MITSUBISHI ELECTRIC RESEARCH LABORATORIES
http://www.merl.com

FX-GAN: Self-Supervised GAN Learning via Feature Exchange

Huang, Rui; Xu, Wenju; Lee, Teng-Yok; Cherian, Anoop; Wang, Ye; Marks, Tim

TR2020-014 February 20, 2020

Abstract

We propose a self-supervised approach to improve the training of Generative Adversarial Networks (GANs) via inducing the discriminator to examine the structural consistency of images. Although natural image samples provide ideal examples of both valid structure and valid texture, learning to reproduce both together remains an open challenge. In our approach, we augment the training set of natural images with modified examples that have degraded structural consistency. These degraded examples are automatically created by randomly exchanging pairs of patches in an image's convolutional feature map. We call this approach feature exchange. With this setup, we propose a novel GAN formulation, termed Feature eXchange GAN (FX-GAN), in which the discriminator is trained not only to distinguish real versus generated images, but also to perform the auxiliary task of distinguishing between real images and structurally corrupted (feature-exchanged) real images. This auxiliary task causes the discriminator to learn the proper feature structure of natural images, which in turn guides the generator to produce images with more realistic structure. Compared with strong GAN baselines, our proposed self-supervision approach improves generated image quality, diversity, and training stability for both the unconditional and class-conditional settings.

IEEE Winter Conference on Applications of Computer Vision (WACV)

FX-GAN: Self-Supervised GAN Learning via Feature Exchange

Rui Huang[†]
Carnegie Mellon University
ruih2@alumni.cmu.edu

Wenju Xu[†]
University of Kansas
xuwenju@ku.edu

Teng-Yok Lee Anoop Cherian Ye Wang Tim K. Marks
Mitsubishi Electric Research Laboratories (MERL), Cambridge, MA
{tlee,cherian,yewang,tmarks}@merl.com

Abstract

We propose a self-supervised approach to improve the training of Generative Adversarial Networks (GANs) via inducing the discriminator to examine the structural consistency of images. Although natural image samples provide ideal examples of both valid structure and valid texture, learning to reproduce both together remains an open challenge. In our approach, we augment the training set of natural images with modified examples that have degraded structural consistency. These degraded examples are automatically created by randomly exchanging pairs of patches in an image's convolutional feature map. We call this approach feature exchange*. With this setup, we propose a novel GAN formulation, termed Feature eXchange GAN (FX-GAN), in which the discriminator is trained not only to distinguish real versus generated images, but also to perform the auxiliary task of distinguishing between real images and structurally corrupted (feature-exchanged) real images. This auxiliary task causes the discriminator to learn the proper feature structure of natural images, which in turn guides the generator to produce images with more realistic structure. Compared with strong GAN baselines, our proposed self-supervision approach improves generated image quality, diversity, and training stability for both the unconditional and class-conditional settings.*

1. Introduction

Generative adversarial networks (GANs) [9] learn complex data distributions by pitting a generator and a discriminator against each other in an adversarial game. The generator attempts to generate valid data from a *learned* data distribution, while the discriminator is trained to distinguish these generated data from samples of the *true* data distribution. These two components are optimized in a min-max game, towards an equilibrium where the discriminator is unable to distinguish whether the generated data are real or fake. Given that the key learning signal for training the generator comes from the discriminator, the losses against which the discriminator is trained implicitly guides effective training of the generator. Thus, training the discriminator to learn useful image properties may produce better generators. This leads to more stable [21] and scalable [2] optimization via suitably regularizing the otherwise unstable saddle-point seeking min-max game.

Discriminators in vanilla GANs [9] use only a single-bit label, namely whether the input is real or fake. There have been several previous attempts at providing the discriminator with meta-information regarding the real samples. For example, conditional GANs [19] extend the generator and discriminator to use auxiliary information, such as the class label of an input image [30, 22], leading to better quality generation [27]. Similarly, the style and structure of the real images are used in Wang et al. [32], while photo realism is captured in Huang et al. [11]. However, given the huge amount of data needed for training GANs, auxiliary tasks that require expensive labels would not be practical.

Another popular method for pre-training machine learning models is self-supervision [6, 20, 35], in which training labels are derived automatically by defining a task over the input data. For example, in [23, 28], the image patches are shuffled, and the task is to recover the shuffling permutation, under the implicit assumption that a network which is able to perform well on this deliberate task should have learned the structure of images adequately. Self-supervised learning has also recently been explored for regularizing GAN training. For example, in Chen et al. [3], right-angle rotations are applied to the input images [3], and the discriminator is asked to predict the correct rotation angle. Such a loss on the discriminator essentially equips it for recognizing global structure in real data.

[†]Work done while interning at MERL.

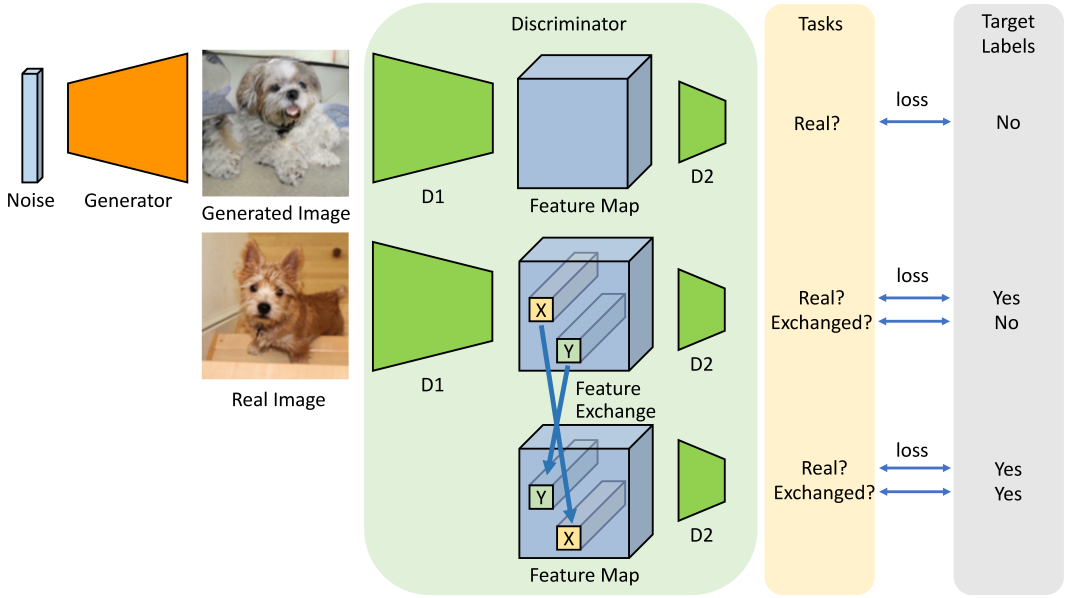


Figure 1. Overview of our proposed FX-GAN framework. In addition to the task of distinguishing between generated and real data samples, we introduce the feature exchange operation and a new task of distinguishing between exchanged and original data samples(images). D1 and D2 are neural network layers of the discriminator. The feature maps produced by D1 from real images(second row) are duplicated. The duplicated feature maps(third row) are being applied to the exchange operator and pass through D2 to produce a prediction for the two tasks. The exchange operation exchanges feature blocks at two random spatial locations, as illustrated by the blue arrows.

In this paper, we propose a different task for self-supervision, which we call *Feature eXchange* (hence the name FX-GAN). In contrast to [3], our task requires the discriminator to learn to distinguish local structural inconsistencies in its inputs. Such inconsistencies are introduced into the input real images by swapping the feature vectors at a pair of random spatial locations in the images' intermediate convolutional feature maps. The network framework is illustrated in Figure 1. The CNN-based discriminator is split into two parts, denoted as D1 and D2 in Figure 1, in which D1 produces a spatial grid of channel-wise feature maps for real image inputs. We randomly choose two locations on this grid and *exchange* the features at the chosen locations. The resulting exchanged feature map is passed through the subsequent layers of the discriminator network (D2). Our modified discriminator is trained with two losses: (i) the standard adversarial loss that classifies its inputs as real or fake, and (ii) an auxiliary loss that classifies the inputs as being exchanged or not. Our intuition for the latter loss is that if the discriminator learns to classify exchanged feature maps on real images, then it will be better at distinguishing such local structural inconsistencies in the generated images. This improved discriminator can indirectly signal the generator to produce more structurally coherent content, i.e., more realistic synthesized images. Note that in contrast to the rotation-based task in [3], our proposed scheme captures global structural consistency in a bottom-up manner via distinguishing local structural coherency at arbitrary image locations within the generated images; our proposed task thus provides a stronger form of self-supervision.

Our key motivation for the proposed methodology comes from the observation [27, 34] that it is easier for GANs to learn to generate texture elements and small scale features than to generate objects with realistic structure (e.g., a dog's face must have the correct structural composition of eyes, nose, and mouth). We conjecture that this is due to the usual architectural choices for the discriminator, which is typically a CNN, and which is not readily amenable to learning feature representations that capture higher-order structural details in images. Recall that CNNs were originally designed for tasks that demand various forms of structural invariance (such as in object recognition, semantic segmentation, etc.); as a result, long-range higher-order information aggregation (as for image generation) can only occur gradually across many layers. We believe our proposed self-supervised task encourages a CNN-based discriminator to learn structural consistency more directly.

To summarize, our contributions are as follows: (i) we design a novel self-supervised task that creates *structurally inconsistent* feature maps from real input images, (ii) we propose a new discriminator architecture designed for this self-supervised task, which in addition to distinguishing its inputs as real vs. fake, should also classify whether or not they are structurally corrupted. We provide extensive experiments of our proposed multi-task GAN framework

on several datasets and achieve improved image generation results, as measured using both inception score and Fréchet inception distance (FID), against strong baselines used in [34]. Not only does our approach improve generation performance in the unconditional GAN setting targeted by a concurrently developed self-supervised GAN approach [3], but it also improves performance in class-conditional GANs. We believe our work is an important step towards exploring unsupervised learning of structured generative models.

2. Related work

Self-supervised learning defines a set of methods that learn useful feature representations from data by solving an indirect task for which labels can be easily and automatically generated. There is emerging interest in applying self-supervised learning to various tasks. Gidaris *et al*. [8] show that the auxiliary task of learning to identify image rotation can be beneficial for classification because the structure of objects matters for predicting rotation. Image colorization [35] and context encoder [25] attempt to recover image color and missing regions for feature learning. Agarwal *et al*. use motion cues [1], and Lee *et al*. use temporal ordering in video sequences [15] as sources of self-supervision. In ALI [7], a discriminative network is proposed that is trained to distinguish between joint latent or data-space samples from the generative network and joint samples from the inference network. For image-conditioned GANs, CycleGAN [37] introduces the cyclic loss that demands the generated image has a one-to-one correspondence with its input. This auxiliary task stabilizes GAN training in the unsupervised learning regime via avoiding the problem of mode-collapse.
GAN coupled with auxiliary tasks can be useful for both improved generation of complex images [30] and semi-supervised learning [26]. Conditional GANs are currently the most commonly used generative models capable of synthesizing complicated multi-class datasets, such as ImageNet. The AC-GAN was the first model to introduce an auxiliary classification loss for the discriminator [24]. More recently, the StarGAN, proposed by Choi *et al*. [4], applied AC-GAN to multi-domain image translation. FUNIT includes one real/fake head per class [17]. This architecture improves performance with fewer training samples from each class.

As alluded to above, Chen *et al*. [3] propose a self-supervised GAN model termed SS-GAN by rotating input images by either $0°, 90°, 180°,$ or $270°$ and introduce the auxiliary self-supervision task of rotation prediction. This task requires the model to learn something about the global structure of real images, which improves the performance of their unconditional GAN base model. However, they do not address the problem of using self-supervision to improve the performance of class-conditional GANs. In this paper, we show that our FX-GAN method improves image generation not only in the unconditional settings of SS-GAN, but also in the 1000-class conditional setting of ImageNet generation. Furthermore, our feature exchange task induces the network to learn both the long-range global structure and shorter-range local structure of images of objects.
Spatial structure of images for learning has been explored in several previous works. Villegas *et al*. [31] proposed generating structure to guide subsequent image generation in a supervised setting. The method of Zhu *et al*. [36] learns a discriminative model for the perception of realism in composite images. Relation networks [29] form explicit pairs of features for reasoning. Noroozi and Favaro [23] showed that the task of solving jigsaw puzzles leads to useful feature representations for classification. Liu *et al*. [18] showed that a CNN will not perform well in tasks related to spatial reasoning by default. Traditional convolutional GANs generate high-resolution details as a function of only spatially local points in lower-resolution feature maps. In the Self-Attention GAN (SAGAN) [34], however, details can be generated using cues from all feature locations. Moreover, their discriminator can verify if the highly detailed features in distant portions of the image are consistent with each other. Furthermore, recent work has shown that generator conditioning affects GAN performance. In [16], selective transfer units are incorporated with an encoder-decoder to adaptively select and modify encoder features for enhanced attribute editing. Inspired by these works, our model proposes the feature exchange (FX) auxiliary task to force the GAN to focus on both the local and global spatial structure of images. We are not aware of any previous works that use feature exchange as proposed for improving GAN training.

3. Method

In this section, we first review the standard GAN terminology and introduce our notation, after which we present the architecture of feature exchange GAN.

3.1. Generative Adversarial Networks

Let us first formally define the generator and discriminator networks and the basic optimization setup used to learn to generate realistic samples from random noise. The *generator* neural network, $\mathcal{G} : \mathbb{R}^v \rightarrow \mathbb{R}^d$, takes as input a v-dimensional random noise vector and outputs a d-dimensional image. The *discriminator* neural network, $\mathcal{D} : \mathbb{R}^d \rightarrow \mathbb{R}^1$, takes in an image and produces a score. (In contrast, FX-GAN's dual-task discriminator produces two scores, as we explain in Section 3.2.) The standard GAN optimization problem using a hinge-loss can be stated as:

$$P_{\mathcal{D}} := \min_{\mathcal{D}} \mathbb{E}_{\mathbf{z}\sim p(\mathbf{z})} \left[1+\mathcal{D}(\mathcal{G}(\mathbf{z}))\right]_+ + \mathbb{E}_{\mathbf{x}\sim p(\mathbf{x})} \left[1-\mathcal{D}(\mathbf{x})\right]_+ \tag{1}$$

$$P_{\mathcal{G}} := \max_{\mathcal{G}} \mathbb{E}_{\mathbf{z}\sim p(\mathbf{z})} [\mathcal{D}(\mathcal{G}(\mathbf{z})], \tag{2}$$

where the problems $P_{\mathcal{D}}$ and $P_{\mathcal{G}}$ are optimized over the parameters of the discriminator and generator neural networks respectively. Here $\mathbf{z}$ is sampled from a noise distribution, and $\mathbf{x}$ is sampled from the data distribution (a large data set). The notation $[\,]_+$ stands for the hinge loss, i.e., $[\,\cdot\,]_+ = \max(0, \cdot)$. In the setup defined above, the discriminator's target is to produce a score less than -1 on the fake (generated) images, while producing a score greater than 1 for real images. As for the generator, its parameters are optimized so as to make the discriminator produce higher scores for generated images. The use of a margin (the value 1 in Equations (1) and (2)) makes the training more robust, as proposed in [21]. The generator and discriminator solve two opposing requirements, and the optimization is a standard min-max game, although not a zero-sum one.

3.2. Feature Exchange GAN (FX-GAN)

Our main goal in this paper is to introduce self-supervisory signals into the GAN training that can speed up the convergence of the optimization and/or improve the quality of the generated images. The key self-supervisory signal we rely on is the *image context*; specifically, to empower the discriminator to recognize whether its input *real* images have been structurally corrupted (by exchanging patches of feature vectors in their feature space). Our hope is that having such a quality for the discriminator will implicitly influence the generator to produce *fake* samples that are devoid of such structural anomalies, leading to more coherent and structured content.

The Exchange Operator: An overview of our algorithm is provided in Figure 1. Formally, suppose $X \in \mathbb{R}^{h \times w \times d}$ denotes a feature map tensor, and let $X_{p,q} \in \mathbb{R}^d$ represent the feature vector at spatial location (p, q). Then, we define the *exchange* operator, $\xi(X; \pi)$, which inputs X and outputs a feature map tensor that is identical to X except that its feature vector at some spatial location (i, j) is exchanged with that at location $\pi(i, j)$, where π denotes a random permutation of the location index set $\mathcal{K}$. That is, for some $(i, j) \in \mathcal{K}$,

$$\xi(X;\pi)_{p,q} = \begin{cases} X_{i,j} & \text{if } (p,q) = \pi(i,j), \\ X_{\pi(i,j)} & \text{if } (p,q) = (i,j), \\ X_{p,q} & \text{otherwise.} \end{cases}$$

Intuitively, the exchange operator exchanges the spatial locations of some elements of the input feature map tensor.

In Section 4.4, we experiment with varying the parameters of feature exchange. For example, in addition to exchanging a pair of individual feature vectors (referred to as block size 1×1), we also experiment with exchanging a small block of feature vectors $\big($e.g., a $3 \times 3 \times d$ block centered at spatial location $(i, j)\big)$ with a block of the same size centered at spatial location $\pi(i, j)$. However, we found that exchanging a single pair of feature vectors (1×1 block) performs best. (We also experimented with exchanging feature vectors at more than one pair of locations, but did not observe an improvement.) This indicates that exchanging a large number of feature vectors is not useful, because that makes it too easy for the discriminator to recognize that an exchange has occurred.

To simplify our subsequent notation, we will slightly abuse the inputs to the exchange operator. For example, for $\mathcal{D}(\xi(\mathbf{x}))$, we mean that as the input $\mathbf{x}$ feeds forward through the discriminator neural network $\mathcal{D}$, the intermediate feature tensor (of dimension $h \times w \times p$) is extracted and *exchanged* (using a random permutation π) en route to producing the output of $\mathcal{D}$.

The idea of feature exchange can be combined with a variety of existing GAN algorithms and network architectures. To do so, we divide an existing algorithm's discriminator $\mathcal{D}$ into a sequence of two parts, $\mathcal{D}_1$ and $\mathcal{D}_2$. The input to $\mathcal{D}$ goes through $\mathcal{D}_1$ to produce a down-sampled feature tensor, on which a possible exchange operator may be applied if the input was a real image. The tensor is then passed through $\mathcal{D}_2$ to produce the discriminator output. We apply the exchange operator in a feature space of appropriate dimension with a suitable block size (see the discussion in Section 4.4) to avoid making the task too easy or too difficult for the discriminator. In contrast, exchanging pixel blocks in the raw pixel space would be prone to produce artifacts (such as discontinuous patch boundaries) that lead to inferior training (i.e., the discriminator may trivially identify the exchange via learning a representation for low-level texture rather than high-level semantic structure).

Dual-Task Discriminator: Our key insight is to strengthen the discriminator $\mathcal{D}$ to recognize corrupted (exchanged) images, in addition to identifying whether its inputs are real or fake. To this end, we modify the final linear layer of a discriminator $\mathcal{D}$ so that instead of one scalar output, it has *two*: (i) $\mathcal{D}_{\mathrm{r/f}}$, which has the standard discriminator goal of recognizing whether its inputs are real or fake, and (ii) $\mathcal{D}_{\mathrm{fx}}$, whose goal is to recognize whether or not its inputs have been feature-exchanged. The FX-GAN loss functions incorporate the following dual objectives:

$$\ell_{\mathrm{fx}}(\mathbf{x}) = [1 + \mathcal{D}_{\mathrm{fx}}(\mathbf{x})]_+ + [1 - \mathcal{D}_{\mathrm{fx}}(\xi(\mathbf{x}))]_+ , \tag{3}$$

$$\ell_{\mathrm{r/f}}(\mathbf{x}, \mathbf{z}) = \big[1 + \mathcal{D}_{\mathrm{r/f}}(\mathcal{G}(\mathbf{z}))\big]_+ + \big[1 - \mathcal{D}_{\mathrm{r/f}}(\mathbf{x})\big]_+ + \lambda \big[1 - \mathcal{D}_{\mathrm{r/f}}(\xi(\mathbf{x}))\big]_+ . \tag{4}$$

The loss defined in (3) identifies whether the real input images are feature-exchanged. Note that we only consider exchanging features of the real images, because corrupting the fake samples as they are being generated may not be fruitful (this is corroborated by our experiments). The loss $\ell_{\mathrm{r/f}}$ in (4) is an extension of the loss in (1), but it differs in that we also include a term for the feature-exchanged real images (which are considered *real*). We found that using a weighting λ for this new term is beneficial.

Letting $\mathcal{D}'$ refer to the new dual-task discriminator, we

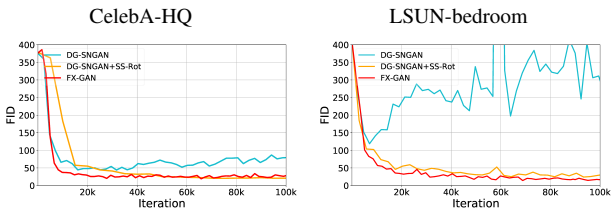


Figure 2. Training performance of unconditional GANs on two datasets. Both forms of self-supervision, FX (red) and SS-Rot (orange), make the training more stable and yield large improvements. On both datasets, our FX self-supervision task yields results comparable to or better than the SS-Rot self-supervision proposed in [3].

Dataset	Method	FID
CelebA-HQ	SS-GAN-Rot [3]	24.36
	DG-SNGAN	42.20
	DG-SNGAN + SS-Rot	20.50
	DG-SNGAN + FX (FX-GAN)	**19.25**
LSUN-bedroom	SS-GAN-Rot [3]	13.30
	DG-SNGAN	112.30
	DG-SNGAN + SS-Rot	24.21
	DG-SNGAN + FX (FX-GAN)	**12.90**
CIFAR10	SS-GAN-Rot [3]	**15.65**
	DG-SNGAN	26.33
	DG-SNGAN + SS-Rot	26.36
	DG-SNGAN + FX (FX-GAN)	24.63

Table 1. Comparison of unconditional GANs. Across all three datasets in the unconditional setting, FX-GAN improves upon the baseline model DG-SNGAN. (Results labeled as SS-GAN-Rot [3] are those reported in [3] for their best-performing unconditional model, SS-GAN(sBN)).

can state the complete FX-GAN learning problem as:

$$P'_{\mathcal{D}} := \min_{\mathcal{D}'} \mathbb{E}_{\mathbf{x}\sim p(\mathbf{x}),\mathbf{z}\sim p(\mathbf{z})}[\ell_{\mathrm{r/f}}(\mathbf{x},\mathbf{z}) + \gamma\,\ell_{\mathrm{fx}}(\mathbf{x})] \quad (5)$$

$$P'_{\mathcal{G}} := \max_{\mathcal{G}} \mathbb{E}_{\mathbf{z}\sim p(\mathbf{z})}[\mathcal{D}_{\mathrm{r/f}}(\mathcal{G}(\mathbf{z}))], \quad (6)$$

where γ is another constant weight. The objective $P'_{\mathcal{G}}$ for the generator remains the same as $P_{\mathcal{G}}$ provided in (2).

4. Experiments and Results

In this section, we evaluate the empirical benefits of FX-GAN on several standard datasets. After reviewing the settings in Section 4.1, we show the overall improvement with FX-GAN on unconditional and conditional cases in Sections 4.2 and 4.3, respectively. Then we discuss other aspects of FX-GAN in Section 4.4, including the feature-exchange parameters and impact on training speed.

4.1. Setup of Experiments

To evaluate our approach, we extend the publicly available TensorFlow-based implementation of Self-Attention GAN (SAGAN) [34] to incorporate feature-exchange functionality. This implementation provides two strong baseline GAN architectures, namely: i) the SAGAN itself, which learns a self-attention layer inside the generator and the discriminator that produces an attention map for each spatial location in the feature tensor, and ii) the same architecture as SAGAN, but without the self-attention layer. Because this architecture applies spectral normalization (SN) [21] to both the discriminator (D) and generator (G), we denote it as DG-SNGAN.

Implementation Details: For all experiments, we used Adam as the optimizer [13] with parameters $\beta_1 = 0$ and $\beta_2 = 0.9$ [21], and used Two Time-scale Update Rule (TTUR) [10] to stabilize the training. For our FX algorithm, we set the weights of our feature-exchange terms in the loss functions (4) and (5) as $\lambda = \gamma = 0.1$. The exchanged feature block size is 1×1 except in Sec 4.4. In all experiments, only one pair of feature blocks are exchanged.

4.2. FX with unconditional GANs

We first verify the effectiveness of our approach on unconditional image generation. We use the the same datasets used by Chen *et al.* [3], including CelebA-HQ [12], CIFAR10 [14], and the bedroom category of the LSUN dataset [33]. To measure the performance of unconditional GANs, we use Fréchet Inception Distance (FID) [10], where smaller FID is better. Here we only used DG-SNGAN as the baseline model. (As both SAGAN and FX can change the feature maps of convolution layers, the combination will introduce extra parameters to test.) We do evaluate SAGAN with FX below for conditional GANs. Hereafter, we use FX-GAN to denote the combination of DG-SNGAN + FX.

Implementation Details: For both the LSUN-bedroom and CelebA-HQ datasets, we resized the images to $128\times128\times3$ as in [3]. Our architecture, which is the same as SAGAN, is described in Table 1 of the supplementary material. We used 256 as the batch size, and used learning rates of 0.00005 for the generator and 0.0001 for the discriminator. We apply FX on the feature maps of size 32×32 pixels. For CIFAR10, since its image size is only 32×32 pixels, we changed the architecture, as described in Table 2 of the supplementary material. We also changed the batch size to 64, and set the generator and discriminator learning rates to 0.0004 and 0.0001, respectively.

LSUN-bedroom and CelebA-HQ Results: Figure 2 shows FID curves from training unconditional GANs on the CelebA-HQ and LSUN-bedroom datasets. The top two sections of Table 1 show the FID of the best performing network for each dataset and model combination. In the first row of each section (labeled SS-GAN-Rot [3]), we present the results reported in [3] for their own implementation of their best-performing unconditional model, SS-GAN(sBN).

DG-SNGAN + SS-Rot FX-GAN Real Images

Figure 3. Generated images of GANs trained on the LSUN-bedroom dataset. FX-GAN overall produces more realistic scene layouts and local details.

For a fairer comparison, we also implement their rotational self-supervision on top of our baseline DG-SNGAN; we call the combination DG-SNGAN + SS-Rot. Our proposed model is labeled FX-GAN (the DG-SNGAN baseline + FX self-supervision).

The unconditional version of our baseline, DG-SNGAN, is unstable on both the LSUN-bedroom and CelebA-HQ datasets, and the training often diverges. For both forms of self-supervision, our FX-GAN as well as DG-SNGAN + SS-Rot, stabilize the training and significantly outperform the baseline model. On the CelebA-HQ dataset, FX-GAN just slightly outperforms the other self-supervised model.

On LSUN-Bedroom, our proposed method achieves a much better FID score than the rotational self-supervision DG-SNGAN + SS-Rot. The improvement may be because our feature exchange introduces more subtle changes to the original image or feature than the entire-image rotation of SS-Rot. Feature exchange forces the discriminator in FX-GAN to pay more attention to the consistency of neighboring features. Figure 3 demonstrates example generated images of both self-supervised GAN methods trained on the LSUN-bedroom dataset. Overall, FX-GAN produces more realistic scene layouts and local details.

CIFAR10 Results: The bottom section of Table 1 shows the best FID obtained when training each unconditional GAN model on the CIFAR10 [14] dataset without using the class labels. Adding SS-Rotation self-supervision to the baseline DG-SNGAN model does not improve the baseline's performance. In contrast, our FX-GAN model's self-supervision does improve over the baseline, reducing the FID from 26.33 to 24.63.

A final note about the SS-Rot self-supervision. As implemented by [3], the SS-Rot self-supervision improved generation performance on two datasets (LSUN-bedroom and CIFAR10), but actually degraded performance on the third dataset (CelebA-HQ). In our implementation, SS-Rot exhibits a similarly uneven performance, improving results on (a different) two of the datasets (CelebA-HQ and LSUN-bedroom) and slightly degrading results on the third dataset (CIFAR-10). In contrast, our proposed FX self-supervision consistently improves upon the performance of the baseline method on all datasets tested.

Base Method	Self-Supervision	Inception Score	FID
DG-SNGAN	None	47.6	21.3
	FX (FX-GAN)	**51.0**	**19.5**
SAGAN	None	49.0	19.7
	FX	**49.8**	**18.9**

Table 2. Performance of class-conditional GAN generators trained on ImageNet.

4.3. FX with Conditional GANs on ImageNet

We use ImageNet [5] to evaluate the benefits of the proposed FX self-supervision on conditional GAN training. In addition to measuring the FID, here we also measure the inception score [27], which is commonly used to compare image generation performance on this dataset.

Implementation Details: We resize and crop the images to 128×128 pixels as in [21], and use the same architecture as for LSUN-bedroom and CelebA-HQ. We use 256 as the batch size, 0.0001 as the learning rate for the generator, and 0.0004 as the learning rate for the discriminator. We tested FX by exchanging a pair of 1×1 blocks in the feature maps of 32×32 spatial dimension. For the model SAGAN, we apply self-attention to the feature tensor of spatial dimension 32×32, which is the best model reported by [34]. As both FX and self-attention access the same feature maps, the order of FX and self-attention matters. We apply FX before self-attention, because we found that training can be unstable when using the reverse order.

DG-SNGAN FX-GAN

Figure 4. The generated images by DG-SNGAN and FX-GAN trained on ImageNet. FX-GAN generates images that are both more diverse and more realistic.

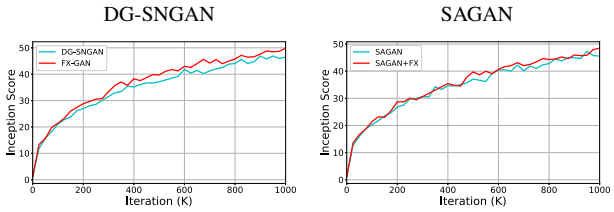


Figure 5. Smoothed Inception score curves of the generators trained without (blue) and with FX (red) on ImageNet. Feature exchanged models lead to higher Inception score overall.

ImageNet Results: Figure 5 plots the inception scores throughout 1 million iterations of training for each of the baseline models, DG-SNGAN and SAGAN, both without (blue) and with (red) FX self-supervision. For both baseline models, the inception scores with FX are better (higher) than without FX. Table 2 shows the maximum inception scores and minimum FIDs achieved by each model during the 1 million iterations of training. The table shows that FX self-supervision boosts the performance of both baselines, but especially improves upon DG-SNGAN. One reason for explaining FX provides less improvement on SAGAN may be that both FX and self-attention can change the feature maps within the discriminator, and thus each might interfere with the benefits of the other. Exploring their synergy will be our future work.

Qualitative Results: Figure 4 shows example images of four classes (monarch butterfly, crib, library, ptarmigan) generated by the baseline DG-SNGAN and our proposed FX-GAN model. Notice the improved diversity and realism of the FX-GAN images. Figure 6 shows examples of interpolated samples between generated images from our class-conditional FX-GAN. Each row of images corresponds to a particular class. The ends of each row are images generated from different random noise vectors, while the intermediate images are generated from vectors whose values were interpolated linearly between the two noise vectors.

4.4. Discussion

Hyperparameters: Our feature exchange algorithm has two hyperparameters: the scale of feature map at which the

Figure 6. Interpolation examples of FX-GAN generated images (one class per row). The images on the left and right ends of each row are generated from random noise vectors. The intermediate images in each row are generated from vector values that were linearly interpolated between the two end vectors.

Block size	1×1	2×2	3×3
FID	**24.63**	26.33	25.90

Table 3. Effect of varying the block size to exchange within a 16x16 feature map. For each block size, the table shows the lowest (best) FID obtained in 100,000 iterations of training our unconditional FX-GAN on CIFAR10.

Feature map size	4×4	8×8	16×16
FID	37.22	25.96	**24.63**

Table 4. Effect of varying the feature map used for exchange. For each feature map size, the table shows the lowest (best) FID obtained in 100,000 iterations of training our unconditional FX-GAN with block size 1×1 on CIFAR10.

exchange is performed, and the size of the blocks of feature vectors that are exchanged within that feature map. To evaluate the effects of these hyperparameters, we trained an unconditional FX-GAN on CIFAR10 with batch size 64 and measured the minimal FID within 100,000 training steps. Table 3 shows the results when the feature exchange operator is applied with different block sizes in the 16 feature map. We can see that FID increases when the block size is larger than 1. Table 4 shows the results of applying the operator with 1×1 block size in different feature maps, where 16×16 is the largest (highest-resolution) feature map. It shows that when the feature map size is reduced, FID increases. As each pixel in a larger feature map has a smaller receptive field in the original image space, it potentially makes the task of identifying structure inconsistency more difficult. The results of both Tables 3 and 4 suggest that a smaller feature block in a larger feature map leads to more apparent benefit of FX self-supervision. This finding suggests that the performance of the generator is affected by the difficulty of training the "Exchanged?" task (see Figure 1). When the block size is large, it is easy to tell whether or not the feature map is being exchanged. In this case, the training loss is mainly related to the "Real?" task, and the benefit of FX vanishes. However, when the exchange operator is applied directly to the real images, it creates artifacts in images and leads to inferior results empirically.

Speed: Although FX introduces an extra stage, the additional overhead is relatively minor. In our experiments on 4 Titan XP GPUs, the training time per iteration was increased by about 20%, from 1.86 seconds to 2.22 seconds, when FX was enabled. The output of D1 for real images is only computed once for the second and third rows of Figure 1. The additional computation comes from passing the exchanged feature maps through the layers in D2.

5. Conclusion

We present Feature Exchange GAN (FX-GAN), a self-supervised framework for improving GAN learning performance. We extend the discriminator to indicate not only whether the input image is real, but also whether there is structural inconsistency. Based on our feature exchange operator, a new loss function is proposed to train the multi-task discriminator, which leads to a regularized feature representation for the discriminator and hence a better generator. Experimental results show that when combined with two different strong GAN baselines, our feature-exchange self-supervision can achieve improved generated image quality and diversity on several datasets including ImageNet. FX-GAN yields significant improvements in both image generation and training stability, in unconditional and class-conditional GAN settings. Our analysis of different parameter settings of FX-GAN indicates a correlation between the difficulty of identifying the structural inconsistency and the improvement in generated image quality.

References

[1] P. Agrawal, J. Carreira, and J. Malik. Learning to see by moving. In *ICCV*, 2015.

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

[3] T. Chen, X. Zhai, M. Ritter, M. Lucic, and N. Houlsby. Self-supervised gans via auxiliary rotation loss. In *CVPR*, 2019.

[4] Y. Choi, M. Choi, M. Kim, J.-W. Ha, S. Kim, and J. Choo. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *CVPR*, 2018.

[5] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR*, 2009.

[6] C. Doersch, A. Gupta, and A. A. Efros. Unsupervised visual representation learning by context prediction. In *CVPR*, 2015.

[7] V. Dumoulin, I. Belghazi, B. Poole, O. Mastropietro, A. Lamb, M. Arjovsky, and A. Courville. Adversarially learned inference. *ICLR*, 2017.

[8] S. Gidaris, P. Singh, and N. Komodakis. Unsupervised representation learning by predicting image rotations. In *ICLR*, 2018.

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

[10] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *NIPS*, 2017.

[11] R. Huang, S. Zhang, T. Li, and R. He. Beyond face rotation: Global and local perception gan for photorealistic and identity preserving frontal view synthesis. In *ICCV*, 2017.

[12] T. Karras, T. Aila, S. Laine, and J. Lehtinen. Progressive growing of GANs for improved quality, stability, and variation. In *International Conference on Learning Representations*, 2018.

[13] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *ICLR*, 2014.

[14] A. Krizhevsky, G. Hinton, et al. Learning multiple layers of features from tiny images. Technical report, Citeseer, 2009.

[15] H.-Y. Lee, J.-B. Huang, M. Singh, and M.-H. Yang. Unsupervised representation learning by sorting sequences. In *ICCV*, 2017.

[16] M. Liu, Y. Ding, M. Xia, X. Liu, E. Ding, W. Zuo, and S. Wen. Stgan: A unified selective transfer network for arbitrary image attribute editing. In *CVPR*, 2019.

[17] M.-Y. Liu, X. Huang, A. Mallya, T. Karras, T. Aila, J. Lehtinen, and J. Kautz. Few-shot unsupervised image-to-image translation. *ICCV*, 2019.

[18] R. Liu, J. Lehman, P. Molino, F. P. Such, E. Frank, A. Sergeev, and J. Yosinski. An intriguing failing of convolutional neural networks and the coordconv solution. In *NeurIPS*, 2018.

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

[20] I. Misra, C. L. Zitnick, and M. Hebert. Shuffle and learn: unsupervised learning using temporal order verification. In *ECCV*, 2016.

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

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

[23] M. Noroozi and P. Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *ECCV*, 2016.

[24] A. Odena, C. Olah, and J. Shlens. Conditional image synthesis with auxiliary classifier gans. In *ICML*, 2017.

[25] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros. Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2536–2544, 2016.

[26] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen. Improved techniques for training gans. In *Advances in neural information processing systems*, pages 2234–2242, 2016.

[27] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, X. Chen, and X. Chen. Improved techniques for training gans. In *NIPS*, 2016.

[28] R. Santa Cruz, B. Fernando, A. Cherian, and S. Gould. Deeppermnet: Visual permutation learning. In *CVPR*, 2017.

[29] A. Santoro, D. Raposo, D. G. Barrett, M. Malinowski, R. Pascanu, P. Battaglia, and T. Lillicrap. A simple neural network module for relational reasoning. In *NIPS*, 2017.

[30] J. T. Springenberg. Unsupervised and semi-supervised learning with categorical generative adversarial networks. *arXiv preprint arXiv:1511.06390*, 2015.

[31] R. Villegas, J. Yang, Y. Zou, S. Sohn, X. Lin, and H. Lee. Learning to generate long-term future via hierarchical prediction. In *ICML*, 2017.

[32] X. Wang and A. Gupta. Generative image modeling using style and structure adversarial networks. In *ECCV*. Springer, 2016.

[33] F. Yu, A. Seff, Y. Zhang, S. Song, T. Funkhouser, and J. Xiao. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. *arXiv preprint arXiv:1506.03365*, 2015.

[34] H. Zhang, I. Goodfellow, D. Metaxas, and A. Odena. Self-attention generative adversarial networks. *ICML*, 2019.

[35] R. Zhang, P. Isola, and A. A. Efros. Colorful image colorization. In *ECCV*, 2016.

[36] J.-Y. Zhu, P. Krahenbuhl, E. Shechtman, and A. A. Efros. Learning a discriminative model for the perception of realism in composite images. In *ICCV*, 2015.

[37] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *ICCV*, 2017.

FX-GAN: Self-Supervised GAN Learning via Feature Exchange Supplementary Material

Rui Huang[†]
Carnegie Mellon University
ruih2@alumni.cmu.edu

Wenju Xu[†]
University of Kansas
xuwenju@ku.edu

Teng-Yok Lee Anoop Cherian Ye Wang Tim K. Marks
Mitsubishi Electric Research Laboratories (MERL)
{tlee,cherian,yewang,tmarks}@merl.com

1. Error analysis

Figure 1 shows the errors of the *Exchanged?* task prediction (see Fig. 1 of the paper) when training on ImageNet from iterations 300,000 to 900,000. Note that with self-attention (SAGAN + FX), the error for task reduced to zero more quickly than without self-attention (FX-GAN). This is expected, because the goal of self-attention is to learn to attend to regions that are semantically closely related. Because of this, the inconsistency caused by feature exchange is easier for the discriminator to distinguish, so the proposed feature-exchange loss, ℓ_{fx}, will not be as effective at regularizing the discriminator's representation. For FX-GAN, the error decreases much more slowly, which makes the regularization from ℓ_{fx} more effective and leads to larger improvements in the results. In future work, we could adaptively adjust the difficulty for learning the *Exchanged?* task.

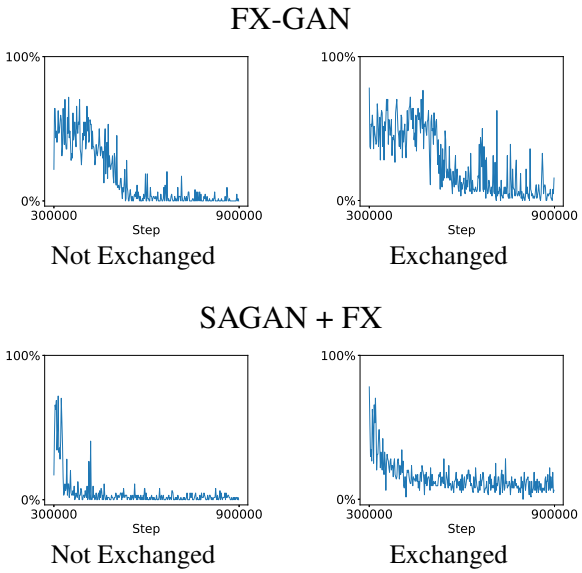


Figure 1. Errors made by the *Exchanged?* task prediction(percentage of images misclassified per batch) when training on ImageNet from iterations 300,000 to 900,000. Errors in the left column are images whose features were *not* exchanged but were misclassified as exchanged. Errors in the right column are images whose features were exchanged but were misclassified as *not* exchanged. *Top row:* FX-GAN. *Bottom row:* SAGAN + FX.

2. Network architecture

For datasets ImageNet, CelebA-HQ, and LSUN bedroom, our network architecture is the same as SAGAN [1]. In the discriminator, each image is first resized to 128×128 pixels, then passed through a sequence of residual blocks. Each residual block downsamples each spatial dimension by 2 and expands the number of channels. Table 1(a) describes the discriminator network architecture by giving the size of the tensor in the spatial and channel dimensions, at the input to the network and after each residual block. For example, the input to the discriminator is a 128×128-pixel image with 3 channels. For the generator, the input noise is first converted into a tensor of $4 \times 4 \times 1024$ elements, then passed through a sequence of deconvolution filters to increase the spatial size and reduce the number of channels. Table 1(b) lists the size of the tensor after each deconvolution. For CIFAR10, since the input size is smaller (32×32), we adjust the network architecture to have fewer residual blocks and fewer deconvolution layers, as described in Table 2.

3. Qualitative results from FX-GAN versus DG-SNGAN

In this supplementary material, we present example images that we generated using the two models that are eval-

[†]Work done while interning at MERL.

Dimension	Input size	Size after each residual block						
x, y	128	64	32	16	8	4	2	2
channels	3	64	128	256	512	1024	2048	2048

(a) Discriminator

Dimension	Input size	Size after each deconvolution					
x, y	4	8	16	32	64	128	128
channels	1024	1024	512	256	128	64	3

(b) Generator

Table 1. The network architecture for LSUN-bedroom, CelebA-HQ, and ImageNet. The numbers represent the tensor shapes after the residual blocks of the discriminator (a) and after the deconvolution blocks of the generator (b).

Dimension	Input size	Size after residual blocks			
x, y	32	32	16	8	4
channels	3	64	128	256	512

(a) Discriminator

Dimension	Input size	Size after deconvolutions			
x, y	4	8	16	32	32
channels	256	256	128	64	3

(b) Generator

Table 2. The network architecture for CIFAR10.

uated in the top section of Table 2 of the paper. The first model is the baseline model, DG-SNGAN. The second is our proposed FX-GAN model (a.k.a. DG-SNGAN + FX). Both models were trained for 1,000,000 iterations on ImageNet (1,000 classes) to perform class-conditional generation of 128×128-pixel images.

3.1. Images generated by our FX-GAN model

Figures 2, 3, and 4 show examples of class-conditional image generation by our proposed model, FX-GAN. Each figure shows 64 generated examples of one class. Each of the 64 images was generated using a different random noise vector.

3.2. Interpolated images generated by FX-GAN

In Figure 5, we show example interpolations of class-conditional images generated by FX-GAN. Each row of images contains a separate interpolation corresponding to a particular class. The ends of each row are images generated from different random noise vectors, while the intermediate images are generated from vectors whose values were interpolated linearly between the two noise vectors.

3.3. Qualitative comparison to DG-SNGAN

We qualitatively compare the class-conditional image generation performance of our FX-GAN model vs. the baseline DG-SNGAN model in Figures 6–12. These examples demonstrate subjective improvements in structural consistency, detail, and/or image diversity for FX-GAN. Interestingly, for some classes, as seen in Figures 10, 11, and 12, the DG-SNGAN baseline seems to exhibit some form of mode collapse (reduction), where greatly reduced image diversity is observed. Across all of the classes, we generally observed that FX-GAN was far more resistant to this type of mode collapse.

References

[1] H. Zhang, I. Goodfellow, D. Metaxas, and A. Odena. Self-attention generative adversarial networks. *ICML*, 2019.

Figure 2. FX-GAN generated examples for ImageNet class 15, “robin.”

Figure 3. FX-GAN generated examples for ImageNet class 914, “yawl.”

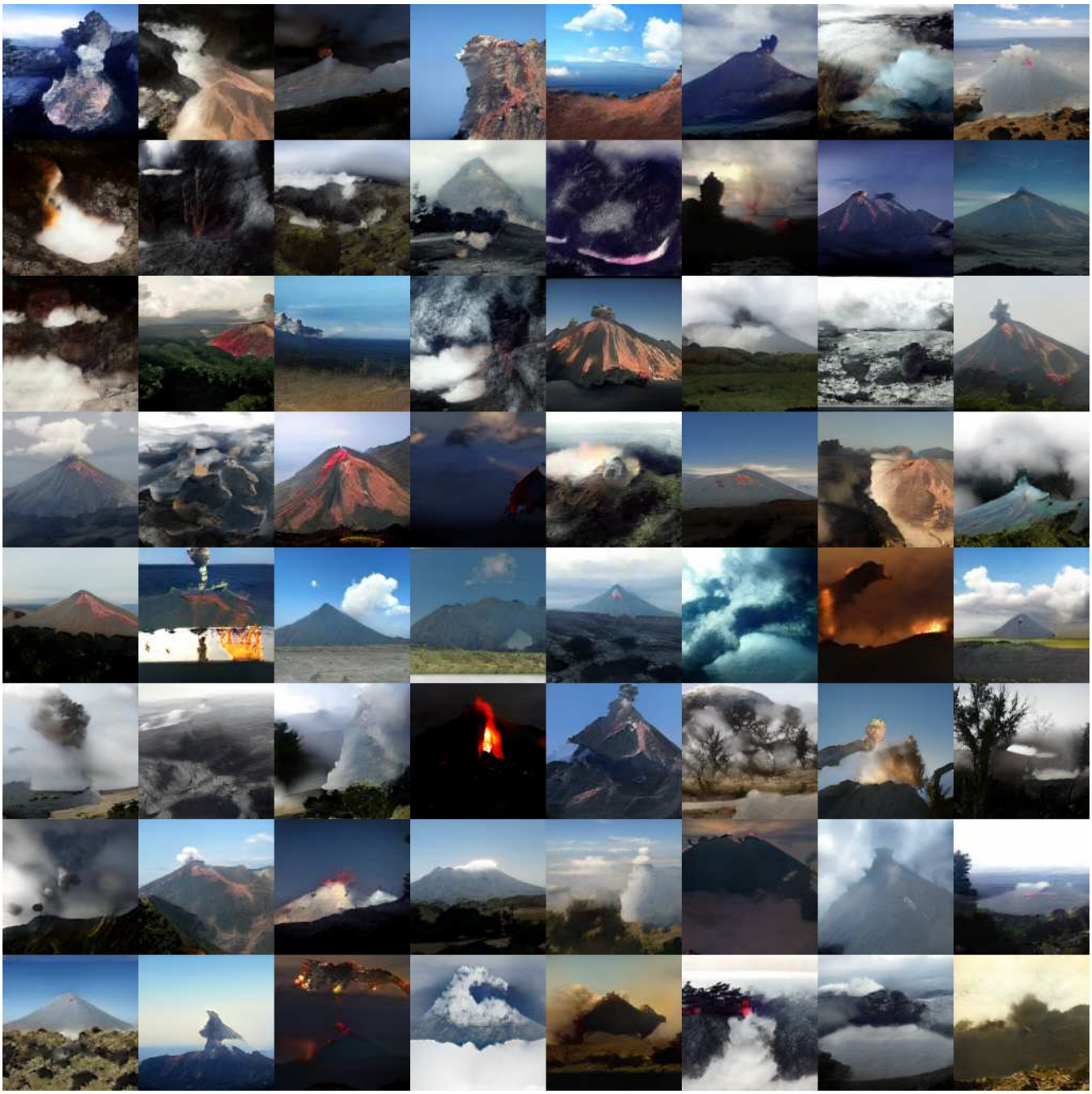


Figure 4. FX-GAN generated examples for ImageNet class 980, “volcano.”

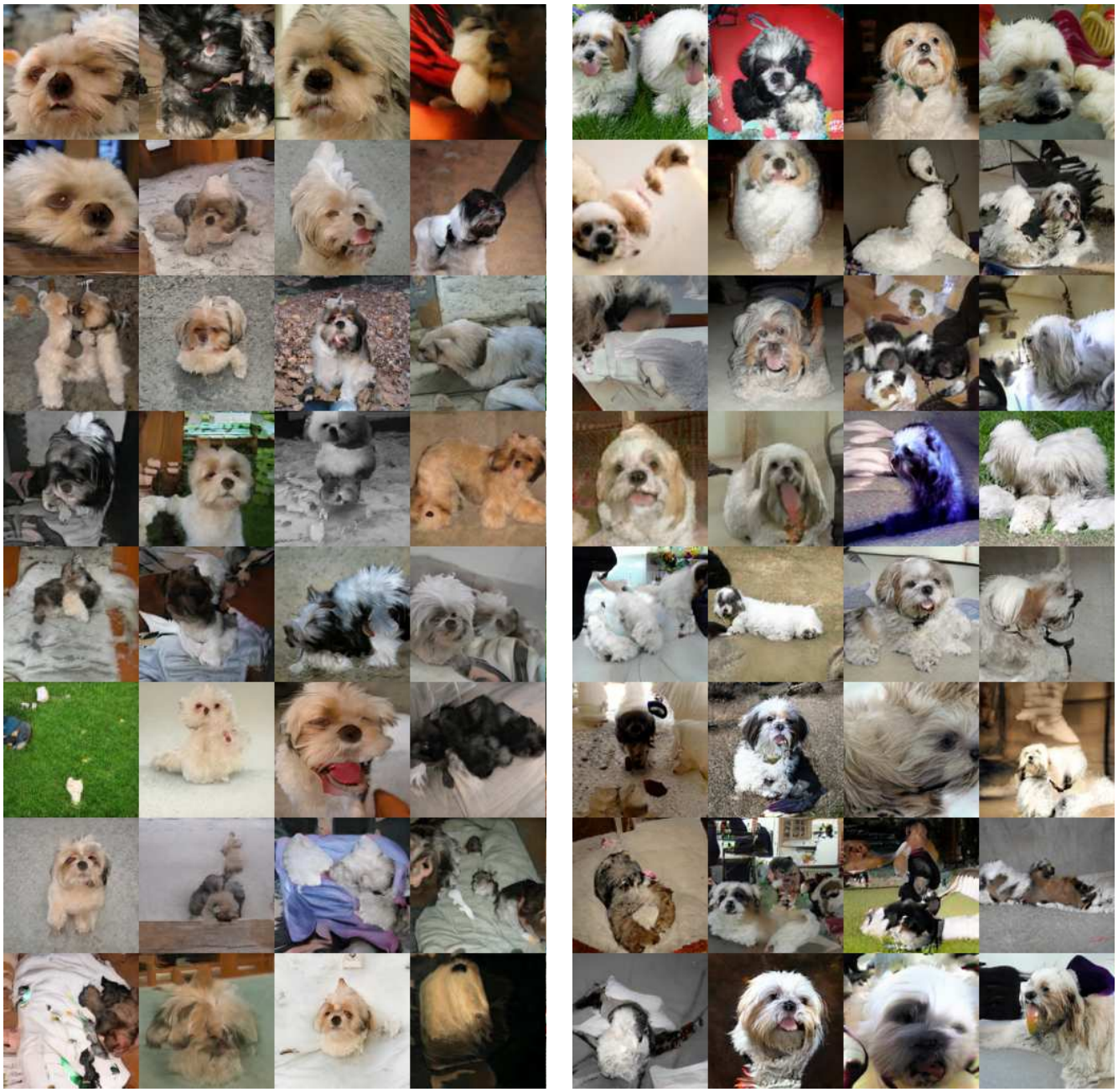
Figure 5. FX-GAN generated image interpolation examples, one class per row. The images on the left and right ends of each row are generated from random noise vectors. The intermediate images in each row are generated from vector values that were linearly interpolated between the two end vectors.

DG-SNGAN FX-GAN

Figure 6. Comparison of DG-SNGAN vs. FX-GAN generated examples for ImageNet class 24, "great grey owl." Note that FX-GAN has learned to generate more realistic eyes than the baseline method.

DG-SNGAN FX-GAN

Figure 7. Comparison of DG-SNGAN vs. FX-GAN generated examples for ImageNet class 81, "ptarmigan." Note that FX-GAN has learned to generate more realistic body shapes than the baseline method.



DG-SNGAN FX-GAN

Figure 8. Comparison of DG-SNGAN vs. FX-GAN generated examples for ImageNet class 155, "Shih-Tzu." Note that FX-GAN has learned to generate more realistic facial arrangements than the baseline method.

DG-SNGAN FX-GAN

Figure 9. Comparison of DG-SNGAN vs. FX-GAN generated examples for ImageNet class 574, "golf ball." Note that FX-GAN has learned to generate more realistic golf ball colors and textures than the baseline method.

DG-SNGAN FX-GAN

Figure 10. Comparison of DG-SNGAN vs. FX-GAN generated examples for ImageNet class 323, "monarch butterfly." Note that FX-GAN has learned to generate better details and color variations than the baseline method.

DG-SNGAN FX-GAN

Figure 11. Comparison of DG-SNGAN vs. FX-GAN generated examples for ImageNet class 520, "crib." Note that FX-GAN creates a much greater variety of crib styles, textures, and colors.

DG-SNGAN FX-GAN

Figure 12. Comparison of DG-SNGAN vs. FX-GAN generated examples for ImageNet class 624, “library.” Note that FX-GAN creates a much greater variety of bookshelf styles, textures, and colors.