# Effective Data Augmentation with Multi-Domain Learning GANs

**Shin'ya Yamaguchi,**[1] **Sekitoshi Kanai,**[1,2] **Takeharu Eda**[1]

[1]NTT Software Innovation Center

[2]Keio University

Tokyo, Japan

{shinya.yamaguchi.mw, sekitoshi.kanai.fu, takeharu.eda.bx}@hco.ntt.co.jp

## Abstract

For deep learning applications, the massive data development (e.g., collecting, labeling), which is an essential process in building practical applications, still incurs seriously high costs. In this work, we propose an effective data augmentation method based on generative adversarial networks (GANs), called Domain Fusion. Our key idea is to import the knowledge contained in an outer dataset to a target model by using a multi-domain learning GAN. The multi-domain learning GAN simultaneously learns the outer and target dataset and generates new samples for the target tasks. The simultaneous learning process makes GANs generate the target samples with high fidelity and variety. As a result, we can obtain accurate models for the target tasks by using these generated samples even if we only have an extremely low volume target dataset. We experimentally evaluate the advantages of Domain Fusion in image classification tasks on 3 target datasets: CIFAR-100, FGVC-Aircraft, and Indoor Scene Recognition. When trained on each target dataset reduced the samples to 5,000 images, Domain Fusion achieves better classification accuracy than the data augmentation using fine-tuned GANs. Furthermore, we show that Domain Fusion improves the quality of generated samples, and the improvements can contribute to higher accuracy.

## Introduction

Deep learning models have demonstrated state-of-the-art performance in various tasks using high dimensional data such as computer vision (Real et al. 2019), speech recognition (Zeyer et al. 2018), and natural language processing (Vaswani et al. 2017). These models achieve the high performance by optimizing their millions of parameters through the training on labeled data. Since the models can easily overfit the small data due to the enormous parameters, the generalization performance tends to be in proportion to the size of labeled data. In fact, Sun et al. (2017) experimentally showed that the test performance on vision tasks could be improved logarithmically with the labeled data size. To obtain higher performance of deep models, we must develop as many labeled data as possible by collecting data and attaching labels. However, developing the labeled data becomes

one of the main obstacles in the entire deployment of deep models since it requires a lot of time and high costs.

One of the most common techniques to alleviate the costs of labeled data developments is *data augmentation (DA)*. To improve the performance of the target task (e.g., classification or regression), DA amplifies the variation of existing labeled data (target data) by adding small transformations (e.g., random expansion, flip, and rotation). Since DA improves the performance despite its simplicity and has no dependency on network architectures, it is widely applied to many applications (Krizhevsky, Sutskever, and Hinton 2012; Ko et al. 2015). However, when we train target models on low-volume datasets, the improvements by DA is limited because DA is designed to transform an existing sample into a slightly modified sample. In other words, DA does not generate truly *unseen* data, which have information not included in the data to be transformed. For example, in image recognition, DA is not able to transform running-horse images into sitting-horse images. Therefore, the benefit of DA is limited when we only have low-volume datasets.

Several methods (Tran et al. 2017; Zheng, Zheng, and Yang 2017; Calimeri et al. 2017; Zhu et al. 2018; Antoniou, Storkey, and Edwards 2018) have been presented to overcome the limitation of DA by applying generative adversarial networks (GANs, Goodfellow et al. (2014)). GANs generate various and realistic data samples by learning data distributions; they can generate unseen samples from the learned distributions. The existing methods employ this ability and use the generated samples as additional input for the target task. Although these GAN-based methods succeed at improving the target performance, they assume that there is a sufficient volume of data for training GANs. In fact, in the case of low volume data, the generated samples have less fidelity and variety and can degrade the target performance (Wang et al. 2018; Shmelkov, Schmid, and Alahari 2018). This is because low volume data has insufficient knowledge, and thus, we need to utilize supplementary knowledge for training GANs. To train GANs with low-volume target data, Wang et al. (2018) proposed Transferring GANs (TGANs) which incorporate a fine-tuning technique into GANs. However, Wang et al. experimentally show TGANs do not improve the generating performance very well when we have

only 1 K target dataset.

In this paper, we propose *Domain Fusion* (DF), which is an effective data augmentation technique exploiting GANs trained on a target and another dataset. To generate helpful samples, DF incorporates knowledge from the *outer domain*, which is another domain from the target, into a GAN. Specifically, we train GANs on target and outer datasets, simultaneously unlike TGAN. After training GANs, we use the generated samples in the target domain for the target tasks. In order to generate the target samples explicitly, we adopt conditional GANs that can produce the conditioned samples by assigning class labels. As a result, DF transfers the helpful knowledge of the outer domain into generated target samples via the shared parameters of GANs. We call this training method *multi-domain training*, and the trained GANs *multi-domain learning GANs*.

Furthermore, to enhance the quality of the generated samples, we propose two improvement techniques for DF. First, we introduce a metric to select an outer dataset that includes knowledge to generate more helpful target samples. An appropriate outer dataset needs to be selected for the target domain since the performance of DF depends on the choice. To this end, we develop a new metric based on Fréchet inception distance (FID, Heusel et al. (2017)) and multiscale structural similarity (MS-SSIM, Wang, Simoncelli, and Bovik (2003)) that focuses on the relevance between the target and outer domain, and the diversity of the outer samples. Second, when generating samples from a GAN, we apply filtering to remove extremely broken samples that could lead to negative effects on target models. For this purpose, we use *discriminator rejection sampling* (DRS,Azadi et al. (2019)), which uses the information from a discriminator of a GAN to omit the bad samples. We extend the DRS algorithm for conditional GANs to generate high-quality class-conditional samples. Applying these improvements, we can generate more helpful target samples.

Our experimental results demonstrate that the samples from our GANs in DF more improve the accuracy in a low data regime compared to TGANs. Furthermore, we show that our GANs can produce higher quality samples than TGANs in terms of FID and Inception Score. We also experimentally confirm the correlation between the quality of generated samples and the classification accuracy. More importantly, we show that the classifiers trained by a combination of DF and conventional DA outperform the ones trained by only using conventional DA.

Our main contributions are as follows:

- We propose a new data augmentation method using GANs called Domain Fusion, which transfers knowledge of the outer dataset into the target models by using a GAN trained on multi-domain via the shared parameters. We also propose a metric for outer dataset selection, and modified DRS for filtering generated samples.

- We confirm that the correlations between the quality of generated samples and the target-task performances in our experiments on CIFAR-100, FGVC-Aircraft and Indoor Scene Recognition in low-volume data regime. These results support that Domain Fusion improve the target mod-

els because of the high quality generated samples.

## Background

### Generative Adversarial Networks

A generative adversarial network (GAN) is composed of a *generator* network $G_\theta(z)$, and a *discriminator* network $D_\phi(x)$ (Goodfellow et al. 2014). The $G$ generates fake samples from random noise $z \sim p_z$ and the $D$ has a role to distinguish an observation $x$ whether $x$ comes from generator $G(z)$ or data distribution $p_{\text{data}}$. The objective functions for training a discriminator and a generator are respectively formalized as follows:

$$
\begin{aligned}
L_D &= -\mathbb{E}_{x \sim p_{\text{data}}} \log D_\phi(x) \\
&\quad -\mathbb{E}_{z \sim p_z} \log\left(1 - D_\phi(G_\theta(z))\right), \quad (1) \\
L_G &= -\mathbb{E}_{z \sim p_z} \log D_\phi(G_\theta(z)). \quad (2)
\end{aligned}
$$

Through a tandem training of $G$ and $D$, $D$ learns to maximize the probability of assigning the "real" label into real examples, whereas $G$ learns to maximize the probability of failing the distinction by the $D$. When $G$ and $D$ converge to equilibrium point, the generator network $G$ produces realistic samples as good representation of data distribution $p_{\text{data}}$.

In Domain Fusion, we use conditional GANs (cGANs) (Odena, Olah, and Shlens 2017; Miyato and Koyama 2018) that generate samples conditioned by class labels. The objective functions are given by rewriting Eq. (1) and (2):

$$
\begin{aligned}
L_D &= -\mathbb{E}_{x \sim p_{\text{data}}} \log D_\phi(x, y) \\
&\quad -\mathbb{E}_{z \sim p_z} \log\left(1 - D_\phi(G_\theta(z, y), y)\right), \quad (3) \\
L_G &= -\mathbb{E}_{z \sim p_z} \log D_\phi(G_\theta(z, y), y). \quad (4)
\end{aligned}
$$

While there are several formulations for cGANs, we adopt a projection based conditioning (Miyato and Koyama 2018) as our implementation of cGAN. This approach concatenates the embedded conditional vector to the feature vector of the generator and discriminator to learn the condition.

### Data Augmentation with GANs

There are several studies applying GANs into data augmentation schemes. Calimeri et al. (2017) have proposed an approach simply applying generated samples as additional datasets for medical imaging tasks. Zhu et al. (2018) have shown an application using conditional GANs for augmenting plant images. For re-identification tasks in computer vision, the study of (Zheng, Zheng, and Yang 2017) has presented a training method with unconditional generated samples. Tran et al. (2017) have presented a way to train classification models with GANs in semi-supervised fashion. Similarly to our work, these studies leveraged generated samples from GANs as supplementary training data for target models. This is an intuitive and flexible strategy because we can easily use the generated samples as augment dataset like conventional DA. However, in low volume data, these types of data augmentation suffers from the problem of insufficient training a GAN as described in the next section. In fact, Shmelkov, Schmid, and Alahari (2018) have shown that the generated samples from low-data trained GANs degrade the

accuracy of classifiers. Our approach can help these existing GAN-based methods to reduce the negative effects of this problem since it improves the quality of the generated samples in the case of low volume data.

## Training GANs with Low Data Volume

In a low volume training data regime, Wang et al. (2018) have shown a fine-tuning technique for training of GANs, called *Transferring GANs*. The authors tried to initialize weights of a GAN by leveraging pretrained generators and discriminators with greater volume outer datasets such as ImageNet. They investigated the effect of the target data size by the experiments where GANs were pretrained on the outer dataset (ImageNet) and then fine-tuned to the target dataset (LSUN Bedrooms). Their results showed that fine-tuned GANs generate high-quality samples in the case of large target data (18.5 of FID by 1M samples), but relatively low-quality samples in the case of less volume target data (93.4 of FID by 1K samples). Since 1K of target samples still requires us much effort for developing dataset, training of GANs with low data volume is still challenging.

## Domain Fusion

In this section, we present Domain Fusion using multi-domain learning GANs. A multi-domain learning GAN is trained on the target dataset and outer dataset simultaneously. The procedure of Domain Fusion consists of the following three steps; (a) selecting an outer dataset, (b) multi-domain training a GAN, (c) sampling target labeled samples from the trained GAN. In the rest of this section, we describe each of the steps.

### Selecting Outer Dataset

First, we select an outer dataset that has useful knowledge for the target domain. In this paper, we denote a dataset $S$ composed of $X$ and $Y$, where $X$ is a set of data samples (e.g., images) and $Y$ is a set of labels. If we have a target dataset $S_T$, the outer dataset $S_O$ is selected from the candidates $\{S_i\}$ according to $\mathcal{M}(S_T, S_i)$ which is our outer dataset metric of $S_i$ for $S_T$:

$$S_O = \{(x, y) \mid x \in X_i, y \in Y_i, (X_i, Y_i) \in S_i, \\ i = \arg\min_i \mathcal{M}(S_T, S_i)\}. \quad (5)$$

In fact, it is non-trivial what metrics we should choose for outer dataset selection. We propose a metric that makes account both the relevance between the target and outer dataset, and the diversity of outer samples (see Improvements Section).

### Multi-Domain Training

Next, we train a conditional GAN; discriminator $D(x, y)$ to minimize Eq. (3) and generator $G(z, y)$ to minimize Eq. (4) on both $S_T$ and $S_O$. The objective functions of the multi-domain training are defined as follows:

$$L_D = \alpha L_{D_T} + (1 - \alpha) L_{D_O}, \quad (6)$$
$$L_G = \alpha L_{G_T} + (1 - \alpha) L_{G_O}, \quad (7)$$

---

**Algorithm 1** Multi-Domain Training of Domain Fusion

**Input:** Set of target data $X_T$, set of outer data $X_O$, set of target labels $Y_T$, set of outer labels $Y_O$, batchsize $B$, learning rate $\eta_\theta, \eta_\phi$, scaling factor $\alpha$
**Output:** Trained Generator $G_\theta$
1: Randomly initialize parameters $\theta, \phi$
2: **while** not convergence **do**
3:     **for** $k$ steps **do**
4:         $\{x_T^i\}_{i=1}^B, \{y_T^i\}_{i=1}^B \leftarrow \texttt{GetSample}(X_T, Y_T, B)$
5:         $\{z_T^i\}_{i=1}^B \leftarrow \texttt{GenNoise}(B)$
6:         $L_{D_T} \leftarrow -\sum_i^B \log D_\phi(x_T^i, y_T^i)$    ▷ Eq.(8)
7:             $-\sum_i^B \log(1 - D_\phi(G_\theta(z_T^i, y_T^i), y_T^i))$
8:         $\{x_O^i\}_{i=1}^B, \{y_O^i\}_{i=1}^B \leftarrow \texttt{GetSample}(X_O, Y_O, B)$
9:         $\{z_O^i\}_{i=1}^B \leftarrow \texttt{GenNoise}(B)$
10:       $L_{D_O} \leftarrow -\sum_i^B \log D_\phi(x_O^i, y_O^i)$   ▷ Eq.(9)
11:          $-\sum_i^B \log(1 - D_\phi(G_\theta(z_O^i, y_O^i), y_O^i))$
12:       $\phi \leftarrow \phi - \eta_\phi \nabla_\phi(\alpha L_{D_T} + (1 - \alpha) L_{D_O})$ ▷ Eq.(6)
13:     **end for**
14:     $\{y_T^i\}_{i=1}^B \leftarrow \texttt{GetLabel}(Y_T, B)$
15:     $L_{G_T} \leftarrow -\sum_i^B \log D_\phi(G_\theta(z_T^i, y_T^i), y_T^i))$ ▷ Eq.(10)
16:     $\{y_O^i\}_{i=1}^B \leftarrow \texttt{GetLabel}(Y_O, B)$
17:     $L_{G_O} \leftarrow -\sum_i^B \log D_\phi(G_\theta(z_O^i, y_O^i), y_O^i))$ ▷ Eq.(11)
18:     $\theta \leftarrow \theta - \eta_\theta \nabla_\theta(\alpha L_{G_T} + (1 - \alpha) L_{G_O})$   ▷ Eq.(7)
19: **end while**

---

where,

$$\begin{aligned} L_{D_T} &= -\mathbb{E}_{x_T \sim p_{\text{target}}} \log D_\phi(x_T, y_T) \\ &\quad -\mathbb{E}_{z \sim p_z} \log(1 - D_\phi(G_\theta(z, y_T), y_T)), \quad (8) \\ L_{D_O} &= -\mathbb{E}_{x_O \sim p_{\text{outer}}} \log D_\phi(x_O, y_O) \\ &\quad -\mathbb{E}_{z \sim p_z} \log(1 - D_\phi(G_\theta(z, y_O), y_O)), \quad (9) \\ L_{G_T} &= -\mathbb{E}_{z \sim p_z} \log D_\phi(G_\theta(z, y_T), y_T), \quad (10) \\ L_{G_O} &= -\mathbb{E}_{z \sim p_z} \log D_\phi(G_\theta(z, y_O), y_O), \quad (11) \end{aligned}$$

and $0 \le \alpha \le 1$ is a hyperparameter balancing the learning scale between the target and outer dataset ($\alpha = 0.5$ in default setting). In each step of the optimization, we sample data from the both target and outer dataset, and then compute the objective functions. For both the target and outer domain, we adopt conditional GANs (CGANs) because the labels allow GANs to generate the target samples explicitly. Furthermore, GANs with labels can achieve a higher generation performance than one without the labels (Lučić et al. 2019). We assume that $Y_T$ and $Y_O$ are disjoint each other. In the training, we can summarize $Y_O$ into one class since the target tasks do not use labels of the outer dataset. However, we experimentally found that class-wise training with $Y_O$ as well as $Y_T$ contributes to the higher quality of generated samples. We infer that this is because $Y_O$ makes the learning of the outer domain be easier, and such learned representations help to generate target samples. The overall procedure of the multi-domain training is illustrated in Algorithm 1.

## Sampling Target Examples

After training, we generate a set of new data samples $X_{\text{gen}}$ from the trained generator $G(z,y)$ as follows:

$$X_{\text{gen}} = \{x \,|\, x = G(z,y), z \sim p_z, y \in Y_{\text{T}}\}. \qquad (12)$$

Note that the input label $y$ is an element of $Y_{\text{T}}$ since the purpose of a Domain Fusion is to augment the target dataset $S_{\text{T}}$. We generate equal amount of samples for each label.

In general, trained conditional GANs generate samples by only using a generator $G$. However, the generated samples can include poor quality samples that have been rejected by the discriminator at the training. To obtain more high-quality samples, we apply discriminator rejection sampling (DRS, Azadi et al. (2019)). In the next section, we show our modified DRS algorithm for conditional sampling.

Finally, the generated $X_{\text{gen}}$ is integrated into the target dataset $S_{\text{T}}$.

$$
\begin{aligned}
S_{\text{aug}} &= \{(x,y) \,|\, x \in X_{\text{aug}}, y \in Y_{\text{aug}}\}, &(13)\\
X_{\text{aug}} &= X_{\text{T}} \cup X_{\text{gen}}, &(14)\\
Y_{\text{aug}} &= Y_{\text{T}} &(15)
\end{aligned}
$$

We assume that generated data $X_{\text{gen}}$ derived from the generator $G(z, y \in Y_{\text{T}})$ have attribute consistency of the specified labels $y \in Y_{\text{T}}$. Thus, the augmented dataset $S_{\text{aug}}$ is directly used as the input for the target model training as the alternative of the target dataset $S_{\text{T}}$.

# Improvements

## Outer Dataset Selection Metric

In Domain Fusion, the choice of an outer dataset for the target is a dominant factor determining both the target model performance and the quality of generated samples. In order to select a proper outer dataset, we focus on the *relevance* between the target and outer dataset, and the *diversity* of an outer dataset.

**Relevance Between the Target and Outer Dataset**  In the context of transfer learning, measuring the relevance between outer and target domain is widely used to avoid *negative transfer*, i.e., the target models could perform worse than the case of non transferring. For GANs, Wang et al. (2018) attempt to select the outer dataset by measuring Fréchet inception distance (FID, Heusel et al. (2017)) to the target dataset. An FID between two datasets $X_i$ and $X_j$ is computed on features of ImageNet pretrained Inception Net:

$$\text{FID}(X_i, X_j) = \|\mu_i - \mu_j\|_2^2 + \text{Tr}(\Sigma_i + \Sigma_j - 2(\Sigma_i \Sigma_j)^{\frac{1}{2}}), \tag{16}$$

where $\mu_i$ and $\Sigma_i$ are the mean and covariance of the feature vectors of Inception Net for input $X_i$. A lower FID means that $X_i$ and $X_j$ are highly related to each other. Following Wang et al., we adopt FID as part of our metrics to measure the relevance of the target and outer dataset. In our use, FID is a more preferable than other relevance metrics (e.g., general Wasserstein distance and maximum mean discrepancy) because there is no need to train additional feature extractors or kernel functions for each pair of datasets.

**Diversity of an Outer Dataset**  In (Wang et al. 2018), they also reported the limitation of FID to predict actual quality of the generated samples from fine-tuned GANs. This indicates that even if the outer dataset is highly relevant to the target, the outer dataset does not necessarily improve the quality of the generated target samples. Thus, only using FID is insufficient for proper outer dataset selection.

In Domain Fusion, we propose a metric with an additional perspective of diversity to select an outer dataset. We assume that an outer dataset with diverse samples is preferable for the target sample generation because the more diverse samples can contain more useful and general information for target sample generations. In order to select the dataset containing more diverse samples, we exploit multi-scale structural similarity (MS-SSIM, Wang, Simoncelli, and Bovik (2003)). MS-SSIM is an approach to assess structural similarity in multi-scale, and it is well accepted as an evaluation method for image compression tasks. Recently, MS-SSIM is used for evaluating the diversity of generated samples by GANs Odena, Olah, and Shlens; Miyato and Koyama (2017; 2018). We apply MS-SSIM to assess the diversity of existing datasets for selecting more helpful outer datasets. An MS-SSIM of two data samples $x_i$ and $x_j$ is defined as follows:

$$\text{SSIM}(x_i,x_j) = l_M(x_i,x_j)^{\alpha_M} \prod_{m=1}^{M} c_m(x_i,x_j)^{\beta_j} s_m(x_i,x_j)^{\gamma_m}, \tag{17}$$

where $l = \frac{2\mu_{x_i}\mu_{x_j}+C_1}{\mu_{x_i}^2+\mu_{x_j}^2+C_1}$, $c = \frac{2\sigma_{x_i}\sigma_{x_j}+C_2}{\sigma_{x_i}^2+\sigma_{x_j}^2+C_2}$, $s = \frac{\sigma_{x_ix_j}+C_3}{\sigma_{x_i}\sigma_{x_j}+C_3}$, and $M$ denotes a scale number. $l$ is computed only once at the maximum $M$, and $c, s$ are computed at all scales. $\mu_{x_i}$ and $\sigma_{x_i}$ are the mean and standard deviation of $x_i$. $\sigma_{x_ix_j}$ is the covariance of $x_i$ and $x_j$. $\alpha$, $\beta$, and $\gamma$ represent the hyper-parameters, and $C_1$, $C_2$, and $C_3$ are small constants computed by the dynamic range of the pixel values and scalar constants. The ranges of MS-SSIM is between 0 (high diversity) and 1 (low diversity), and $\text{MS-SSIM}(x_i, x_i) = 1$.

To evaluate the diversity of a dataset, we calculate the mean MS-SSIM for all the combinations of the samples in the dataset.

$$\overline{\text{SSIM}}(X) = \frac{\sum_{x_i \in X}^{x_i \neq x_j} \sum_{x_j \in X}^{x_j \neq x_i} \text{SSIM}(x_i, x_j)}{(|X|^2 - |X|)}, \tag{18}$$

where $|X|$ denotes the size of $X$. We consider that the mean MS-SSIM indicates the diversity of the dataset.

**Outer Dataset Metric $\mathcal{M}$**  By combining FID and mean MS-SSIM, we compute an outer dataset metric $\mathcal{M}$ for a target dataset $X_{\text{T}}$ and an outer dataset $X_{\text{O}}$ as follows:

$$\mathcal{M}(X_{\text{T}}, X_{\text{O}}) = \text{FID}(X_{\text{T}}, X_{\text{O}}) \cdot \overline{\text{SSIM}}(X_{\text{O}}) \tag{19}$$

A lower $\mathcal{M}$ indicates a more proper outer dataset. We aim to select an outer dataset with both high relevance to a target dataset and high diversity within the samples. This metric helps to pick such outer datasets according to the multiplication of FID and MS-SSIM representing the relevance and diversity, respectively. The role of MS-SSIM (diversity), which is in $[0, 1]$, is to weight FID (relevance), which is in $[0, +\infty]$. In Experimental Results Section, we show that FID and MS-SSIM complementarily contribute to choosing an appropriate outer dataset in practice.

Table 1: List of outer datasets. Each dataset size is the total number of the train and test size expect for Pascal-VOC.

| Dataset | Classes | Size |
|---|---|---|
| Oxford 102 Flowers (Nilsback and Zisserman 2008) | 102 | 8,189 |
| Stanford Cars (Krause et al. 2013) | 196 | 16,185 |
| Food-101 (Bossard, Guillaumin, and Van Gool 2014) | 101 | 101,000 |
| Describable Textures (DTD) (Cimpoi et al. 2014) | 47 | 5,640 |
| LFW (Huang et al. 2007) | 1 | 13,000 |
| SVHN (Netzer et al. 2011) | 10 | 99,289 |
| Pascal-VOC 2012 Cls. (Everingham et al. 2015) | 20 | 5,717 |

## Filtering by Modified DRS

In general, after training of GANs, we obtain the generated samples from GANs by only using the generator. This is because we implicitly assume a successfully trained generator can always generate the samples fooling the discriminator with a probability of 1/2 (Goodfellow et al. 2014). However, since this assumption does not hold in real world, the generator can produce broken samples that are easily detected by the discriminator as fake. For data augmentation, we must avoid such broken samples.

In order to filter out broken samples, we adopt discriminator rejection sampling (DRS, Azadi et al. (2019)) to Domain Fusion. DRS is a rejection sampling method proposed for GANs, which computes an acceptance probability for each sample by using the density ratio from the discriminator. Since DRS cuts off the broken samples according to the acceptance probability, sampling with DRS produces more high-quality samples than one with a generator alone.

Since the original paper of DRS has only shown the algorithm for unconditional sampling, we cannot directly apply the algorithm to Domain Fusion, which requires conditional sampling for the data augmentation. Therefore, we modify the DRS algorithm for conditional sampling. The modification is to compute the density ratio for each class label. In the original DRS, one density ratio is estimated for a GAN without considering classes. This may cause losing the diversity of samples of a specific class, because the sampling difficulty varies according to each class (Brock, Donahue, and Simonyan 2019). By estimating the class-wise density ratio, we aim to coordinate the acceptance probability for each class. Applying this modification, we can obtain class conditional generated samples with high fidelity and variety. (Our modified algorithm is shown in the supplemental materials.)

## Experimental Results

In this section, we show the evaluation of Domain Fusion (DF) on the image classification task using three datasets: CIFAR-100, FGVC-Aircraft, and Indoor Scene Recognition. We compare our proposed DF with the conditional GAN (CGAN) and Transferring GAN (TGAN).

## Settings

**Target Datasets**   The target task was the image classification on CIFAR-100 (Krizhevsky and Hinton 2009), FGVC-Aircraft (Maji et al. 2013), and Indoor Scene Recognition (ISR) (Quattoni and Torralba 2009). We used CIFAR-100

instead of CIFAR-10 because CIFAR-100 can contribute to a more realistic evaluation with a larger number of labels and fewer samples per class. These three datasets are characterized by samples with different features; CIFAR-100 is composed of the classes with various modes (vegetables, cars, furniture, etc.), FGVC-Aircraft includes only one mode (airplane) and has fine-grained classes that slightly differ each other, and ISR is also constructed by one mode (indoor scenes) but has more diverse and rough-grained information than FGVC-Aircraft. To evaluate the performance in low volume data setting, we reduced each training set of CIFAR-100 (50,000 images), FGVC-Aircraft (6,667 images), and ISR (5,360 images) to 5000 images, which are randomly sampled for each class. Note that although the reductions for FGVC-Aircraft and ISR are relatively smaller than one of CIFAR-100, they originally have small absolute dataset volume per class; they are difficult to train the models even if we use full of the datasets. We trained conditional GANs, and then, trained the classification model by using the generated samples as the additional dataset. At the test step, we used the original test images (CIFAR-100: 10,000 images, FGVC-Aircraft: 3,333 images, ISR: 1,340 images) to accurately evaluate the trained models.

**Outer Datasets**   Table 1 describes the list of the candidate for the outer dataset. These are image datasets of various domain that are often used for the evaluation of computer vision tasks. At training of DF and TGAN, we used train and test sets of these outer datasets except for Pascal-VOC. We used only train set of Pascal-VOC for training because Pascal-VOC is employed for the reverse-side evaluation which flips the target and outer dataset each other (The reverse-side evaluation is appeared in the supplemental materials). For fair evaluation of the outer datasets, we randomly sampled 5,000 images from each dataset, and used the samples for training GANs. We coordinated the number of samples to equal among classes. Since these datasets contain various images of resolutions, we resized all of the images into $32 \times 32$ by bilinear interpolation.

**Implementation Details   GANs.** We used ResNet-based SNGAN (Miyato et al. 2018; Miyato and Koyama 2018) for $32 \times 32$ resolution images as the implementation of conditional GANs. The model architecture was the same as (Miyato and Koyama 2018). We trained a GAN for 50k iterations with a batch of 256 using Adam ($\beta_1 = 0$, $\beta_2 = 0.9$) (Kingma and Ba 2014). Following (Heusel et al. 2017), the learning rate of generators and discriminators were $1.0 \times 10^{-4}$ and $4.0 \times 10^{-4}$, respectively. We linearly shifted both the learning rates to 0. Moreover, to fairly evaluate the models for each outer dataset, we incorporated early stopping with Inception Score (IS) (Salimans et al. 2016). The trigger of early stopping was set by estimated IS in each 1,000 iterations for 12,800 generated samples. We stopped training when the consecutive drop count of IS reaches to 5. In multi-domain training, we set $\alpha = 0.5$ for all experiments. In order to use filtering by DRS, we added additional sigmoid layers into the discriminator of the conditional SNGAN, and trained the additional layers for 10,000 steps for each class label (the learning rate was $1.0 \times 10^{-7}$).

Table 2: Performance comparison among data augmentation using GANs (top-1 and top5 classification accuracy (%), FID, and IS). TGAN and DF are the cases of applying Pascal-VOC as an outer dataset that marks the best score of our metric $\mathcal{M}$ for all targets. TGAN-Best denotes the best cases of TGAN approach when using another outer dataset that achieves the best accuracy. AVG represents average scores of 7 outer datasets. Note that, when we used 100% volume of CIFAR-100, FGVC-Aircraft, and ISR (without generated images and any other data augmentations), the classifiers respectively achieved 61.71%, 30.25%, and 27.27% test accuracy under these conditions.

| | CIFAR-100 | | | | FGVC-Aircraft | | | | Indoor Scene Recognition | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Top-1 Acc. | Top-5 Acc. | FID | IS | Top-1 Acc. | Top-5 Acc. | FID | IS | Top-1 Acc. | Top-5 Acc. | FID | IS |
| Without DA | 27.2±0.1 | 54.3±0.5 | – | – | 22.6±2.3 | 48.4±3.4 | – | – | 24.0±2.0 | 52.0±0.7 | – | – |
| CGAN | 26.5±0.4 | 53.6±0.3 | 59.2±0.9 | 4.99±0.07 | 23.6±0.6 | 50.9±0.7 | 110.7±2.8 | 3.41±0.05 | 25.7±0.7 | 52.6±0.7 | 97.9±0.1 | 3.48±0.02 |
| TGAN | 25.9±0.5 | 52.1±0.6 | 60.9±5.9 | 5.20±0.20 | 24.1±0.4 | 51.0±0.7 | 109.0±3.6 | 3.45±0.03 | 24.0±0.2 | 51.0±1.4 | 104.1±5.3 | 3.49±0.07 |
| DF (ours) | **28.9±0.5** | **56.2±0.4** | **53.5±1.7** | **5.32±0.03** | **27.3±0.9** | **55.4±0.4** | **97.9±1.6** | **3.53±0.15** | **26.1±0.7** | **53.8±0.9** | **96.5±4.0** | **3.61±0.08** |
| TGAN-Best | 28.2±0.5 | 55.7±0.2 | 54.5±5.2 | 5.16±0.03 | 26.2±0.3 | 52.9±0.3 | 109.5±2.0 | 3.47±0.03 | 25.7±1.0 | 54.9±1.2 | 97.8±5.8 | 3.50±0.03 |
| TGAN-AVG | 26.7±1.4 | 53.6±1.9 | 60.5±3.5 | 4.98±0.22 | 23.8±3.4 | 49.8±4.6 | 113.4±8.5 | 3.42±0.04 | 23.4±1.5 | 51.0±2.3 | 111.8±12.8 | 3.38±0.18 |
| DF-AVG | 28.1±0.9 | 55.1±1.5 | 56.3±2.5 | 5.24±0.24 | 25.2±1.5 | 52.3±1.8 | 105.8±15.2 | 3.47±0.06 | 24.2±1.2 | 52.4±1.8 | 106.5±13.9 | 3.46±0.25 |

Table 3: Ablation study of Domain Fusion

| | CIFAR-100 | | | | FGVC-Aircraft | | | | Indoor Scene Recognition | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Top-1 Acc. | Top-5 Acc. | FID | IS | Top-1 Acc. | Top-5 Acc. | FID | IS | Top-1 Acc. | Top-5 Acc. | FID | IS |
| CGAN with DRS | 27.3±0.3 | 54.5±1.3 | 58.7±0.8 | 5.05±0.01 | 24.6±0.8 | 52.4±0.9 | 110.0±3.5 | 3.42±0.09 | 24.8±0.9 | 52.8±0.5 | 99.9±6.8 | 3.42±0.06 |
| TGAN with DRS | 26.6±1.5 | 53.5±1.2 | 59.9±5.9 | 5.22±0.03 | 24.4±1.2 | 52.2±0.6 | 107.4±3.0 | 3.49±0.05 | 24.9±0.8 | 53.2±1.4 | 103.9±2.5 | 3.44±0.09 |
| DF w/o $\mathcal{M}$ and DRS (Worst) | 25.5±0.3 | 52.4±0.2 | 60.9±0.1 | 4.75±0.13 | 24.2±0.3 | 50.9±1.8 | 105.2±5.8 | 3.35±0.01 | 24.2±0.3 | 50.9±1.8 | 105.2±5.8 | 3.35±0.01 |
| DF w/o DRS | 28.3±0.7 | 55.7±0.5 | 54.9±2.4 | 5.16±0.04 | 27.0±0.5 | 54.0±0.3 | 98.4±2.6 | 3.50±0.05 | 25.4±0.1 | 53.4±1.7 | 99.0±1.3 | 3.57±0.06 |
| DF | **28.9±0.5** | **56.2±0.4** | **53.5±1.7** | **5.32±0.03** | **27.3±0.9** | **55.4±0.4** | **97.9±1.6** | **3.53±0.15** | **26.1±0.7** | **53.8±0.9** | **96.5±4.0** | **3.61±0.08** |

For TGAN, we trained the conditional GANs on an outer dataset for 50k iterations with the early stopping, and then fine-tuned the pretrained GANs for a target dataset in the same setting.

**Classifiers.** The architecture for the target classifier was ResNet-18 for 224×224 (He et al. 2016) with Adam optimizer for 100 epochs, batches of size 512. We selected the batch size by grid search over 128, 256, 512, 1024 on all three target datasets to maximize the average accuracy across the datasets. The hyperparameters for Adam were $\alpha_{\text{Adam}} = 2.0 \times 10^{-4}$, $\beta_1 = 0$, $\beta_2 = 0.9$. We applied no conventional data augmentation (e.g., flip, rotation) to the input images without noted. We used 50,000 samples (4,000 real images + 46,000 generated images) as training set, and 1,000 real images as validation set. In all cases, we run the test for measuring mean accuracy on each test set of the target datasets.

**Evaluation Metrics** We evaluated DF on the two aspects: the performance of target classification models and the quality of generated samples on target domain. For the classifiers, we assessed the performance by top-1 and top-5 accuracy. The sample quality was measured by Fréchet Inception Distance (FID) (Heusel et al. 2017) and Inception Score (IS) (Salimans et al. 2016). For each target dataset, we computed FID and IS with 128 generated samples per class. FID was calculated between the generated samples and the real images in the 100% volume train set. In all experiments, we trained GANs and classifiers three times, and show the mean and standard deviation of accuracy, FID, and IS.

### Effects of Generated Sample Quality
### Evaluation of Classification Accuracy
**Comparison to Other GAN-based Data Augmentations**
First, we evaluated the efficacy of Domain Fusion (DF) in

Table 4: Performance comparison to conventional DA

| | CIFAR-100 | | FGVC-Aircraft | | Indoor Scene Recognition | |
|---|---|---|---|---|---|---|
| | Top-1 Acc. | Top-5 Acc. | Top-1 Acc. | Top-5 Acc. | Top-1 Acc. | Top-5 Acc. |
| cDA | 30.7±0.7 | 57.3±0.3 | 29.6±0.9 | 58.5±1.6 | 31.0±0.3 | 59.6±0.7 |
| DF+cDA | **32.1±0.7** | **59.2±0.4** | **31.2±0.7** | **60.2±1.0** | **32.4±1.7** | **61.6±1.1** |

terms of the classification accuracy by comparing it to other GAN-based data augmentations. We compared the performance against two patterns of GAN-based data augmentation: generating target samples from (i) CGAN: conditional GANs trained on each target dataset only (Zhu et al. 2018), and (ii) TGAN: conditional Transferring GANs pretrained on an outer dataset (Wang et al. 2018). We also show the performance of classifiers trained on a target dataset without data augmentation (Without DA).

Table 2 lists the results of the top-1 and top-5 accuracy on the classification task, and summarizes the FID and IS of generated samples from GANs. For the results of DF and TGAN, we report the accuracy with the outer dataset which has best our metric score $\mathcal{M}$ (Pascal-VOC). Additionally, for TGAN, we show the best accuracy among 7 outer datasets as TGAN-Best (CIFAR-100 and ISR: Food-101, FGVC-Aircraft: Stanford Cars). We can see that our DF achieves the best classification accuracy among all patterns. As reported in (Shmelkov, Schmid, and Alahari 2018), CGAN dropped the accuracy from Without DA in the cases of CIFAR-100. On the other hand, we see that DF, which transfers outer knowledge to target models, outperforms Without DA. DF also generated the target samples with better FID and IS than CGAN. These results suggest that the quality improvements of the generated samples contribute to the target accuracy. Compared to TGAN, DF helps more accurate classifications and generates better samples. For all of the target datasets, we confirmed the differences between DF and
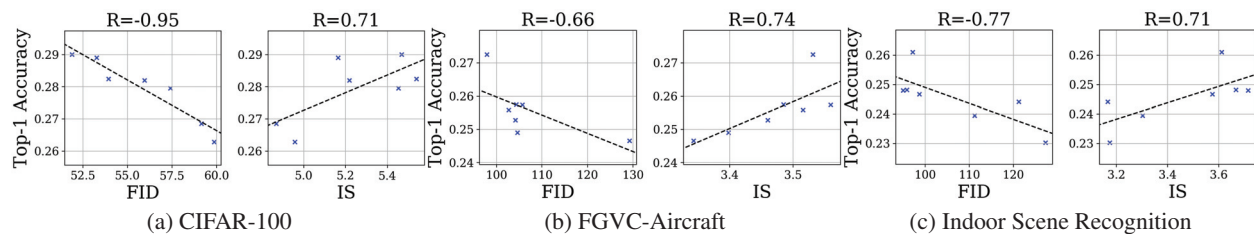
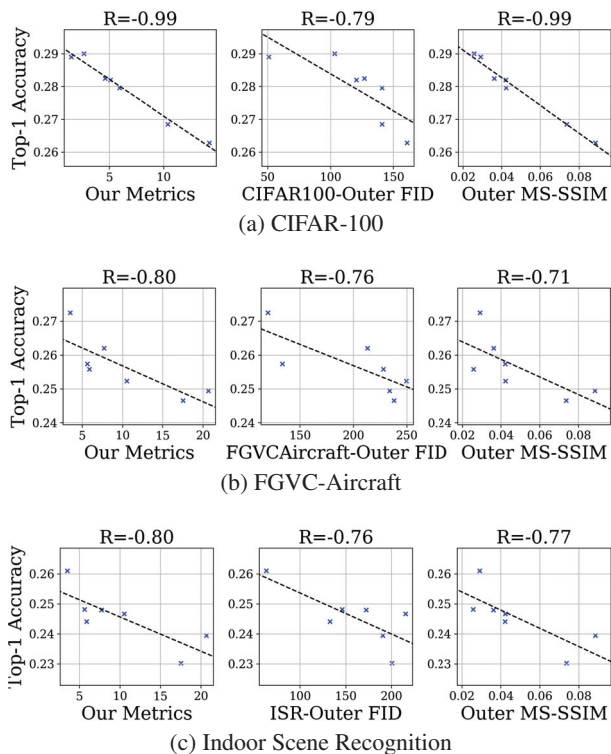Figure 1: Correlation between generated sample quality and top-1 accuracy



Figure 2: Comparison of metrics

denotes the worst cases among outer datasets that use no filtering by DRS, and the outer dataset was LFW for all target datasets. We see that applying our metric $\mathcal{M}$ into DF allows us to select an appropriate outer dataset for each target dataset, and DRS boosts the performance of target classifiers and GANs. Furthermore, we tested CGAN with DRS and TGAN with DRS, but they underperformed our DF in terms of both the accuracy and the sample quality. This result indicates that DF improves the performances of classifiers and GANs by importing outer dataset knowledge, rather only filtering generated samples by DRS.

**Combining to Conventional Data Augmentation** We also investigated the classification performance when combining conventional DA (cDA) and DF. For training the classifiers, we adopted multiple DA transformations: random flip (for x-axis), random expand (100% to 400% of expansion ratio), random rotation (0 to 15.0 of angle). These transformations were applied to images when the images are loaded into a batch. In Table 4, we show the top-1 and top-5 classification accuracies by applying cDA and the combination of cDA and DF. The outer dataset of DF is Pascal-VOC which has the best our metric score $\mathcal{M}$ for all target datasets. In all cases of the target datasets, we see that DF outperforms only using cDA regarding to the classification performance improvements. These results indicate that DF generates useful samples that are not obtained from cDA.

The results of Table 2 imply that there are a meaningful relation between the target accuracy and the quality of the generated samples. We analyzed the relation by testing DF on 7 outer datasets. Figure 1 shows the relation between quality (FID and IS) of generated samples from DF on each outer dataset (x-axis) and test accuracy on a target dataset (y-axis). The dashed line in each panel represents linear regression, and $R$ denotes correlation coefficient. These plots indicate that the target accuracy depends on the quality of generated samples. According to these results, DF produces strong or moderate correlations between the test accuracy and both FID and IS. Further, the visualization results in Figure 3 show that the samples from DF express more clear features for each class than ones from CGAN and TGAN. Therefore, we can see that DF improves the target performance because the GANs generates target samples with high quality.

TGAN are statistically significant by using the paired t-test with 0.05 of the p-value for all of the top-1/top-5 accuracy, FID, and IS. These differences may be caused by the transfer strategies of DF and TGAN. Since TGANs try to transfer outer knowledge by fine-tuning, they suffer from forgetting knowledge (Goodfellow et al. 2013) in the pretrained GANs while retraining for the target dataset. Multi-domain training in DF seems to more effectively transfer the outer knowledge to the target samples without forgetting the knowledge than fine-tuning in TGAN.

In Domain Fusion, as shown in Improvements Section, we apply the metric $\mathcal{M}$ for outer dataset selection and DRS to improve the quality of generated samples and the performance of target classifiers. As an ablation study, we compare the performances of DF and the cases of DF without our metric $\mathcal{M}$ and DRS. Table 3 shows the results of the ablation study of DF. Note that the row of DF w/o $\mathcal{M}$ and DRS

Figure 3: Comparison of generated samples

**Evaluation of Metric $\mathcal{M}$**

We turn to evaluate our metric $\mathcal{M}$ for selecting an outer dataset. We computed $\mathcal{M}$ by using 5,000 sampled images of each outer dataset and the target datasets. Figure 2 (left column) represents the relation between our metric $\mathcal{M}$ and the top-1 accuracy by DF for each outer data. As the results of the $\mathcal{M}$ calculation, we obtain ranking of preferable outer datasets for a target dataset. In this experiment, the ranking order for CIFAR-100 is Pascal-VOC (1.5), Food-101 (2.6), DTD (4.1), Stanford Cars (5.1), Flowers (6.0), SVHN (10.5), LFW (14.2). For FGVC-Aircraft, the order is Pascal-VOC (3.5), Stanford Cars (5.6), Food-101 (5.9), DTD (7.7), Flowers (10.5), SHVN (17.5), LFW (20.7). Further, the order of ISR is Pascal-VOC (1.8), Food-101 (3.7), Stanford Cars (5.6), DTD (6.2), Flowers (9.1), SHVN (14.8), LFW (16.8). By our metric, Pascal-VOC is predicted as the best outer dataset for all of the target datasets. Since Pascal-VOC is a general image dataset composed of the various modal classes (e.g., Aeroplane, Dogs and Bottles), it has much diversity of the samples (SSIM of 0.029). Moreover, the relevance between each target dataset and Pascal-VOC is also relatively high because Pascal-VOC partially share the classes with the target datasets (CIFAR-100: FID of 50.79, FGVC-Aircraft: FID of 120.05, ISR: FID of 63.2). From these observations, general datasets such as Pascal-VOC possibly tend to be selected by our metric $\mathcal{M}$ and to contribute for target models successfully.

The lower score of $\mathcal{M}$ tends to well predict the higher top-1 accuracy on the classification ($R = -0.99$ in CIFAR-100, $R = -0.80$ in FGVC-Aircraft and ISR). We also compare $\mathcal{M}$ to other metrics: FID between the target and each outer dataset, MS-SSIM of the samples of each outer dataset (described in the center and right columns of Figure 2 respectively). Although the FID and MS-SSIM correlate with the top-1 accuracy, our metric $\mathcal{M}$ have the equal or stronger correlation than them. In particular, for FGVC-Aircraft and ISR, our metric $\mathcal{M}$ succeeds to predict better outer datasets by cooperating FID and MS-SSIM complementarily.

**Conclusion**

This paper presented Domain Fusion (DF); a generative data augmentation technique based on multi-domain learning GANs. For improving accuracy in a target task when using a low-volume target dataset, DF exploits outer knowl-edge via the samples from GANs trained on the target and outer dataset simultaneously. We also proposed a metric to select the outer dataset that consists of two perspectives: relevance and diversity. In experiments of the classification task using 3 target and 7 outer datasets, we found that DF improved the target performance and the quality of generated samples.

**References**

Antoniou, A.; Storkey, A.; and Edwards, H. 2018. Data augmentation generative adversarial networks.

Azadi, S.; Olsson, C.; Darrell, T.; Goodfellow, I.; and Odena, A. 2019. Discriminator rejection sampling. In *International Conference on Learning Representations*.

Bossard, L.; Guillaumin, M.; and Van Gool, L. 2014. Food-101 – mining discriminative components with random forests. In *European Conference on Computer Vision*.

Brock, A.; Donahue, J.; and Simonyan, K. 2019. Large scale GAN training for high fidelity natural image synthesis. In *International Conference on Learning Representations*.

Calimeri, F.; Marzullo, A.; Stamile, C.; and Terracina, G. 2017. Biomedical data augmentation using generative adversarial neural networks. In *International Conference on Artificial Neural Networks*. Springer.

Cimpoi, M.; Maji, S.; Kokkinos, I.; Mohamed, S.; ; and Vedaldi, A. 2014. Describing textures in the wild. In *Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition*.

Everingham, M.; Eslami, S. M. A.; Van Gool, L.; Williams, C. K. I.; Winn, J.; and Zisserman, A. 2015. The pascal visual object classes challenge: A retrospective. *International Journal of Computer Vision* 111.

Goodfellow, I. J.; Mirza, M.; Xiao, D.; Courville, A.; and Bengio, Y. 2013. An empirical investigation of catastrophic forgetting in gradient-based neural networks. *arXiv preprint arXiv:1312.6211*.

Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative adversarial nets. In *Advances in Neural Information Processing Systems 27*.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual

learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*.

Heusel, M.; Ramsauer, H.; Unterthiner, T.; Nessler, B.; and Hochreiter, S. 2017. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in Neural Information Processing Systems*.

Huang, G. B.; Ramesh, M.; Berg, T.; and Learned-Miller, E. 2007. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical report.

Kingma, D., and Ba, J. 2014. Adam: A method for stochastic optimization. *International Conference on Learning Representations*.

Ko, T.; Peddinti, V.; Povey, D.; and Khudanpur, S. 2015. Audio augmentation for speech recognition. In *16th Annual Conference of the International Speech Communication Association*.

Krause, J.; Stark, M.; Deng, J.; and Fei-Fei, L. 2013. 3d object representations for fine-grained categorization. In *4th International IEEE Workshop on 3D Representation and Recognition*.

Krizhevsky, A., and Hinton, G. 2009. Learning multiple layers of features from tiny images. Technical report, Citeseer.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012. Imagenet classification with deep convolutional neural networks. In *Proceedings of the 25th International Conference on Neural Information Processing Systems*.

Lučić, M.; Tschannen, M.; Ritter, M.; Zhai, X.; Bachem, O.; and Gelly, S. 2019. High-fidelity image generation with fewer labels. In *Proceedings of the 36th International Conference on Machine Learning*.

Maji, S.; Kannala, J.; Rahtu, E.; Blaschko, M.; and Vedaldi, A. 2013. Fine-grained visual classification of aircraft. Technical report.

Miyato, T., and Koyama, M. 2018. cGANs with projection discriminator. *International Conference on Learning Representations*.

Miyato, T.; Kataoka, T.; Koyama, M.; and Yoshida, Y. 2018. Spectral normalization for generative adversarial networks. *International Conference on Learning Representations*.

Netzer, Y.; Wang, T.; Coates, A.; Bissacco, A.; Wu, B.; and Ng, A. Y. 2011. Reading digits in natural images with unsupervised feature learning. In *NIPS workshop on deep learning and unsupervised feature learning*, volume 2011.

Nilsback, M.-E., and Zisserman, A. 2008. Automated flower classification over a large number of classes. In *Proceedings of the Indian Conference on Computer Vision, Graphics and Image Processing*.

Odena, A.; Olah, C.; and Shlens, J. 2017. Conditional image synthesis with auxiliary classifier gans. *Proceedings of the 34th International Conference on Machine Learning*.

Quattoni, A., and Torralba, A. 2009. Recognizing indoor scenes. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*.

Real, E.; Aggarwal, A.; Huang, Y.; and Le, Q. V. 2019. Regularized evolution for image classifier architecture search. *The Thirty-Third AAAI Conference on Artificial Intelligence*.

Salimans, T.; Goodfellow, I.; Zaremba, W.; Cheung, V.; Radford, A.; Chen, X.; and Chen, X. 2016. Improved techniques for training gans. In *Advances in Neural Information Processing Systems 29*.

Shmelkov, K.; Schmid, C.; and Alahari, K. 2018. How good is my gan? In *Proceedings of the European Conference on Computer Vision*.

Sun, C.; Shrivastava, A.; Singh, S.; and Gupta, A. 2017. Revisiting unreasonable effectiveness of data in deep learning era. In *2017 IEEE International Conference on Computer Vision*.

Tran, T.; Pham, T.; Carneiro, G.; Palmer, L.; and Reid, I. 2017. A bayesian data augmentation approach for learning deep models. In *Advances in Neural Information Processing Systems 30*.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; and Polosukhin, I. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30*.

Wang, Y.; Wu, C.; Herranz, L.; van de Weijer, J.; Gonzalez-Garcia, A.; and Raducanu, B. 2018. Transferring gans: generating images from limited data.

Wang, Z.; Simoncelli, E. P.; and Bovik, A. C. 2003. Multi-scale structural similarity for image quality assessment. In *The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*, volume 2. IEEE.

Zeyer, A.; Irie, K.; Schlüter, R.; and Ney, H. 2018. Improved training of end-to-end attention models for speech recognition. In *19th Annual Conference of the International Speech Communication Association*.

Zheng, Z.; Zheng, L.; and Yang, Y. 2017. Unlabeled samples generated by gan improve the person re-identification baseline in vitro. In *Proceedings of the IEEE International Conference on Computer Vision*.

Zhu, Y.; Aoun, M.; Krijn, M.; and Vanschoren, J. 2018. Data augmentation using conditional generative adversarial networks for leaf counting in arabidopsis plants. In *British Machine Vision Conference*.