldsymbol{y} \in \{\boldsymbol{y} \in \mathbb{R}^K | \sum_1^K = 1 \text{ and } 0 \leq \boldsymbol{y}_i\}$ when referring to the class unless otherwise specified. Thus, class y can be interpreted as a one-hot vector $\boldsymbol{y} = [\{\mathbb{1}[y = j]\}_{j=1}^K]^\mathsf{T}$. We denote the distribution of labeled data as $p(\boldsymbol{x}, \boldsymbol{y})$, whereas the distribution of unlabeled data is $p(\boldsymbol{u})$. The unlabeled data contain the samples that cannot be classified into one of the K classes, and such samples are considered as open-set samples.

We are given a set of labeled data $\mathcal{D}_l = \{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^{n_l} \sim p(\boldsymbol{x}, \boldsymbol{y})$ with n_l samples and a set of unlabeled data $\mathcal{D}_u = \{\boldsymbol{u}_i\}_{i=1}^{n_u} \sim p(\boldsymbol{u})$ with n_u samples as training data. By using the latent variable $\boldsymbol{z} \in \mathbb{R}^l$ with l being the dimension of the latent variable, the prior distribution is $q(\boldsymbol{z}, \boldsymbol{y}) = q(\boldsymbol{z})q(\boldsymbol{y})$ with a Gaussian distribution $q(\boldsymbol{z})$ and a uniform distribution $q(\boldsymbol{y})$ on $\{\boldsymbol{e}^{(1)}, \boldsymbol{e}^{(2)}, \ldots, \boldsymbol{e}^{(K-1)}, \boldsymbol{e}^{(K)}\}$, where $\boldsymbol{e}^{(i)}$ is the i-th standard basis vector of $\mathbb{R}^K$. The task of open-set semi-supervised image generation is to learn a generator $G : \mathbb{R}^l \times \mathbb{R}^K \to \mathbb{R}^d$ such that G takes a latent variable and a label $\boldsymbol{y} \sim q(\boldsymbol{y})$ and generates images that are indistinguishable from real ones for a given label.

3.2. General training objective functions

To accomplish the task we proposed, the discriminator should be able to deal with the unlabeled data (and the labeled data), that is, it should easily distinguish between closed-set samples and open-set samples. A straightforward way to do this is to use an auxiliary classifier to assign a class to unlabeled samples. Motivated by this observation, we define a cGAN with an auxiliary classifier as follows. Given a generator G, a discriminator D, and an auxiliary classifier C, the general discriminator loss $\mathcal{L}_D$ and generator loss $\mathcal{L}_G$ are defined as

$$\mathcal{L}_D = \mathcal{L}_{\text{adv}}^{\text{lbl}} + \mathcal{L}_{\text{adv}}^{\text{unlbl}} + \lambda \mathcal{L}_{\text{cls}} \tag{1}$$

$$\mathcal{L}_G = \mathbb{E}_{(\boldsymbol{z}, \boldsymbol{y}) \sim q(\boldsymbol{z}, \boldsymbol{y})}[-D(G(\boldsymbol{z}, \boldsymbol{y}), \boldsymbol{y})], \tag{2}$$

where $\mathcal{L}_{\text{adv}}^{\text{lbl}}$ and $\mathcal{L}_{\text{adv}}^{\text{unlbl}}$ are adversarial losses for labeled and unlabeled data, respectively. The classifier loss $\mathcal{L}_{\text{cls}}$ has a hyperparameter λ for balancing the loss terms. The $\mathcal{L}_{\text{adv}}^{\text{lbl}}$ follows the conventional adversarial loss with hinge loss $f_D(\cdot) = \max(0, 1 + \cdot)$ for labeled data:

$$\mathcal{L}_{\text{adv}}^{\text{lbl}} = \mathbb{E}_{(\boldsymbol{x}, \boldsymbol{y}) \sim p(\boldsymbol{x}, \boldsymbol{y})}[f_D(-D(\boldsymbol{x}, \boldsymbol{y}))]$$
$$+ \mathbb{E}_{(\boldsymbol{z}, \boldsymbol{y}) \sim q(\boldsymbol{z}, \boldsymbol{y})}[f_D(D(G(\boldsymbol{z}, \boldsymbol{y}), \boldsymbol{y}))]. \tag{3}$$

Naively, the classifier loss $\mathcal{L}_{\text{cls}}$ can be cross-entropy loss:

$$\mathcal{L}_{\text{cls}} = -\mathbb{E}_{(\boldsymbol{x}, \boldsymbol{y}) \sim p(\boldsymbol{x}, \boldsymbol{y})}[\log p_c(\boldsymbol{y}^\mathsf{T} \boldsymbol{s} | \boldsymbol{x})]. \tag{4}$$

We note that the discriminator D and the classifier C share the feature extractor part. Mathematically, $D : \mathbb{R}^d \times \mathbb{R}^K \to \mathbb{R}$ takes the form $D(\boldsymbol{x}, \boldsymbol{y}) = W_1 \tilde{D}(\boldsymbol{x}) + b + \tilde{D}(\boldsymbol{x})^\mathsf{T} W_2 \boldsymbol{y}$, $\tilde{D} : \mathbb{R}^d \times \mathbb{R}^K \to \mathbb{R}^h$ is a feature extractor in D, $\boldsymbol{s}$ is $[0, 1, 2, \ldots, K-1]^\mathsf{T}$, and $p_c(y|\boldsymbol{x}) = C_y(\tilde{D}(\boldsymbol{x}))$ is the y-th output of the auxiliary classifier $C : \mathbb{R}^h \to \mathbb{R}^K$. Here, $W_1 \in \mathbb{R}^{h \times 1}$ is the weight parameter of the fully connected layer, $W_2 \in \mathbb{R}^{h \times K}$ is the embedding matrix for a label, and h is the dimension of the extracted feature. The scalar representation of a label is represented by $\boldsymbol{y}^\mathsf{T} \boldsymbol{s}$ with a one-hot vector $\boldsymbol{y}$. We will further adaptively customize $\mathcal{L}_{\text{adv}}^{\text{lbl}}, \mathcal{L}_{\text{adv}}^{\text{unlbl}}$, and $\mathcal{L}_{\text{cls}}$ for different methods, as discussed later.

4. Proposed method

4.1. Threshold-based method

We introduce two baseline methods for our task, called RejectGAN and OpensetGAN. These methods are semi-supervised GANs extended by employing a classifier to assign predicted classes to unlabeled samples as new labels. RejectGAN only considers labeled samples and closed-set samples in unlabeled data in the training of GANs by filtering out unlabeled samples that have low confidence, as open-set samples. OpensetGAN explicitly utilizes open-set samples in the unlabeled data by assigning $K+1$ classes, the open-set class, to unlabeled samples with low confidence. Below, we describe the details of these methods.

RejectGAN. The principal purpose of this method is to train cGANs on the dataset with sufficient volume and clean labels. Whenever the labels of samples are available, we use them to update an auxiliary classifier in the same manner as ACGAN [14]. At the same time, we assign the predicted classes to unlabeled samples when the classifier predictions have a probability associated with a predicted class equal to or higher than a threshold. Then, by eliminating unlabeled samples with a probability associated with the predicated class less than a threshold, we train the discriminator only with unlabeled samples with high confidence, labeled samples, and samples synthesized by a generator. Accordingly, we modify the adversarial loss for unlabeled samples as

$$\mathcal{L}_{\text{adv}}^{\text{unlbl}} = \mathbb{E}_{\boldsymbol{u} \sim p(\boldsymbol{u})}[f_D(-D(\boldsymbol{u}, \hat{\boldsymbol{y}})) | \max(\tilde{\boldsymbol{y}}) \geq c], \tag{5}$$

where c is a threshold of confidence, $\tilde{\boldsymbol{y}} = C(\tilde{D}(\boldsymbol{u}))$ is a predicted probability vector, and $\hat{\boldsymbol{y}} = \boldsymbol{e}^{(\arg\max \tilde{\boldsymbol{y}})}$ is a predicted label with $\boldsymbol{e}^{(i)}$ being the i-th standard basis vector of $\mathbb{R}^K$. A confidence score larger than the threshold means that the sample is clearly classified into $y \in \mathcal{Y}$. We ignore the samples with a confidence score lower than the threshold in calculating the loss.

OpensetGAN. In addition to learning class-specific features, this method is also aimed at learning class-invariant image features by adding a novel class for the open-set samples to the discriminator inputs. In contrast to RejectGAN, the discriminator $D : \mathbb{R}^d \times \mathbb{R}^{K+1} \to \mathbb{R}$ takes a sample and a label vector with the length of $K + 1$ to consider open-set samples detected by the auxiliary classifier. Accordingly, OpensetGAN extends the adversarial losses $\mathcal{L}_{\text{adv}}^{\text{lbl}}$ and $\mathcal{L}_G$ to accept the $K + 1$-dimensional condition vector:

$$\mathcal{L}_{\text{adv}}^{\text{lbl}} = \mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})\sim p(\boldsymbol{x},\boldsymbol{y})}[f_D(-D(\boldsymbol{x}, \boldsymbol{y}^\mathsf{T} \boldsymbol{I}_{K,K+1}))]$$
$$+ \mathbb{E}_{(\boldsymbol{z},\boldsymbol{y})\sim q(\boldsymbol{z},\boldsymbol{y})}[f_D(D(G(\boldsymbol{z},\boldsymbol{y}), \boldsymbol{y}^\mathsf{T} \boldsymbol{I}_{K,K+1}))], \quad (6)$$
$$\mathcal{L}_G = \mathbb{E}_{(\boldsymbol{z},\boldsymbol{y})\sim q(\boldsymbol{z},\boldsymbol{y})}[-D(G(\boldsymbol{z},\boldsymbol{y}), \boldsymbol{y}^\mathsf{T} \boldsymbol{I}_{K,K+1})], \quad (7)$$

where $\boldsymbol{I}_{K,K+1} = (\boldsymbol{I}_K \ \boldsymbol{0})$ is the $K \times (K+1)$ rectangular identity matrix with the $K \times K$ identity matrix $\boldsymbol{I}_K$ and the zero vector $\boldsymbol{0}$. OpensetGAN assigns the $K+1$ classes to unlabeled data to train on the labeled data with the K classes, and then the adversarial loss for unlabeled samples is

$$\mathcal{L}_{\text{adv}}^{\text{unlbl}} = \mathbb{E}_{\boldsymbol{u}\sim p(\boldsymbol{u})}[f_D(-D(\boldsymbol{u}, \hat{\boldsymbol{y}}))]. \quad (8)$$

Here, we define the classifier output $\hat{\boldsymbol{y}} \in \mathbb{R}^{K+1}$ as

$$\hat{\boldsymbol{y}} = \begin{cases} \boldsymbol{e}^{(\arg\max \tilde{\boldsymbol{y}})}, & \text{if } \max(\tilde{\boldsymbol{y}}) \geq c \\ \boldsymbol{e}^{(K+1)}, & \text{otherwise}, \end{cases} \quad (9)$$

and we denote the K known classes and an additional open-set class by $\{\boldsymbol{e}^{(1)}, \boldsymbol{e}^{(2)}, \ldots, \boldsymbol{e}^{(K+1)}\}$ with the standard basis vectors of $\mathbb{R}^{K+1}$. The convert matrix $\boldsymbol{I}_{K,K+1}$ adds the zero-filled $K+1$-th column to a label vector for a discriminator. In Eq. (9), we assign one of the known classes to unlabeled samples with high confidence score and the open-set class to unlabeled samples with low confidence score.

4.2. The intuition behind OSSGAN

The methods described above do not efficiently exploit a given dataset because employing a threshold hinders the training of GAN. Furthermore, their performance is sensitive to the threshold, and the average entropy of predicted probabilities shifts with each iteration. In the training phase, finding the optimal threshold in each iteration is difficult because we do not know the ratio of known to unknown samples in unlabeled samples. As a result, because it eliminates closed-set samples with low confidence score in unlabeled data, RejectGAN misses out on useful information for training a generator and discriminator. Similarly, OpensetGAN
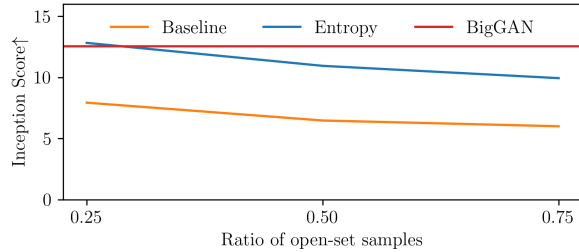

Figure 2. Effect of assigning high entropy labels to open-set samples on Tiny ImageNet. Entropy indicates the method of verifying our approach. BigGAN is trained on only labeled samples. Here, the experiments contain enough labeled samples for training cGANs. The method assigning high entropy labels to open-set samples maintains the performance of BigGAN.

may fail to learn the class feature owing to misclassifying known class samples as open-set samples. Since these flaws cause the instability of threshold-based methods, we require a more robust method that does not need careful tuning.

To devise a threshold-free method, we use the entropy of the label as a confidence score and feed continuous labels with confidence into the discriminator. Then, the discriminator uses the information that samples with low entropy are clearly classified into the known classes, and samples with high entropy are not classified into the known classes, resulting in no unlabeled samples being missed.

To ensure that the idea of assigning high entropy labels to open-set samples is effective for the task, we compare the method of manually assigning high entropy labels (as an oracle) to open-set samples with supervised and semi-supervised methods. The supervised method is BigGAN [1]. The semi-supervised method, which is referred to as Baseline in Fig. 2, assigns one of the known classes to open-set samples with an auxiliary classifier. BigGAN is only trained on closed-set samples. The high entropy method, which is abbreviated as Entropy, is trained on closed-set and open-set samples labeled $[1/K, 1/K, 1/K, \ldots, 1/K]^\mathsf{T}$. Entropy achieves a performance comparable to that of BigGAN in the case of a ratio of open-set samples of 25%, as shown in Fig. 2. It outperforms the baseline method in all other cases. These findings indicate that the strategy of assigning high entropy labels to open-set class samples helps to save training cGANs from contamination by open-set samples.

4.3. OSSGAN

To automatically assign high entropy labels to open-set samples in unlabeled data, we propose a method that quantifies the likelihood that a sample belongs to any one of the closed-set classes and feeds the samples with likelihoods into the discriminator. OSSGAN is made up of three parts: a generator, a discriminator, and an auxiliary classi-

fier (Fig. 3). To train the auxiliary classifier, we use both real labeled and generated samples. This is due to the fact that if the number of training data is insufficient, the auxiliary classifier performs poorly. To train the discriminator, we use real labeled, real unlabeled, and generated samples. While either the label of the real labeled samples or the label of the generated samples can be fed directly into the discriminator, we use the classifier output as the label of the unlabeled samples. As a result, when the samples are labeled or generated, the discriminator $D : \mathbb{R}^d \times \mathbb{R}^K \to \mathbb{R}$ is fed a one-hot vector; when the samples are unlabeled, it is fed a continuous vector. Our adversarial loss for unlabeled samples is defined as

$$\mathcal{L}_{\text{adv}}^{\text{unlbl}} = \mathbb{E}_{\boldsymbol{u} \sim p(\boldsymbol{u})}[f_D(-D(\boldsymbol{u}, C(\tilde{D}(\boldsymbol{u}))))]. \quad (10)$$

To provide more informative clues for identifying open-set samples from closed-set ones to the discriminator, we introduce the entropy regularization term into the cross-entropy loss. While the standard classification loss, cross-entropy loss, results in low entropy, entropy regularization maintains high entropy of open-set samples in unlabeled data. With the term maximizing the average entropy of the auxiliary classifier outputs, we customize $\mathcal{L}_{\text{cls}}$:

$$\begin{aligned}
\mathcal{L}_{\text{cls}} = & - \mathbb{E}_{(\boldsymbol{x},\boldsymbol{y}) \sim p(\boldsymbol{x},\boldsymbol{y})}[\log p_c(\boldsymbol{y}^\mathsf{T}\boldsymbol{s}|\boldsymbol{x}))] \\
& - \mathbb{E}_{(\boldsymbol{x},\boldsymbol{y}) \sim p(\boldsymbol{x},\boldsymbol{y})}[H(C(\tilde{D}(\boldsymbol{x})))] \\
& - \mathbb{E}_{(\boldsymbol{z},\boldsymbol{y}) \sim q(\boldsymbol{z},\boldsymbol{y})}[\log p_c(\boldsymbol{y}^\mathsf{T}\boldsymbol{s}|G(\boldsymbol{z},\boldsymbol{y})))] \\
& - \mathbb{E}_{(\boldsymbol{z},\boldsymbol{y}) \sim q(\boldsymbol{z},\boldsymbol{y})}[H(C(\tilde{D}(G(\boldsymbol{z},\boldsymbol{y}))))]. \quad (11)
\end{aligned}$$

Without the regularization, the classifier predicts low entropy values to all unlabeled samples, resulting in assigning a known class even to unlabeled samples that should be treated as an open set. The entropy term is defined by

$$H(\boldsymbol{x}) = -\frac{\Sigma_{x \in \boldsymbol{x}} x \log x}{\log K}. \quad (12)$$

The range of the function is $0 \leq H(\boldsymbol{x}) \leq 1$. We use a normalized function because it is difficult to set the hyperparameter λ in the loss function when a function with a range of $0 \leq H(\boldsymbol{x})$ can take a large value. The contribution of the entropy regularization term varies with different K in the experiment if we use the original entropy function. While the cross-entropy loss makes the entropy smaller, the entropy regularization makes the entropy larger. The cross-entropy loss more strongly affects closed-set samples, resulting in the clear separation between the closed- and open-set samples. The number of labeled samples in the minibatch is quite small because unlabeled samples make up the majority of the dataset. To avoid a classifier that considers only the generated samples, which dominate the minibatch, we balance the ratio of the classification loss terms for labeled and generated samples by the number of labeled samples. Finally, the overall objective function of OSSGAN is $\mathcal{L}_G$ of Eq. (2) and $\mathcal{L}_D$ consisting of Eqs. (3), (10), and (11).
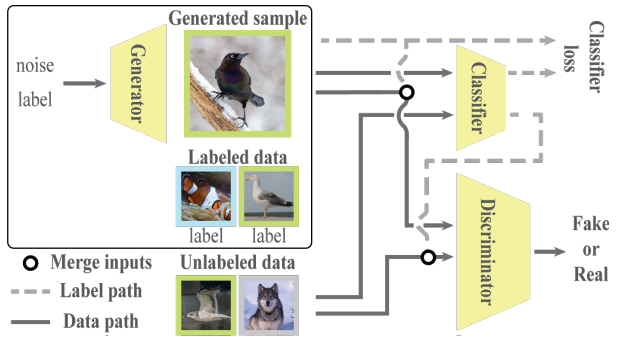

Figure 3. Overview of the proposed method. A classifier is trained on labeled and generated data with the classifier loss consisting of cross-entropy and entropy regularization terms and infers labels for unlabeled data. The generator is trained in the same fashion as conventional cGANs. For the generated and labeled data, the discriminator is trained in the same way as conventional cGANs. It further takes unlabeled samples by regarding them as labeled samples.

This method employs the raw probability vector for handling open-set samples instead of using thresholds and is free from investigating the optimal threshold. It also makes use of the inter-class similarity and class-invariant visual attributes that an auxiliary classifier acquires throughout the training process. For example, if the dog, cat, and monkey classes are given as known classes and a cow image is included in the open-set sample, the image can be used as an image with common mammalian attributes.

4.4. Implementation details

We choose BigGAN [1] as an example to verify OSSGAN. Here, our method can be applied to different cGANs (*e.g.*, SAGAN [23] and SNGAN [13]). In detail, we build all the methods that are used in the experiments upon BigGAN [1] by integrating DiffAugment [25].

For the experiments, we use the hierarchical latent space with 20 dimensions for each latent variable and the shared embedding with $d_{\boldsymbol{z}} = 128$. We use minibatch sizes of 1024 and 256 for the resolutions of 64×64 and 128×128, respectively. The learning rates are 1×10^{-4} and 4×10^{-4} for the generator and discriminator, respectively.

5. Experiments setting

Datasets. Using existing entirely labeled datasets, we create partially labeled datasets for benchmarking open-set semi-supervised image generation. The dataset construction procedure is divided into three stages (Fig. 4). We use three constants: the number of closed-set classes, the ratio of labeled samples in closed-set class samples, and the usage ratio in open-set samples. First, we divide the entirely labeled dataset into closed-set classes and open-set classes
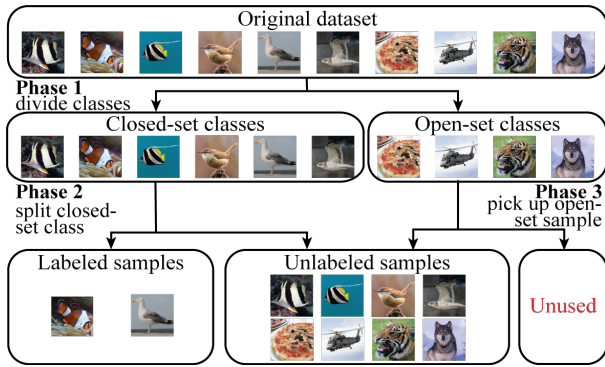
Figure 4. Procedure of building open-set semi-supervised datasets. The procedure has three constants for governing the number of labeled, closed-set unlabeled, and open-set unlabeled samples.
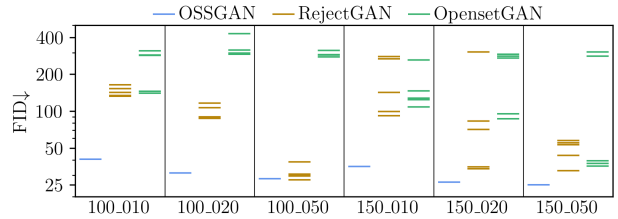

Figure 5. Effect of the threshold for detecting open-set class on Tiny ImageNet. We only report the constant scores for OSSGAN, as it does not employ a threshold. For OpensetGAN and Reject-GAN, the optimal thresholds are investigated in the range from 0 to 1. Our method achieves a lower (better) FID than the threshold-based methods because of the difficulty of selecting a threshold. The data configuration of 100_010 indicates 100 closed classes and 10% labeled samples. The same notations are applicable to other data configurations.

based on the number of closed-set classes. Then, using the ratio of labeled samples in closed-set class samples, we split the closed-set samples into labeled and unlabeled samples. Finally, we select samples from open-set samples based on the usage ratio in open-set samples, and we merge unlabeled samples in closed-set samples with the samples selected from open-set class samples.

We use the Tiny ImageNet [21], which consists of 200 diverse categories. Each class contains 500 and 100 images for training and testing, respectively. We use the number of closed-set classes of $\{150, 100, 50\}$, the ratio of the labeled samples in closed-set class samples of $\{0.95, 0.5, 0.2, 0.1\}$, and the usage ratio in open-set class samples of 1. Then, using the above constants, we create 12 data configurations. The easiest case has labeled samples with approximately three-quarters of the dataset, and the hardest case has labeled samples with less than 3% of the dataset.

We also use ImageNet ILSVRC2012 [15]. It consists of 1000 categories and provides 1.2 million images. For the dataset, we use the number of closed-set classes of 50, the ratio of the labeled samples in closed-set class samples of 0.2, and the usage ratio in open-set class samples of 0.1. The subset consists of around 12,000 labeled images and around 200,000 unlabeled images.

Compared methods. In the experiments, we select Big-GAN [1] with DiffAugment [25] as a base model and carefully integrate (open-set) semi-supervised methods into it. We compare OSSGAN with BigGAN [1], RandomGAN, SingleGAN, S^3GAN [10], RejectGAN, and OpensetGAN. Here, DiffAugment [25] is applied to the compared methods. We train BigGAN on only the labeled samples. In other words, the method is trained on clean datasets. The other methods and OSSGAN are trained on our constructed open-set dataset, as mentioned above. RejectGAN and OpensetGAN are introduced in the above section. Random-GAN does not have a classifier for the unlabeled images and

assigns labels $y \in \mathcal{Y}$ chosen uniformly at random to unlabeled samples. RandomGAN is very simple, but it learns reasonably well when occasionally assigning correct labels by happenstance. SingleGAN is a naive version of OSS-GAN and assigns the high entropy labels to all unlabeled samples regardless of their content.

For OSSGAN, S^3GAN, RejectGAN, and Openset-GAN, the weighting parameter λ is selected from $\{0.1, 0.2, 0.4, 0.6\}$, respectively. The cross-entropy loss can be large, so care should be taken to set it such that the training balance between the generator and the discriminator is not disturbed. For RejectGAN and OpensetGAN, the threshold c is selected from $\{0.1, 0.3, 0.5, 0.7, 0.9, 0.95\}$.

Evaluation metrics. We employ Inception Score (IS) [17], Fréchet Inception Distance (FID) [5], $F_{1/8}$ score [16], and F_8 score [16] to measure the whole quality of generated samples. FID measures both image quality and diversity with the feature distance between the generated and reference images, but it was not possible to separate the evaluated values into fidelity and diversity. In contrast, $F_{1/8}$ and F_8 aim to quantify fidelity and diversity, respectively. We sample 10K generated images for all the metrics and use the evaluation set as the reference distribution for FID.

6. Experiment results

We first evaluate the effects of the entropy regularization term. To this end, we compute the difference in average entropies at the 100k-th iteration. The difference without the regularization term is 0.104, whereas the difference with the regularization term is 0.283. This shows that optimizing a classifier with entropy regularization results in clearly separated open-set and closed-set samples in unlabeled samples. Consequently, we can expect the regularization to improve the cGAN model performance even when the model is trained on a dataset contaminated with open-set samples.

#known class	#labeled sample/class	#unlabeled sample		OSSGAN		S^3GAN [10]		RandomGAN		SingleGAN		BigGAN [25]	
		closed-set	open-set	FID↓	IS↑	FID↓	IS↑	FID↓	IS↑	FID↓	IS↑	FID↓	IS↑
150	475	3750	25000	25.97	15.30	20.95	16.51	26.51	12.80	24.38	13.82	**17.82**	**17.65**
	250	37500	25000	25.06	13.81	24.05	14.38	33.56	10.86	30.11	11.89	**18.57**	**16.99**
	100	60000	25000	**26.34**	**13.66**	27.25	13.12	34.13	10.82	39.67	10.81	62.77	9.74
	50	67500	25000	**35.39**	**11.61**	54.32	9.36	45.23	10.78	64.25	9.68	109.50	5.94
100	475	2500	50000	31.75	13.81	28.66	14.80	36.70	11.77	**18.66**	**17.67**	18.88	17.60
	250	25000	50000	28.19	13.88	29.19	14.05	51.33	10.02	25.97	16.66	**22.85**	**17.02**
	100	40000	50000	**31.28**	**13.29**	33.18	12.60	56.50	9.94	37.79	11.46	56.36	9.82
	50	45000	50000	**40.61**	**11.54**	230.84	2.50	42.05	10.58	43.45	10.51	128.33	4.59
50	475	1250	75000	56.33	13.63	55.12	13.22	68.67	10.25	71.19	9.72	**23.81**	**13.85**
	250	12500	75000	**56.49**	**13.48**	57.16	12.50	67.98	9.88	74.66	9.36	78.39	6.47
	100	20000	75000	**58.36**	**12.41**	75.94	8.76	72.97	9.82	77.17	9.14	99.78	5.14
	50	22500	75000	**61.60**	**11.84**	95.01	8.29	79.75	8.53	73.41	9.08	161.65	4.17

Table 1. FID and Inception Scores obtained in experiments of different configurations on Tiny ImageNet. Each row shows the configuration, statistics, and results of compared methods in an experiment.

References

BigGAN [1]

RandomGAN

S^3GAN [10]

OSSGAN

Figure 6. Visual comparison of class-conditional image synthesis results on ImageNet. Our method produces plausible images while respecting the given condition.
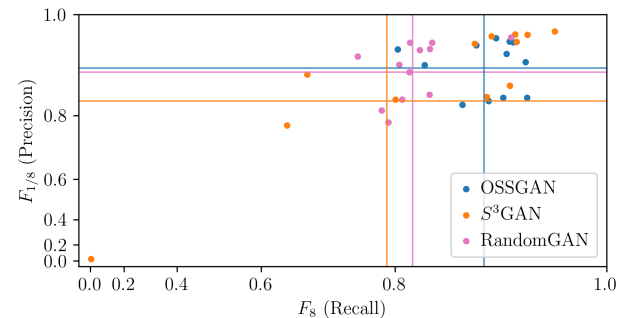

Figure 7. Scatter of precision and recall scores obtained in Tiny ImageNet experiments. Dots show the recall and precision in experiments. Horizontal line shows the average precision of a method. Vertical line shows the average recall of a method. Dots in the upper right indicate high fidelity and high diversity. Conversely, dots in the lower right indicate low fidelity and low diversity.

	FID↓	IS↑	$F_{1/8}$↑	F_8↑
BigGAN [25]	190.88	3.97	0.1178	0.0570
RandomGAN	105.71	12.41	0.6104	0.7679
S^3GAN [10]	180.30	4.38	0.2053	0.1472
OSSGAN	**78.43**	**18.42**	**0.8379**	**0.8359**

Table 2. Experimental results for our ImageNet dataset. Our method outperforms the baseline methods in terms of all the quantitative metrics.

We then investigate whether incorporating the threshold has a negative impact on cGAN training. The FID scores in baseline methods with different thresholds and in OSSGAN are shown in Fig. 5. We see that the quantitative performances of OpensetGAN and RejectGAN are heavily dependent on threshold selection. Their results are not on par with our results, except in the best-case scenario. This is because the average entropy of the open-set samples varies greatly during training, ranging from 0.4 to 0.9. We obtain the average entropies of open-set and closed-set samples, only for benchmarking purposes, not in practice, because we cannot

	Tiny ImageNet	ImageNet
Ours w/o H and Fake	60.09	117.78
Ours w/o H	70.55	85.65
Ours w/o Fake	60.06	81.53
Ours (OSSGAN)	**35.39**	**78.43**

Table 3. Results of ablation study. We report the FID scores of each method. In the ablation parts, H and Fake indicate the entropy regularization term and the cross-entropy loss for fake samples, respectively. The last row is for our OSSGAN model.

divide unlabeled samples into open-set and closed-set samples. Because we cannot set a threshold with respect to the average entropy, the method free of threshold adjustments is more useful for our purposes.

We next compare OSSGAN to S^3GAN, RandomGAN, SingleGAN, and BigGAN on 12 data configurations of Tiny ImageNet, as decribed in Sec. 5. Table 1 shows the quantitative results separated into three segments. Each segment has the same number of closed-set classes while it has different ratios of labeled samples in the closed-set classes. The total number of samples (labeled and unlabeled samples) in each experiment is 100,000. We note that the lower the row, the more difficult the experiment.

We summarize the experimental results of our Tiny ImageNet datasets. For the dataset with sufficient labeled samples for each class (*e.g.*, rows 1, 2, and 5), it is not surprising that BigGAN performs best thanks to the use of data-efficient DiffAugment module. In contrast, when the data difficulty increases, its performance degrades drastically. RandomGAN and SingleGAN perform better than BigGAN for datasets with insufficient number of labeled samples for each class. While S^3GAN provides further improvements over the methods that do not rely on the auxiliary classifier for such datasets, it worsens in extreme cases. Our method outperforms the baselines for the dataset with limited number of labeled samples even in extreme cases. The performance gains are a result of smoothly utilizing the likelihoods quantified by the classifier. OSSGAN with 20% labeled samples achieves comparable performance to Big-GAN (see row 10 and row 12 of Tab. 1).

We also conduct experiments with high-resolution images for a subset of ImageNet. In this experiment, we use labeled data from only 1% of the original ImageNet. Table 2 shows the $F_{1/8}$, F_8, FID, and IS scores for the dataset mentioned in Sec. 5. BigGAN and S^3GAN completely fail for this dataset. The low $F_{1/8}$ and high F_8 scores indicate that RandomGAN probably fails to generate images of a class that matches the given class condition. Our method achieves the best quantitative performance in all metrics. Figure 6 also shows the examples generated by the methods. The qualitative results are consistent with the quantitative results.

The $F_{1/8}$ and F_8 scores of compared methods in Tiny ImageNet experiments with several configurations and the average $F_{1/8}$ and F_8 scores of each method are shown in Fig. 7. The proposed method has a higher recall, that is, diversity, than other methods. RandomGAN, which assigns all classes to each image, fails to learn class-specific features, resulting in low diversity. In contrast, our method learns class-specific and class-invariant features by utilizing confidence, leading to achieve high diversity.

We perform an ablation study to evaluate the contribution of each loss term of our model. For ablation models, we drop either or both the entropy regularization term and the cross-entropy loss for fake samples. The components are indicated by H and Fake, respectively. As Tab. 3 shows, both components contribute to the performance individually, and the combination of the components yields significant improvement. Applying only Fake sometimes harms the performance in the case of a sufficient number of labeled samples for training the classifier.

7. Conclusion

We introduced a novel task of practical image generation. We proposed open-set semi-supervised image generation, a problem that takes into account the properties of the data available when developing applications for image generation models in the real world. Furthermore, to address the proposed task, we designed OSSGAN by integrating the unlabeled samples with the confidence score obtained by the auxiliary classifier into the training of cGANs. Thanks to the utilization of entropy regularization, OSSGAN promoted the discriminator to learn the class-invariant features and to avoid missing the useful features of closed-set classes. The results of our comprehensive experiments on several configurations showed that OSSGAN outperformed other baseline methods and performed well with limited labeled samples. Therefore, our OSSGAN will reduce the cost of building datasets for training cGANs, leading to the expansion of the range of real-world applications of cGANs. The limitation of the proposed method is that the performance improvement by OSSGAN depends on the success of the training of the classifier. In combination with unsupervised and self-supervised learning, we expect to be able to achieve learning of appropriate classifiers from fewer labels than in our method.

References

[1] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale GAN training for high fidelity natural image synthesis. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2018. 1, 2, 3, 4, 5, 6, 7

[2] Ting Chen, Xiaohua Zhai, Marvin Ritter, Mario Lucic, and Neil Houlsby. Self-supervised GANs via auxiliary rotation loss. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12146–12155, 2019. 2

[3] Zihang Dai, Zhilin Yang, Fan Yang, William W Cohen, and Russ R Salakhutdinov. Good semi-supervised learning that requires a bad GAN. In *Advances in Neural Information Processing Systems*, volume 30, 2017. 2

[4] Zhijie Deng, Hao Zhang, Xiaodan Liang, Luona Yang, Shizhen Xu, Jun Zhu, and Eric P Xing. Structured generative adversarial networks. In *Advances in Neural Information Processing Systems*, volume 30, 2017. 1, 2

[5] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in Neural Information Processing Systems*, pages 6626–6637, 2017. 6

[6] Tobias Hinz, Matthew Fisher, Oliver Wang, and Stefan Wermter. Improved techniques for training single-image GANs. In *Proceedings of the IEEE Workshop on Applications of Computer Vision (WACV)*, pages 1300–1309, 2021. 2

[7] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1125–1134, 2017. 2

[8] Tero Karras, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Training generative adversarial networks with limited data. In *Advances in Neural Information Processing Systems*, 2020. 2

[9] Chongxuan Li, Kun Xu, Jun Zhu, and Bo Zhang. Triple generative adversarial nets. In *Advances in Neural Information Processing Systems*, pages 4091–4101, 2017. 1, 2

[10] Mario Lučić, Michael Tschannen, Marvin Ritter, Xiaohua Zhai, Olivier Bachem, and Sylvain Gelly. High-fidelity image generation with fewer labels. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the International Conference on Machine Learning (ICML)*, volume 97, pages 4183–4192, Long Beach, California, USA, 09–15 Jun 2019. PMLR. 1, 2, 3, 6, 7

[11] Huixiang Luo, Hao Cheng, Yuting Gao, Ke Li, Mengdan Zhang, Fanxu Meng, Xiaowei Guo, Feiyue Huang, and Xing Sun. On the consistency training for open-set semi-supervised learning. *arXiv preprint arXiv:2101.08237*, 2021. 2

[12] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014. 1, 2

[13] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2018. 2, 3, 5

[14] Augustus Odena, Christopher Olah, and Jonathon Shlens. Conditional image synthesis with auxiliary classifier GANs. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 2642–2651. PMLR, 2017. 3

[15] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet large scale visual recognition challenge. 6

[16] Mehdi SM Sajjadi, Olivier Bachem, Mario Lucic, Olivier Bousquet, and Sylvain Gelly. Assessing generative models via precision and recall. In *Advances in Neural Information Processing Systems*, pages 5234–5243, 2018. 6

[17] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training GANs. *Advances in Neural Information Processing Systems*, 29:2234–2242, 2016. 2, 6

[18] Tamar Rott Shaham, Tali Dekel, and Tomer Michaeli. SinGAN: Learning a generative model from a single natural image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4570–4580, 2019. 2

[19] Jost Tobias Springenberg. Unsupervised and semi-supervised learning with categorical generative adversarial networks. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2016. 1, 2

[20] Kumar Sricharan, Raja Bala, Matthew Shreve, Hui Ding, Kumar Saketh, and Jin Sun. Semi-supervised conditional GANs. *arXiv preprint arXiv:1708.05789*, 2017. 1, 2

[21] Jiayu Wu, Qixiang Zhang, and Guoxi Xu. Tiny ImageNet challenge. 6

[22] Qing Yu, Daiki Ikami, Go Irie, and Kiyoharu Aizawa. Multi-task curriculum framework for open-set semi-supervised learning. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 438–454. Springer, 2020. 2

[23] Han Zhang, Ian Goodfellow, Dimitris Metaxas, and Augustus Odena. Self-attention generative adversarial networks. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 7354–7363, 2019. 1, 2, 3, 5

[24] Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitris N Metaxas. StackGAN: Text to photo-realistic image synthesis with stacked generative adversarial networks. In *Proceedings of the International Conference on Computer Vision (ICCV)*, pages 5907–5915, 2017. 2

[25] Shengyu Zhao, Zhijian Liu, Ji Lin, Jun-Yan Zhu, and Song Han. Differentiable augmentation for data-efficient GAN training. In *Advances in Neural Information Processing Systems*, 2020. 2, 5, 6, 7