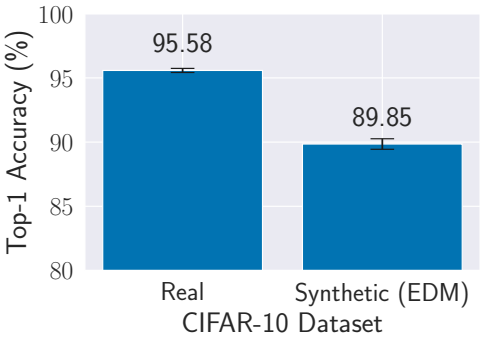
ANALYZING DIFFUSION MODELS ON SYNTHESIZING TRAINING DATASETS

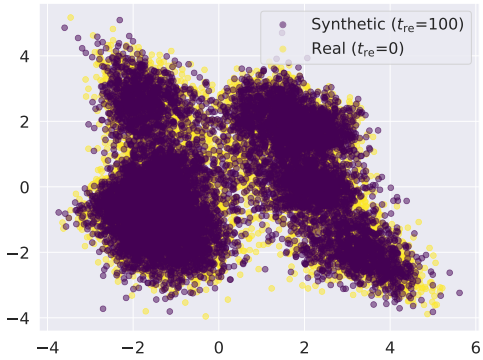
Shin'ya Yamaguchi[1,2]*
[1]NTT [2]Kyoto University

ABSTRACT

Synthetic samples from diffusion models are promising for training discriminative models as replications or augmentations of real training datasets. However, we found that the synthetic datasets degrade classification performance over real datasets when comparing them on the same dataset size. This means that the synthetic samples from modern diffusion models are less informative for training discriminative tasks. This paper investigates the gap between synthetic and real samples by analyzing the synthetic samples reconstructed from real samples through the noising (diffusion) and denoising (reverse) process of diffusion models. By varying the time steps starting the reverse process in the reconstruction, we can control the trade-off between the information in the original real data and the information produced by diffusion models. Through assessing the reconstructed samples and trained models, we found that the synthetic samples are concentrated in modes of the training data distribution as the reverse step increases, and thus, they have difficulty covering the outer edges of the distribution. In contrast, we found that these synthetic samples yield significant improvements in the data augmentation setting where both real and synthetic samples are used, indicating that the samples around modes are useful as interpolation for learning classification boundaries. These findings suggest that modern diffusion models are currently not sufficient to replicate the real training dataset in the same dataset size but are suitable for interpolating the real training samples as the augment datasets.



(a) Top-1 Test Accuracy on CIFAR-10



(b) Feature Visualization (PCA) on ResNet-18

Figure 1: Our motivation and finding. (a): Training with synthetic datasets produced by a modern diffusion model (EDM, Karras et al. (2022)) does not replicate the classification performance of the real dataset in the same dataset size. (b) We input synthetic samples to a classifier trained on a real dataset and found that the features of synthetic samples concentrate on the modes of real feature distribution and do not cover the outer edge of the distribution. This means that the synthetic samples from diffusion models are less informative for training classifiers than real samples.

*Email: shinya.yamaguchi@ntt.com

1 INTRODUCTION

In the past decade, deep generative models have witnessed remarkable advancements in generating high-quality synthetic samples that are human-indistinguishable from real data. Among these generative models, diffusion models (Ho et al., 2020) have attracted much attention because they can outperform the existing generative models such as GANs (Goodfellow et al., 2014) and VAEs (Kingma & Welling, 2014) by learning denoising (reverse) processes through score-based likelihood maximization (Dhariwal & Nichol, 2021; Rombach et al., 2022).

The high-quality samples from diffusion models naturally raise research interest in their applicability for training target discriminative models (e.g., classifiers), and recent studies intensively develop training techniques utilizing synthetic samples from diffusion models. For instance, He et al. (2022) demonstrated that synthetic samples from pre-trained text-image diffusion models (e.g., Stable Diffusion (Rombach et al., 2022)) can achieve impressive zero-/few-shot learning performance by querying the synthetic training samples with crafted prompts representing target dataset categories. Moreover, Burg et al. (2023), Azizi et al. (2023), and Dunlap et al. (2023) highlighted the potential of diffusion models for data augmentation application. They investigated diffusion-based data augmentation methods by modifying diffusion models with nearest neighbor exploration in sampling (Brown et al., 2020), scaling up models (Azizi et al., 2023), and customizing text prompts for querying samples (Dunlap et al., 2023). However, in contrast to these remarkable successes, we observed that models trained on synthetic samples are inferior to models trained on real data when the diffusion models trained only on target datasets (Figure 1a)[1]. This indicates that the synthetic samples from modern diffusion models are less informative than real samples, and there is a gap between real and synthetic datasets in terms of training classifiers. In this paper, by analyzing diffusion models, we aim to answer the following important and open research question: *What is the cause of the gap between real and synthetic datasets?*

This paper mainly analyzes the gap between synthetic and real datasets on dataset replication, i.e., generating the same amount of synthetic samples as the real dataset and then leveraging only the synthetic samples to train the classifier. We focus on two perspectives: (i) the quality of synthetic samples and (ii) the impact of synthetic samples on training classification models. To assess the gap, we introduce the concept of **real sample reconstruction** utilizing the diffusion and reverse process. Real sample reconstruction consists of corrupting real samples by the diffusion process up to pre-defined steps and then restoring the corrupted samples by the reverse process. We refer to the pre-defined step as **reverse step**. By varying the reverse steps, we can continuously control the trade-off between the remaining information from the input real samples and the synthetic information injected by the reverse process (Figure 2). We empirically investigate how the synthetic information affects the sample quality and the classification performance.

Our experimental findings in dataset replication are summarized as follows:

- Diffusion models generate synthetic samples that are nearly indistinguishable as real or fake compared to competitive models such as GANs.
- Increasing the reverse steps (i.e., making sample properties closer to synthetic samples) leads to gradual degradation in the sample quality and classifier performance.
- Leveraging synthetic samples for training classifiers does not adversely affect the tendency of classifier outputs (e.g., attention map).
- Synthetic samples are easily distinguished as fake by a classifier with deep neural networks.
- Synthetic samples concentrate near the modes of the data distribution in the feature space of classifiers (Fig. 1b), and a longer reverse process brings the sample closer to the mode.
- Increasing a large number of synthetic samples can achieve a classification performance comparable to that of the real dataset, but it requires more than three times the number of samples of the real dataset.

These findings suggest that modern diffusion models have limitations in generating samples away from the modes. This can be because diffusion models learn to denoise samples in the direction that

[1]Note that, here, we do not use pre-trained diffusion models like Stable Diffusion but train diffusion models from scratch in order to assess the limitations of the diffusion model itself.

maximizes the likelihood at each step in the reverse process (Song et al., 2021). That is, the reverse process may bring a sample closer to a typical mode and make the sample less informative for training classifiers. Therefore, with the same number of samples, the synthetic dataset can degrade accuracy over real datasets due to the less information far from the modes.

Furthermore, we analyze the synthetic samples in the data augmentation applications. Based on the analysis of dataset replication, we can expect that the synthetic samples around the modes with high likelihood are useful for learning the interpolations between real samples when we combine them with real samples as a data augmentation in training classifiers. Indeed, we experimentally confirm that the data augmentation with the synthetic samples significantly improves baselines. In particular, we found that the synthetic samples generated by real sample reconstruction yield further improvements. This indicates that diffusion models are good at generating synthetic samples that interpolate between real samples and that they can propagate more detailed pattern differences to classifiers through real sample reconstruction. We believe these observations and implications will help drive future developments in synthesizing training datasets by diffusion and other generative models.

2 Related Work

Diffusion model (Sohl-Dickstein et al., 2015; Ho et al., 2020) is a class of generative models inspired by thermodynamics. They learn iteratively denoising process called *reverse process* corresponding to the corruption process adding noises called *diffusion process*. Song et al. (2021) revealed the relationship between diffusion models and denoising score matching with stochastic gradient Langevin dynamics and explained the optimization of the reverse process as score-based likelihood maximization. By introducing conditional guidance in the reverse process, diffusion models successfully control output by class labels (Dhariwal & Nichol, 2021; Ho & Salimans, 2022) and text embedding (Ramesh et al., 2022; Rombach et al., 2022), and a number of subsequent studies are still being published.

Since diffusion models can achieve high-quality synthetic samples in comparison to other generative models (e.g., GANs and VAEs) (Dhariwal & Nichol, 2021), recent studies investigated the capability of diffusion models as a source of training datasets (He et al., 2022; Burg et al., 2023; Azizi et al., 2023; Dunlap et al., 2023). These studies utilized text-image pre-trained diffusion models such as Stable Diffusion (Rombach et al., 2022) for generating synthetic training samples and succeeded in improving classification performance by adding the synthetic samples into training datasets. In contrast, this paper focuses on class conditional diffusion models trained only on target datasets from scratch and does **NOT** consider large pre-trained diffusion models (e.g., Stable Diffusion) to discard the effects of knowledge transfer from external pre-trained datasets.

3 Preliminary

Here, we briefly introduce the principles of diffusion models and real sample reconstruction used for our main analysis.

3.1 Diffusion Models

A diffusion model learns a data distribution $p(x)$ by optimizing the parameterized reverse (denoising) process assuming Markov chain with length T (Sohl-Dickstein et al., 2015; Ho et al., 2020), which corresponds to the forward diffusion process. Specifically, most modern diffusion models are optimized by minimizing the family of the following loss function with respect to the neural network parameter θ (Ho et al., 2020; Dhariwal & Nichol, 2021; Rombach et al., 2022).

$$\mathcal{L}(\theta) = \mathbb{E}_{x,\epsilon\sim\mathcal{N}(0,1),t}\left[\|\epsilon - \epsilon_\theta(x_t, t)\|_2^2\right], \tag{1}$$

where ϵ_θ is the denoising autoencoder parameterized by θ, t is the time step randomly sampled from $\{1, \cdots, T\}$, x is the input, and x_t is a noisy variant of x. In inference time, a synthetic sample $\hat{x}$ is generated by sequentially applying the denoising function for each t from T to 1 as

$$x_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(x_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(x_t, t) + \sigma_t z\right), \tag{2}$$

Algorithm 1 Real Sample Reconstruction

Require: Real sample x, reverse step $t_{\text{re}} > 1$
Ensure: Reconstructed sample $\hat{x}$
// Corrupting x with diffusion process for t_{re}
$x_0 \leftarrow x$
for $t = 1, \cdots, t_{\text{re}}$ **do**
  $x_t \leftarrow \sqrt{\alpha_t} x_{t-1} + \sqrt{1-\alpha_t}\epsilon_{t-1}$
end for
// Restoring $x_{t_{\text{re}}}$ with reverse process
for $t = t_{\text{re}}, \cdots, 1$ **do**
  $\hat{x}_{t-1} \leftarrow \frac{1}{\sqrt{\alpha_t}}\left(x_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(x_t, t) + \sigma_t z\right)$
end for
$\hat{x} \leftarrow \hat{x}_0$

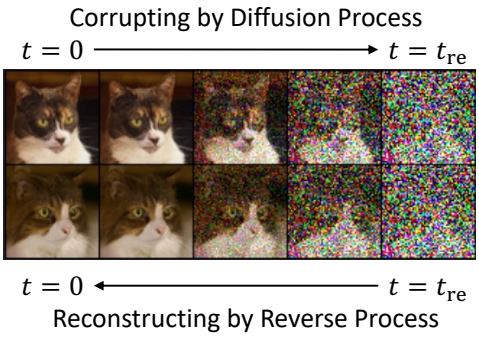


Figure 2: Real Sample Reconstruction

where $\alpha_t = 1 - \beta_t$, β_t is a scheduled variance in $\{\beta_1, \cdots, \beta_T\}$, $\bar{\alpha}_t = \prod_{s=1}^{t} \alpha_s$, $\sigma_t = \sqrt{\frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t}$ and $z \sim \mathcal{N}(0, 1)$. Song et al. (2021) showed that this denoising process corresponds to stochastic gradient Langevin dynamics, which produces samples by iterative updating x_t with the score $\nabla_x \log p(x)$:

$$x_t = x_{t-1} + \frac{\delta}{2}\nabla_x \log p(x_{t-1}) + \sqrt{\delta} z, \tag{3}$$

where δ is a step size. In this paper, we implement diffusion models with conditional EDM (Karras et al., 2022) to generate a synthetic labeled dataset for training classifiers.

3.2 Real Sample Reconstruction

We introduce real sample reconstruction, which produces intermediate samples between real and synthetic by exploiting the diffusion and reverse process. Real sample reconstruction first corrupts a real sample by the diffusion process from 0 to a specified time step t_{re}, and then recovers the corrupted sample by the reverse process from t_{re} to 0. Given a real data point x, we produce a reconstructed sample $\hat{x}$ with a reverse time step t_{re} by following Algorithm 1. This reconstruction algorithm is similar to SDEdit (Meng et al., 2022), which is an image-editing method based on the reconstruction of guide images by diffusion models. Intuitively, $\hat{x}$ is fully real when $t_{\text{re}} = 0$, a fully synthetic when $t_{\text{re}} = T$, and half of real and synthetic when $t_{\text{re}} = T/2$ as depicted in Figure 2. Unlike the purpose of SDEdit, we aim to produce intermediate samples of real and synthetic by simply inputting real images into the diffusion and reverse process.

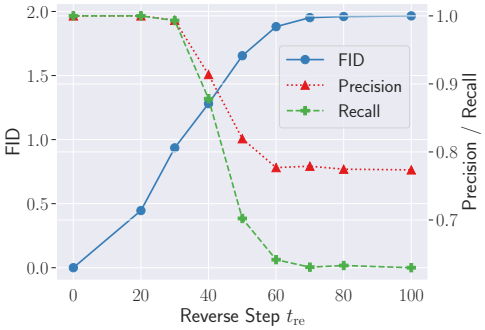
4 Analysis on Dataset Replication

In this section, we report the experimental results of the dataset replication scenario where we produce the same number of synthetic samples as the real dataset and train classifiers by using only the synthetic dataset. We assess (i) the quality of reconstructed samples from diffusion models and (ii) the effects on classifiers trained on the reconstructed samples. We mainly used the CIFAR-10 dataset (Krizhevsky & Hinton, 2009) as the target dataset, the CIFAR-10 pre-trained conditional EDM (Karras et al., 2022) ($T = 100$) as the diffusion model, and ResNet-18 (He et al., 2016) as the classifier architecture.

4.1 Analysis on Synthetic Sample

Evaluation Protocol. To analyze reconstructed synthetic samples, we measured Frechèt inception distance (FID) (Heusel et al., 2017), precision/recall (Kynkäänniemi et al., 2019), and fake detection accuracy (Frank et al., 2020). Among them, FID and precision/recall are measured on the ImageNet pre-trained feature extractor. FID evaluates the gap between real and synthetic datasets, and precision/recall evaluate the probabilities that synthetic/real samples fall within the real/synthetic distributions. Fake detection accuracy is calculated on a classifier trained to distinguish real and synthetic samples on both the pixel and frequency domains. This is useful to find out how different

(a) Reconstructed Samples



(b) FID (Heusel et al., 2017) and Precision/Recall (Kynkäänniemi et al., 2019)

Figure 3: Quality Assessments of Synthetic Sample

Table 2: Top-1 Test Accuracy (%) on CIFAR-10 and CIFAR-100

Reverse Step t_{re}	CIFAR-10	CIFAR-100
0 (Fully Real)	$95.58^{\pm.16}$	$86.70^{\pm.08}$
30	$94.85^{\pm.15}$	$86.16^{\pm.61}$
50	$93.33^{\pm.38}$	$81.07^{\pm.90}$
70	$91.57^{\pm.55}$	$77.88^{\pm.31}$
100 (Fully Synthetic)	$89.85^{\pm.41}$	$77.68^{\pm.73}$

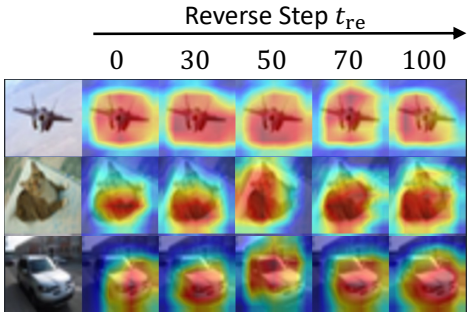


Figure 4: GradCAM Visualization

synthetic and real samples are in terms of input to the classifier. We used 50,000 synthetic samples and 50,000 real samples to calculate the metrics.

Sample Quality. We first show the visualization of the reconstructed samples in Figure 3a. We reconstructed the samples on every 10 steps of $t_{\text{re}} \in [20, 80]$ from EDM. As the reverse step t_{re} increases, the reconstructed samples gradually lose information on the input real sample, and represent information on the synthetic sample. Nevertheless, in visual quality, it is hard to distinguish between a synthetic sample and a real sample for every t_{re}. Next, we show the FID and precision/recall scores calculated on the reconstructed samples in Figure 3b. We see that increasing reverse steps progressively degrades all the quantitative metrics. This indicates that the reverse process may be harmful to maintain the information on the real samples. In particular, the reverse process significantly degrades recall scores, indicating that the synthetic sample does not sufficiently cover the training data distribution.

Fake Detection Accuracy. We demonstrate the fake detection accuracy on the synthetic samples. To evaluate the worst quality case, we used $t_{\text{re}} = 100$ in this experiment. Table 1 shows the fake detection accuracy in the pixel domain and frequency domain (DCT). For comparison, we also print the result of StyleGAN3 (Karras et al., 2021). The higher scores mean easier samples to be detected as fake. While the StyleGAN3 samples were easily distinguished, fewer EDM samples were detected as fake. These results suggest that although the synthetic datasets from diffusion models differ in quantitative measures such as FID, their properties as input to the classifier are almost the same as those of the real samples.

Table 1: Fake Detection Accuracy (CIFAR-10)

Generative Model	Accuracy (%) Pixel	DCT
StyleGAN3 (Karras et al., 2021)	89.56	53.62
EDM (Karras et al., 2022)	56.15	58.91

4.2 Analysis on Training Classifiers

Evaluation Protocol. We analyze trained classifiers on reconstructed synthetic samples by varying the reverse step t_{re}. We evaluate test classification accuracy, attention map visualization by

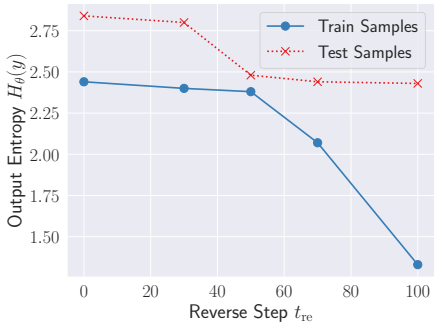


Figure 5: Output Entropy of Classifier

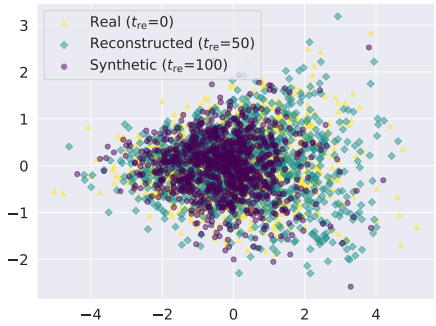


Figure 6: Feature Visualization (`truck` class)

GradCAM (Selvaraju et al., 2017), output entropy, and feature visualization by principle component analysis (PCA). We trained ResNet-18 classifiers for 100 epochs on the synthetic CIFAR-10 datasets yielded by real sample reconstruction with $t_{\text{re}} = 30, 50, 70$, and tested them on the real CIFAR-10 test set. We used the SGD optimizer with a learning rate of 0.01 dropping by multiplying 0.1 for every 30 epochs. We also show the results when using the real dataset (i.e., $t_{\text{re}} = 0$) and the fully synthetic dataset by the reverse process with random noise (i.e., $t_{\text{re}} = 100$). For GradCAM and feature visualization, we used the output of `block4` on ResNet-18. We calculate the marginal output entropy by

$$H_\theta(y) = -\frac{1}{N}\sum_{i}^{N}\sum_{j}^{C} p_\theta(y=j|x_i)\log(p_\theta(y=j|x_i)), \tag{4}$$

where N is a dataset size, C is a class number, $p_\theta(y=j|x_i) = \frac{\exp(f_\theta(x_i)[j])}{\sum_k^C \exp(f_\theta(x_i)[k])}$, f_θ is a classifier.

Classification Performance. Table 2 shows the top-1 test accuracy on the real CIFAR-10 test set for each reverse time step t_{re}. Similar to the sample quality shown in the previous section, we see that the performance of the classifier degrades as the reverse step increases. This implies that the reverse process of diffusion models eliminates information important for solving classification tasks from the original real sample.

Attention Map. Figure 4 shows the visualizations of GradCAM. We input real test samples of CIFAR-10 for each trained model. Interestingly, while the test accuracy is degraded by real sample reconstruction, the synthetic samples used for training do not change the attention of the trained models. This means that the synthetic sample itself has no noticeable negative impact on learning classification tasks.

Output Entropy. Next, we assess the quantitative effects on the classifier's prediction. To this end, we used the classifier trained on the real CIFAR-10 because we can consider it an ideal classifier for the purpose of training dataset replication. Figure 5 plots the output entropy $H_\theta(y)$. We calculated $H_\theta(y)$ by inputting the synthetic samples into the classifier. In Figure 5, `Train Samples` and `Test Samples` mean the calculated entropy scores on the reconstructed samples from real samples of the train/test set. Note that again, in this experiment, we used only the classifier trained on the real CIFAR-10 to assess the characteristics of synthetic samples. We see that increasing the reverse step makes the synthetic samples low entropy, indicating easy samples to be classified. Thus, the diffusion model tends to produce a typical sample that is representative of the class by the reverse process.

Feature Visualization. Finally, we visualize the features of synthetic samples to examine how synthetic samples behave on the classifier. Similar to the previous paragraph, we used the classifier trained on the real CIFAR-10 for feature visualization. We applied PCA to the extracted features of input synthetic samples and reduced the dimension to two. Figure 1b and 6 are the visualization results of all class samples and `truck` class samples, respectively. In Figure 1b, the synthetic sample is concentrated inside the distribution formed by the real samples, while its outer edges are not well covered. Meanwhile, in Figure 6, the reconstructed at $t_{\text{re}} = 50$ appears to cover the region

where the sample at $t_{\mathrm{re}} = 100$ is scarce. These results suggest that the synthetic samples from diffusion models tend to concentrate the center (mode) of training data distribution, and the reverse process gradually pulls the synthetic samples toward the modes of training distribution.

4.3 Discussion

Through the empirical analysis in the previous subsections, we observed that the modern diffusion models can produce quite realistic synthetic samples, but they still have insufficient generative performance for replicating training datasets for classifiers. In particular, the reverse process of diffusion models seems to gradually concentrate the synthetic samples toward the modes of the training data distribution. We can explain this phenomenon by the interpretation of the diffusion model as a score-based generative model. As we discussed in Sec. 3.1, a reverse process corresponds to a step of stochastic gradient Langevin dynamics as shown in Eq. (3). That is, a reverse step contains the gradient of log-likelihood (score) $\nabla_x \log p(x)$. Therefore, the iterative denoising of samples by multiple reverse steps means that the samples are moving closer to the region of high likelihood, i.e., the mode of the distribution. Eq. (3) has a disturbance term by z to prevent the concentration of sampling near the modes, but our experimental results suggest that this cannot be completely prevented for the purpose of replicating training datasets.

5 Analysis on Data Augmentation

The analysis in Section 4 shows that the synthetic samples from diffusion models are less informative than real samples due to the reverse process, which guides the samples to the higher likelihood regions. Conversely, in the higher likelihood regions, we can expect that the synthetic sample quality is sufficient for training classifiers as shown in Figure 4, and they perform as the interpolation of real samples if we can access the real dataset (Figure 6). Therefore, this section investigates the performance effects when adding the synthetic samples to the real datasets, i.e., data augmentation. Further, this section also introduces an application of real sample reconstruction (Algorithm 1) to data augmentation, which can produce interpolating samples as shown in Figure 6.

Evaluation Protocol. To assess the practical performance on data augmentation, we used real-world target datasets: Aircraft (Maji et al., 2013), Bird (Welinder et al., 2010), and Car (Krause et al., 2013). As the diffusion model and classifier, we used EDM and ResNet-18, as well as the previous sections. We varied the size of synthetic datasets by the size ratio to the real dataset. For example, $\times 5$ means the use of a synthetic dataset that is five times the size of the real dataset. We report the test accuracy of classifiers trained by the same setting as Section 4. For the synthetic data augmentation, we trained models by simultaneously using individual batches of real and synthetic samples, i.e., we used a real batch of 64 samples and a synthetic batch of 64 samples for each iteration.

Improvements by Synthetic Data Augmentation. We first examine the effects of data augmentation with synthetic datasets and the impact of synthetic dataset size. Table 3 shows the test accuracy on Aircraft, where **Baseline** is trained only on the real dataset, **Synthetic Only** is trained only on synthetic samples (i.e., dataset replication), **Naïve** is trained on the real and synthetic samples naïvely generated by the diffusion model. We see that the Naïve models stably outperformed the baseline and the dataset replication, indicating that the synthetic samples indeed supplementarily help the classification performance in the data augmentation setting. This is consistent with our expectations, i.e., the synthetic samples in the high likelihood region are helpful for training classifiers. Interestingly, the accuracy can be further improved by increasing the number of synthetic samples. This is because an increase in synthetic samples enlarges the diversity of the sample, and thus, they are useful for classification. In fact, when increasing the synthetic dataset size in the dataset replication setting ($\times 3$ and $\times 5$ of the Synthetic Only row in Table 3), the accuracy is equal to or exceeds the baseline. These observations suggest that although synthetic samples provide less information per sample than real samples, using more synthetic samples can provide useful information to the classifier.

Informativeness of Reconstructed Samples. Table 3 also shows the results of **Reconstructed**, which is trained on the real and reconstructed synthetic samples by real sample reconstruction (Al-

Table 3: Top-1 Test Accuracy on Aircraft (ResNet-18)

Method	Top-1 Accuracy (%)		
Baseline (Real Only)	$64.71^{\pm.91}$		
Dataset Replication	×1	×3	×5
Synthetic Only	$54.02^{\pm.47}$	$64.68^{\pm.34}$	$69.03^{\pm.44}$
Synthetic Data Augmentation	×1	×3	×5
Naïve ($t_{\mathrm{re}} = 100$)	$69.23^{\pm.25}$	$72.99^{\pm.14}$	$75.10^{\pm.37}$
Reconstructed ($t_{\mathrm{re}} = 25$)	$63.17^{\pm.42}$	$63.27^{\pm.25}$	$64.05^{\pm.97}$
Reconstructed ($t_{\mathrm{re}} = 50$)	$64.41^{\pm.44}$	$66.20^{\pm.12}$	$67.85^{\pm.36}$
Reconstructed ($t_{\mathrm{re}} = 75$)	$67.80^{\pm.59}$	$71.60^{\pm.38}$	$73.06^{\pm.14}$
Reconstructed ($t_{\mathrm{re}} = \mathrm{rand}(0, 50)$)	$65.41^{\pm.35}$	$65.15^{\pm.78}$	$66.62^{\pm.63}$
Reconstructed ($t_{\mathrm{re}} = \mathrm{rand}(0, 100)$)	$69.08^{\pm.20}$	$72.36^{\pm.26}$	$73.87^{\pm.30}$
Reconstructed ($t_{\mathrm{re}} = \mathrm{rand}(50, 100)$)	$\mathbf{69.85}^{\pm.29}$	$\mathbf{73.22}^{\pm.32}$	$\mathbf{75.52}^{\pm.21}$
Reconstructed ($t_{\mathrm{re}} = \mathrm{rand}(75, 100)$)	$67.69^{\pm.22}$	$72.11^{\pm.08}$	$73.05^{\pm.60}$

Table 4: Top-1 Test Accuracy on Multiple Target Datasets (ResNet-18)

	Aircraft	Bird	Car
Baseline (Real Only)	$64.71^{\pm.91}$	$61.73^{\pm.30}$	$74.58^{\pm.29}$
Dataset Replication			
Synthetic Only (×1)	$54.02^{\pm.47}$	$43.10^{\pm.44}$	$49.91^{\pm.32}$
Synthetic Only (×3)	$64.78^{\pm.20}$	$54.48^{\pm.51}$	$70.26^{\pm.51}$
Synthetic Only (×5)	$69.03^{\pm.44}$	$57.85^{\pm.39}$	$75.72^{\pm.55}$
Traditional Data Augmentation			
RandAugment (Cubuk et al., 2020)	$66.13^{\pm.65}$	$64.09^{\pm.11}$	$77.21^{\pm.25}$
TrivialAugment (Müller & Hutter, 2021)	$67.65^{\pm.24}$	$\mathbf{65.42}^{\pm.32}$	$78.61^{\pm.50}$
Synthetic Data Augmentation (×5)			
Naïve ($t_{\mathrm{re}} = 100$)	$75.10^{\pm.37}$	$63.51^{\pm.19}$	$81.33^{\pm.17}$
Reconstructed ($t_{\mathrm{re}} = \mathrm{rand}(50, 100)$)	$\mathbf{75.52}^{\pm.21}$	$64.23^{\pm.56}$	$\mathbf{83.17}^{\pm.01}$

gorithm 1). We tried the fixed reverse step $t_{\mathrm{re}} \in \{25, 50, 75\}$. Unfortunately, contrary to our expectations, the reconstructed samples generated from a fixed t_{re} did not achieve better accuracy than Naïve. In particular, the reconstructed sample with $t_{\mathrm{re}} = 25$, i.e., closer to the original real sample like Figure 3a, underperformed the baseline. This may be because small reverse steps produce almost the same samples as the input real samples, which promotes overfitting of the classifier. On the other hand, since the larger reverse steps tend to achieve higher accuracy, synthetic samples that are different from the original real samples yield greater improvement. For further analysis, we tried the randomized reverse step for each sample by a function $\mathrm{rand}(X_{\mathrm{lower}}, X_{\mathrm{upper}})$ generating random numbers from X_{lower} to X_{upper} ($X_{\mathrm{lower}} < X_{\mathrm{upper}}$). Table 3 shows that, as with fixed steps, the reconstructed samples from smaller steps of $[0, 50]$ consistently have smaller gains in accuracy. Importantly, the case of $t_{\mathrm{re}} = \mathrm{rand}(50, 100)$ outperformed Naïve with statistical significance. This indicates that the synthetic samples interpolating real samples by real sample reconstruction have the potential to improve the classifiers more effectively than the naïve random sampling.

Comparison to Traditional Data Augmentation on Multiple Target Datasets. Finally, we examine the practicality of the synthetic data augmentation using naïvely generated and reconstructed samples. Here, we compare the performances of the synthetic data augmentation with state-of-the-art traditional data augmentation techniques: RandAugment (Cubuk et al., 2020) and TrivialAugment (Müller & Hutter, 2021). Table 4 shows the comparison on multiple target datasets. First, the synthetic data augmentation methods significantly outperformed the traditional data augmentation methods on the Aircraft and Car datasets. This is a very surprising result because previous studies using GANs have reported that synthetic data augmentation provides an improvement equal to or less than traditional data augmentation (Shmelkov et al., 2018; Yamaguchi et al., 2020; 2022; 2023). On the contrary, the results on the Bird dataset were not so significant. This can be because the synthetic samples originally lacked information for classifier training in the case of Bird. Indeed, the results of dataset replication in Table 4 show that, even if the number of synthetic samples is increased, it did not achieve an accuracy higher than Baseline (Real Only) for Bird. Moreover, the FID score of EDM for Bird was 14.4, while those for Aircraft and Car were 4.1 and 8.5, respectively. Therefore, if the generative model cannot produce high-quality samples, obtaining the benefits of synthetic data augmentation is difficult because of the lack of information in the synthetic samples.

6 Conclusion and Takeaway

This paper empirically showed the limitations of diffusion models for synthesizing datasets for training classifiers. Modern diffusion models are not sufficient to replicate entire training datasets due to the sampling concentration near the data distribution modes. This can be caused by the reverse denoising process, which naturally moves the samples toward the modes. From these observations, one of the important takeaways is that we should improve diffusion models to cover the outside edges of training data distributions. Another one is that, currently, the data augmentation applications of diffusion models, which utilize both real and synthetic samples, can be more suitable to train high-performance classifiers than replicating entire training datasets and utilizing only synthetic samples. We believe that these observations and implications will be helpful for future research.

Reproducibility Statements. We build our experimental environment with PyTorch and existing open-source repositories as follows.

- **EDM** (Karras et al., 2022) (Official PyTorch Implementation): https://github.com/NVlabs/edm
- **ResNet** (He et al., 2016) (torchvision): https://pytorch.org/vision/stable/models/resnet.html
- **GradCAM** (Selvaraju et al., 2017) (Unofficial Pytorch Implementation): https://github.com/jacobgil/pytorch-grad-cam
- **CIFAR**-10 (Krizhevsky & Hinton, 2009): https://www.cs.toronto.edu/~kriz/cifar.html
- **Aircraft** (Maji et al., 2013): https://www.robots.ox.ac.uk/~vgg/data/fgvc-aircraft/
- **Bird** (Welinder et al., 2010): https://www.vision.caltech.edu/visipedia/CUB-200-2011.html
- **Car** (Krause et al., 2013): https://www.kaggle.com/datasets/jessicali9530/stanford-cars-dataset

Further, to reproduce our results precisely, we will publish our code to an open-source repository on GitHub after acceptance.

REFERENCES

Shekoofeh Azizi, Simon Kornblith, Chitwan Saharia, Mohammad Norouzi, and David J. Fleet. Synthetic data from diffusion models improves imagenet classification. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856. URL https://openreview.net/forum?id=DlRsoxjyPm.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 2020.

Max F Burg, Florian Wenzel, Dominik Zietlow, Max Horn, Osama Makansi, Francesco Locatello, and Chris Russell. Image retrieval outperforms diffusion models on data augmentation. *Transactions on Machine Learning Research*, 2023.

Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 702–703, 2020.

Prafulla Dhariwal and Alex Nichol. Diffusion models beat gans on image synthesis. In *Advances in Neural Information Processing Systems*, 2021.

Lisa Dunlap, Alyssa Umino, Han Zhang, Jiezhi Yang, Joseph E Gonzalez, and Trevor Darrell. Diversify your vision datasets with automatic diffusion-based augmentation. In *Advances in neural information processing systems*, 2023.

Joel Frank, Thorsten Eisenhofer, Lea Schönherr, Asja Fischer, Dorothea Kolossa, and Thorsten Holz. Leveraging frequency analysis for deep fake image recognition. 2020.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems 27*, 2014.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016.

Ruifei He, Shuyang Sun, Xin Yu, Chuhui Xue, Wenqing Zhang, Philip Torr, Song Bai, and Xiaojuan Qi. Is synthetic data from generative models ready for image recognition? *arXiv preprint arXiv:2210.07574*, 2022.

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*, 2017.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in neural information processing systems*, 2020.

Tero Karras, Miika Aittala, Samuli Laine, Erik Härkönen, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Alias-free generative adversarial networks. *Advances in Neural Information Processing Systems*, 2021.

Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. In *Advances in neural information processing systems*, 2022.

Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *International Conference on Learning Representations*, 2014.

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

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

Tuomas Kynkäänniemi, Tero Karras, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Improved precision and recall metric for assessing generative models. *Advances in Neural Information Processing Systems*, 2019.

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

Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. Sdedit: Guided image synthesis and editing with stochastic differential equations. In *International Conference on Learning Representation*, 2022.

Samuel G Müller and Frank Hutter. Trivialaugment: Tuning-free yet state-of-the-art data augmentation. In *Proceedings of the IEEE/CVF international conference on computer vision*, 2021.

Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, 2017.

Konstantin Shmelkov, Cordelia Schmid, and Karteek Alahari. How good is my gan? In *Proceedings of the European Conference on Computer Vision*, 2018.

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, 2015.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representation*, 2021.

P. Welinder, S. Branson, T. Mita, C. Wah, F. Schroff, S. Belongie, and P. Perona. Caltech-UCSD Birds 200. Technical report, California Institute of Technology, 2010.

Shin'ya Yamaguchi, Sekitoshi Kanai, and Takeharu Eda. Effective data augmentation with multi-domain learning gans. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020.

Shin'ya Yamaguchi, Sekitoshi Kanai, Atsutoshi Kumagai, Daiki Chijiwa, and Hisashi Kashima. Transfer learning with pre-trained conditional generative models. *arXiv preprint arXiv:2204.12833*, 2022.

Shin'ya Yamaguchi, Sekitoshi Kanai, Atsutoshi Kumagai, Daiki Chijiwa, and Hisashi Kashima. Regularizing neural networks with meta-learning generative models. In *Advances in Neural Information Processing Systems*, 2023.