Lifelong GAN: Continual Learning for Conditional Image Generation

Mengyao Zhai[1,2*], Lei Chen[1,2*], Fred Tung[1,2], Jiawei He[1,2], Megha Nawhal[1,2], Greg Mori[1,2]
[1]Simon Fraser University [2]Borealis AI
{mzhai, chenleic, ftung, jha203, mnawhal}@sfu.ca mori@cs.sfu.ca

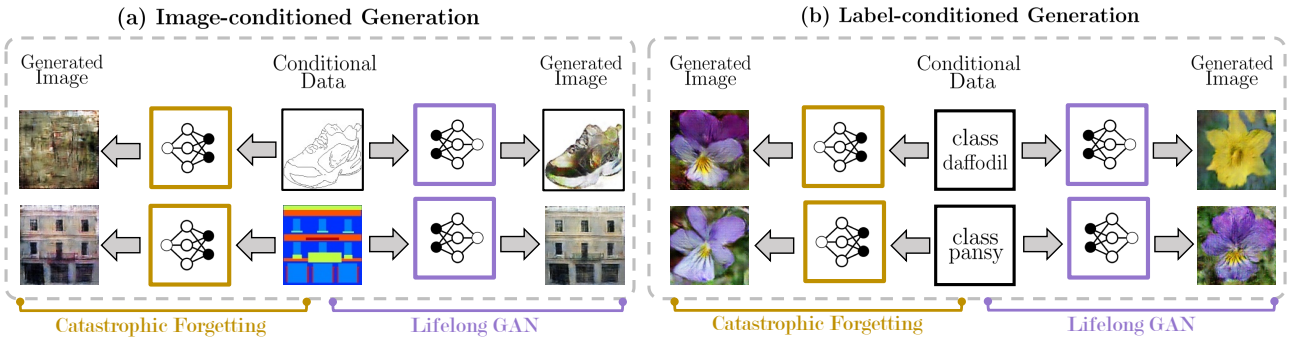


Figure 1: **Lifelong learning of conditional image generation.** Traditional training methods suffer from catastrophic forgetting: when we add new tasks, the network forgets how to perform previous tasks. Our Lifelong GAN is a generic framework for conditional image generation that applies to various types of conditional inputs (*e.g.* labels and images).

Abstract

Lifelong learning is challenging for deep neural networks due to their susceptibility to catastrophic forgetting. Catastrophic forgetting occurs when a trained network is not able to maintain its ability to accomplish previously learned tasks when it is trained to perform new tasks. We study the problem of lifelong learning for generative models, extending a trained network to new conditional generation tasks without forgetting previous tasks, while assuming access to the training data for the current task only. In contrast to state-of-the-art memory replay based approaches which are limited to label-conditioned image generation tasks, a more generic framework for continual learning of generative models under different conditional image generation settings is proposed in this paper. Lifelong GAN employs knowledge distillation to transfer learned knowledge from previous networks to the new network. This makes it possible to perform image-conditioned generation tasks in a lifelong learning setting. We validate Lifelong GAN for both image-conditioned and label-conditioned generation tasks, and provide qualitative and quantitative results to show the generality and effectiveness of our method.

1. Introduction

Learning is a lifelong process for humans. We acquire knowledge throughout our lives so that we become more efficient and versatile facing new tasks. The accumulation of knowledge in turn accelerates our acquisition of new skills. In contrast to human learning, lifelong learning remains an open challenge for modern deep learning systems. It is well known that deep neural networks are susceptible to a phenomenon known as *catastrophic forgetting* [18]. Catastrophic forgetting occurs when a trained neural network is not able to maintain its ability to accomplish previously learned tasks when it is adapted to perform new tasks.

Consider the example in Figure 1. A generative model is first trained on the task edges $\rightarrow$ shoes. Given a new task segmentations $\rightarrow$ facades, a new model is initialized from the previous one and fine-tuned for the new task. After training, the model forgets about the previous task and cannot generate shoe images given edge images as inputs. One way to address this would be to combine the training data for the current task with the training data for all previous tasks and then train the model using the joint data. Unfortunately, this approach is not scalable in general: as

*Equal Contribution

new tasks are added, the storage requirements and training time of the joint data grow without bound. In addition, the models for previous tasks may be trained using private or privileged data which is not accessible during the training of the current task. The challenge in lifelong learning is therefore to extend the model to accomplish the current task, without forgetting how to accomplish previous tasks in scenarios where we are restricted to the training data for only the current task. In this work, we work under the assumption that we only have access to a model trained on previous tasks without access to the previous data.

Recent efforts [24, 3, 7] have demonstrated how discriminative models could be incrementally learnt for a sequence of tasks. Despite the success of these efforts, lifelong learning in generative settings remains an open problem. Parameter regularization [23, 13] has been adapted from discriminative models to generative models, but poor performance is observed [28]. The state-of-the-art continual learning generative frameworks [23, 28] are built on *memory replay* which treats generated data from previous tasks as part of the training examples in the new tasks. Although memory replay has been shown to alleviate the catastrophic forgetting problem by taking advantage of the generative setting, its applicability is limited to label-conditioned generation tasks. In particular, replay based methods cannot be extended to image-conditioned generation. The reason lies in that no conditional image can be accessed to generate replay training pairs for previous tasks. Therefore, a more generic continual learning framework that can enable various conditional generation tasks is valuable.

In this paper, we introduce a generic continual learning framework *Lifelong GAN* that can be applied to both image-conditioned and label-conditioned image generation. We employ *knowledge distillation* [9] to address catastrophic forgetting for conditional generative continual learning tasks. Given a new task, Lifelong GAN learns to perform this task, and to keep the memory of previous tasks, information is extracted from a previously trained network and distilled to the new network during training by encouraging the two networks to produce similar output values or visual patterns. To the best of our knowledge, we are the first to utilize the principle of knowledge distillation for continual learning generative frameworks.

To summarize, our contributions are as follows. *First*, we propose a generic framework for continual learning of conditional image generation models. *Second*, we validate the effectiveness of our approach for two different types of conditional inputs: (1) image-conditioned generation, and (2) label-conditioned generation, and provide qualitative and quantitative results to illustrate the capability of our GAN framework to learn new generation tasks without the catastrophic forgetting of previous tasks. *Third*, we illustrate the generality of our framework by performing continual learning across diverse data domains.

2. Related Work

Conditional GANs. Image generation has achieved great success since the introduction of GANs [8]. There also has been rapid progress in the field of conditional image generation [19]. Conditional image generation tasks can be typically categorized as image-conditioned image generation and label-conditioned image generation.

Recent image-conditioned models have shown promising results for numerous image-to-image translation tasks such as maps $\rightarrow$ satellite images, sketches $\rightarrow$ photos, labels $\rightarrow$ images [10, 35, 34], future frame prediction [26], superresolution [15], and inpainting [30]. Moreover, images can be stylized by disentangling the style and the content [11, 16] or by encoding styles into a stylebank (set of convolution filters) [4]. Models [32, 17] for rendering a person's appearance onto a given pose have shown to be effective for person re-identification. Label-conditioned models [5, 6] have also been explored for generating images for specific categories.

Knowledge Distillation. Proposed by Hinton et al. [9], knowledge distillation is designed for transferring knowledge from a teacher classifier to a student classifier. The teacher classifier normally would have more privileged information [25] compared with the student classifier. The privileged information includes two aspects. The first aspect is referred to as the learning power, namely the size of the neural networks. A student classifier could have a more compact network structure compared with the teacher classifier, and by distilling knowledge from the teacher classifier to student classifier, the student classifier would have similar or even better classification performance than the teacher network. Relevant applications include network compression [21] and network training acceleration [27]. The second aspect is the learning resources, namely the amount of input data. The teacher classifier could have more learning resources and see more data that the student cannot see. Compared with the first aspect, this aspect is relatively unexplored and is the focus of our work.

Continual Learning. Many techniques have been recently proposed for solving continuous learning problems in computer vision [24, 3] and robotics [7] in both discriminative and generative settings.

For discriminative settings, Shmelkov *et al*. [24] employ a distillation loss that measures the discrepancy between the output of the old and new network for distilling knowledge learnt by the old network. In addition, Castro *et al*. [3] propose to use a few exemplar images from previous tasks and perform knowledge distillation using new features from previous classification layers followed by a modified activation layer. For generative settings, continual learning has

been primarily achieved using memory replay based methods. Replay was first proposed by Seff et al. [23], where the images for previous tasks are generated and combined together with the data for the new task to form a joint dataset, and a new model is trained on the joint dataset. A similar idea is also adopted by Wu et al. [28] for label-conditioned image generation. Approaches based on elastic weight consolidation [13] have also been explored for the task of label-conditioned image generation [28], but they have limited capability to remember previous categories and generate high quality images.

In this paper, we introduce knowledge distillation within continual generative model learning, which has not been explored before. Our approach can be applied to both image-conditioned generation, for which the replay mechanism is not applicable, and label-conditioned image generation.

3. Approach

Our proposed Lifelong GAN addresses catastrophic forgetting using knowledge distillation and, in contrast to replay based methods, can be applied to continually learn both label-conditioned and image-conditioned generation tasks. In this paper, we build our model on the state-of-the-art BicycleGAN [35] model. Our overall approach for continual learning for a generative model is illustrated in Figure 2. Given data from the current task, Lifelong GAN learns to perform this task, and to keep the memory of previous tasks, knowledge distillation is adopted to distill information from a previously trained network to the current network by encouraging the two networks to produce similar output values or patterns given the same input. To avoid "conflicts" that arise when having two desired outputs (current training goal and outputs from previous model) given the same input, we generate auxiliary data for distillation from the current data via two operations *Montage* and *Swap*.

3.1. Background: BicycleGAN

We first introduce the state-of-the-art BicycleGAN [35] on which our model is built. Let the encoder be E, generator be G and discriminator be D. Denote the training set as $\mathbb{S} = \{(\mathbf{A}^i, \mathbf{B}^i)|\mathbf{A}^i \in \mathbb{A}, \mathbf{B}^i \in \mathbb{B}\}$ where $\mathbb{A}$ and $\mathbb{B}$ stand for the set of conditional and ground-truth images. For simplicity, we use the notations $\mathbf{A}, \mathbf{B}$ for an instance from the respective domain. The Bicycle-GAN model consists of two cycles and resembles two GAN models: cVAE-GAN and cLR-GAN. Now, we describe the two cycles in detail.

cVAE-GAN. The first model is *cVAE-GAN*, which first encodes ground truth image $\mathbf{B}$ to latent code $\widetilde{\mathbf{z}}$ using the encoder E, then reconstructs the ground truth image as $\widetilde{\mathbf{B}}$ given the conditional image $\mathbf{A}$ and encoded latent code $\widetilde{\mathbf{z}}$.

The loss of *cVAE-GAN* consists of three terms: $\mathcal{L}_1^{\text{image}} = \mathbb{E}_{\mathbf{A},\mathbf{B}\sim p(\mathbf{A},\mathbf{B}),\widetilde{\mathbf{z}}\sim E(\mathbf{B})}[||\mathbf{B} - G(\mathbf{A}, \widetilde{\mathbf{z}})||_1]$ which encourages the output of the generator to match the input; $\mathcal{L}_{\text{KL}} = \mathbb{E}_{\mathbf{B}\sim p(\mathbf{B})}[\text{KL}(E(\mathbf{B})||\mathcal{N}(0, \mathbf{I}))]$ which encourages the encoded latent distribution to be close to a standard Gaussian to enable sampling at inference time; and $\mathcal{L}_{\text{GAN}}^{\text{cVAE}}$, the standard adversarial loss which encourages the generator to generate images that are not distinguishable from real images by the discriminator. The objective function of the *cVAE-GAN* is:

$$\mathcal{L}_{\text{cVAE-GAN}} = \min_{G,E} \max_{D} \quad \mathcal{L}_{\text{GAN}}^{\text{cVAE}} + \lambda \mathcal{L}_1^{\text{image}} + \lambda_{\text{KL}} \mathcal{L}_{\text{KL}}, \tag{1}$$

where λ and λ_{KL} are loss weights for encoding and image reconstruction, respectively.

cLR-GAN. The second model is *cLR-GAN*, which first generates a image $\widetilde{\mathbf{B}}$ given the conditional data $\mathbf{A}$ and latent code $\mathbf{z}$, then reconstructs the latent code as $\widetilde{\mathbf{z}}$ to enforce the latent code $\mathbf{z}$ is used.

The loss of *cLR-GAN* consists of two terms: $\mathcal{L}_1^{\text{latent}} = \mathbb{E}_{\mathbf{A}\sim p(\mathbf{A}),\mathbf{z}\sim p(\mathbf{z})}[||\mathbf{z} - E(G(\mathbf{A}, \mathbf{z}))||_1]$ which encourages utilization of the latent code via reconstruction; and $\mathcal{L}_{\text{GAN}}^{\text{cLR}}$, the standard adversarial loss which encourages the generator to generate images that are not distinguishable from real images by the discriminator. The objective function of the *cLR-GAN* is:

$$\mathcal{L}_{\text{cLR-GAN}} = \min_{G,E} \max_{D} \quad \mathcal{L}_{\text{GAN}}^{\text{cLR}} + \lambda_{\text{latent}} \mathcal{L}_1^{\text{latent}}, \tag{2}$$

where λ_{latent} is the loss weight for recovering the latent code.

BicycleGAN is proposed to take advantage of both cycles, hence the objective function is:

$$\mathcal{L}_{\text{BicycleGAN}} = \min_{G,E} \max_{D} \quad \mathcal{L}_{\text{cVAE-GAN}} + \mathcal{L}_{\text{cLR-GAN}}. \tag{3}$$

3.2. Lifelong GAN with Knowledge Distillation

To perform continual learning of conditional generation tasks, the proposed Lifelong GAN is built on top of Bicycle GAN with the adoption of knowledge distillation. We first introduce the problem formulation, followed by a detailed description of our model, then discuss our strategy to tackle the conflicting objectives in training.

Problem Formulation. During training of the t^{th} task, we are given a dataset of N_t paired instances $\mathbb{S}_t = \{(\mathbf{A}_{i,t}, \mathbf{B}_{i,t})|\mathbf{A}_{i,t} \in \mathbb{A}_t, \mathbf{B}_{i,t} \in \mathbb{B}_t\}_{i=1}^{N_t}$ where $\mathbb{A}_t$ and $\mathbb{B}_t$ denote the domain of conditional images and ground truth images respectively. For simplicity, we use the notations $\mathbf{A}_t, \mathbf{B}_t$ for an instance from the respective domain. The goal is to train a model M_t which can generate images of current task $\widetilde{\mathbf{B}}_t \leftarrow (\mathbf{A}_t, \mathbf{z})$, without forgetting how to generate images of previous tasks $\widetilde{\mathbf{B}}_i \leftarrow (\mathbf{A}_i, \mathbf{z})$, $i = 1, 2, ..., (t-1)$.

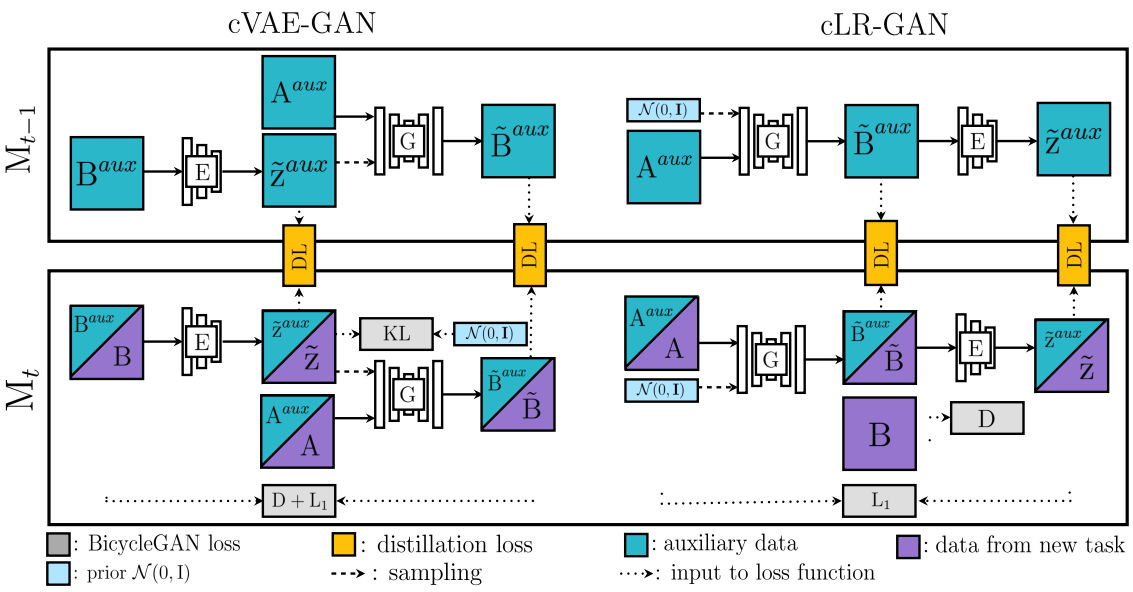


Figure 2: **Overview of Lifelong GAN**. Given training data for the t^{th} task, model M_t is trained to learn this current task. To avoid forgetting previous tasks, knowledge distillation is adopted to distill information from model M_{t-1} to model M_t by encouraging the two networks to produce similar output values or patterns given the auxiliary data as inputs.

Let M_t be the t^{th} model trained, and M_{t-1} be the $(t-1)^{th}$ model trained. Both M_{t-1} and M_t contain two cycles (cVAE-GAN and cLR-GAN) as described in Section 3.1. Inspired by continual learning methods for discriminative models, we prevent the current model M_t from forgetting the knowledge learned by the previous model M_{t-1} by inputting the data of the current task $\mathbb{S}_t$ to both M_t and M_{t-1}, and distilling the knowledge from M_{t-1} to M_t by encouraging the outputs of M_{t-1} and M_t to be similar. We describe the process of knowledge distillation for both cycles as follows.

cVAE-GAN. Recall from Section 3.1 that cVAE-GAN has two outputs: the encoded latent code $\widetilde{\mathbf{z}}$ and the reconstructed ground truth image $\widetilde{\mathbf{B}}$. Given ground truth image $\mathbf{B}_t$, the encoders E_t and E_{t-1} are encouraged to encode it in the same way and produce the same output; given encoded latent code $\widetilde{\mathbf{z}}$ and conditional image $\mathbf{A}_t$, the generators G_t and G_{t-1} are encouraged to reconstruct the ground truth images in the same way. Therefore, we define the loss for the *cVAE-GAN* cycle with knowledge distillation as:

$$\begin{aligned}\mathcal{L}^t_{\text{cVAE}-\text{DL}} &= \mathcal{L}^t_{\text{cVAE}-\text{GAN}} \\ &+ \beta \mathbb{E}_{\mathbf{A}_t,\mathbf{B}_t\sim p(\mathbf{A}_t,\mathbf{B}_t)}\left[||E_t(\mathbf{B}_t) - E_{t-1}(\mathbf{B}_t)||_1\right. \\ &+ \left.||G_t(\mathbf{A}_t, E_t(\mathbf{B}_t)) - G_{t-1}(\mathbf{A}_t, E_{t-1}(\mathbf{B}_t))||_1\right],\end{aligned} \tag{4}$$

where β is the loss weight for knowledge distillation.

cLR-GAN. Recall from Section 3.1 that cLR-GAN also has two outputs: the generated image $\widetilde{\mathbf{B}}$ and the reconstructed latent code $\widetilde{\mathbf{z}}$. Given the latent code $\mathbf{z}$ and conditional image $\mathbf{A}_t$, the generators G_t and G_{t-1} are encouraged to generate images in the same way; given the generated image $\widetilde{\mathbf{B}}_t$, the encoders E_t and E_{t-1} are encouraged to encode the generated images in the same way. Therefore, we define the loss for the *cLR-GAN* cycle as:

$$\begin{aligned}\mathcal{L}^t_{\text{cLR}-\text{DL}} &= \mathcal{L}^t_{\text{cLR}-\text{GAN}} \\ &+ \beta \mathbb{E}_{\mathbf{A}_t\sim p(\mathbf{A}_t),\mathbf{z}\sim p(\mathbf{z})}\left[||G_t(\mathbf{A}_t, \mathbf{z}) - G_{t-1}(\mathbf{A}_t, \mathbf{z})||_1\right. \\ &+ \left.||E_t(G_t(\mathbf{A}_t, \mathbf{z})) - E_{t-1}(G_{t-1}(\mathbf{A}_t, \mathbf{z}))||_1\right].\end{aligned} \tag{5}$$

The distillation losses can be defined in several ways, e.g. the L_2 loss [2, 24], KL divergence [9] or cross-entropy [9, 3]. In our approach, we use L_1 instead of L_2 to avoid blurriness in the generated images.

Lifelong GAN is proposed to adopt knowledge distillation in both cycles, hence the overall loss function is:

$$\mathcal{L}^t_{\text{Lifelong}-\text{GAN}} = \mathcal{L}^t_{\text{cVAE}-\text{DL}} + \mathcal{L}^t_{\text{cLR}-\text{DL}}. \tag{6}$$

Conflict Removal with Auxiliary Data. Note that Equation 4 contains conflicting objectives. The first term encourages the model to reconstruct the inputs of the current task, while the third term encourages the model to generate the same images as the outputs of the old model. In addition, the first term encourages the model to encode the input images to normal distributions, while the second term encourages the model to encode the input images to a distribution learned from the old model. Similar conflicting objectives exist in Equation 5. To sum up, the conflicts appear

when the model is required to produce two different outputs, namely mimicking the performance of the old model and accomplishing the new goal, given the same inputs.

To address these conflicting objectives, we propose to use auxiliary data for distilling knowledge from the old model M_{t-1} to model M_t. The use of auxiliary data for distillation removes these conflicts. It is important that new auxiliary data should be used for each task, otherwise the network could potentially implicitly encode them when learning previous tasks. We describe approaches for doing so without requiring external data sources in Sec. 3.3.

The auxiliary data $\mathbb{S}_t^{\text{aux}} = \{(\mathbf{A}_{i,t}^{\text{aux}}, \mathbf{B}_{i,t}^{\text{aux}}) | \mathbf{A}_{i,t}^{\text{aux}} \in \mathbb{A}_t^{\text{aux}}, \mathbf{B}_{i,t}^{\text{aux}} \in \mathbb{B}_t^{\text{aux}}\}_{i=1}^{N_t}$ consist of N_t^{aux} training pairs where $\mathbb{A}_t^{\text{aux}}$ and $\mathbb{B}_t^{\text{aux}}$ denote the domain of auxiliary conditional data and ground truth data respectively. For simplicity, we use the notations $\mathbf{A}_t^{\text{aux}}, \mathbf{B}_t^{\text{aux}}$ for an instance from the respective domain.

The losses $\mathcal{L}_{\text{cVAE-DL}}^t$ and $\mathcal{L}_{\text{cLR-DL}}^t$ are re-written as:

$$\begin{aligned}\mathcal{L}_{\text{cVAE-DL}}^t &= \mathcal{L}_{\text{cVAE-GAN}}^t \\ &+ \beta \mathbb{E}_{\mathbf{A}_t^{\text{aux}}, \mathbf{B}_t^{\text{aux}} \sim p(\mathbf{A}_t^{\text{aux}}, \mathbf{B}_t^{\text{aux}})} [||E_t(\mathbf{B}_t^{\text{aux}}) - E_{t-1}(\mathbf{B}_t^{\text{aux}})||_1 \\ &+ ||G_t(\mathbf{A}_t^{\text{aux}}, E_t(\mathbf{B}_t^{\text{aux}})) - G_{t-1}(\mathbf{A}_t^{\text{aux}}, E_{t-1}(\mathbf{B}_t^{\text{aux}}))||_1], \end{aligned} \tag{7}$$

$$\begin{aligned}\mathcal{L}_{\text{cLR-DL}}^t &= \mathcal{L}_{\text{cLR-GAN}}^t \\ &+ \beta \mathbb{E}_{\mathbf{A}_t^{\text{aux}} \sim p(\mathbf{A}_t^{\text{aux}}), \mathbf{z} \sim p(\mathbf{z})} [||G_t(\mathbf{A}_t^{\text{aux}}, \mathbf{z}) - G_{t-1}(\mathbf{A}_t^{\text{aux}}, \mathbf{z})||_1 \\ &+ ||E_t(G_t(\mathbf{A}_t^{\text{aux}}, \mathbf{z})) - E_{t-1}(G_{t-1}(\mathbf{A}_t^{\text{aux}}, \mathbf{z}))||_1], \end{aligned} \tag{8}$$

where β is the loss weight for knowledge distillation.

Lifelong GAN can be used for continual learning of both image-conditioned and label-conditioned generation tasks. The auxiliary images for knowledge distillation for both settings can be generated using the Montage and Swap operations described in Section 3.3. For label-conditioned generation, we can simply use the categorical codes from previous tasks.

3.3. Auxiliary Data Generation

We now discuss the generation of auxiliary data. Recall from Section 3.2 that we use auxiliary data to address the conflicting objectives in Equations 4 and 5.

The auxiliary images do not require labels, and can in principle be sourced from online image repositories. However, this solution may not be scalable as it requires a new set of auxiliary images to be collected when learning each new task. A more desirable alternative may be to generate auxiliary data by using the current data in a way that avoids the over-fitting problem. We propose two operations for generating auxiliary data from the current task data:

1. Montage: Randomly sample small image patches from current input images and montage them together to produce auxiliary images for distillation.
2. Swap: Swap the conditional image $\mathbf{A}_t$ and the ground truth image $\mathbf{B}_t$ for distillation. Namely the encoder receives the conditional image $\mathbf{A}_t$ and encodes it to a latent code $\widetilde{\mathbf{z}}$, and the generator is conditioned on the ground truth image $\mathbf{B}_t$.

Both operations are used in image-conditioned generation; in label-conditioned generation, since there is no conditional image, only the montage operation is applicable.

Other alternatives may be possible. Essentially, the auxiliary data generation needs to provide out-of-task samples that can be used to preserve the knowledge learned by the old model. The knowledge is preserved using the distillation losses, which encourage the old and new models to produce similar responses on the out-of-task samples.

4. Experiments

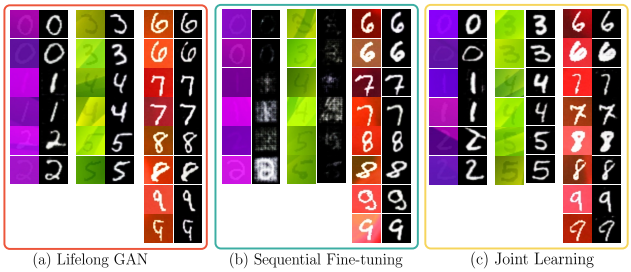
We evaluate Lifelong GAN for two settings: (1) image-conditioned image generation, and (2) label-conditioned image generation. We are the first to explore continual learning for image-conditioned image generation; no existing approaches are applicable for comparison. Additionally, we compare our model with the memory replay based approach which is the state-of-the-art for label-conditioned image generation.

Training Details. All the sequential digit generation models are trained on images of size 64×64 and all other models are trained on images of size 128×128. We use the Tensorflow [1] framework with Adam Optimizer [12] and a learning rate of 0.0001. We set the parameters $\lambda_{\text{latent}} = 0.5$, $\lambda_{\text{KL}} = 0.01$, and $\beta = 5.0$ for all experiments. The weights of generator and encoder in *cVAE-GAN* and *cLR-GAN* are shared. Extra training iterations on the generator and encoder using only distillation loss are used for models trained on images of size 128×128 for better remembering previous tasks.

Baseline Models. We compare Lifelong GAN to the following baseline models: (a) *Memory Replay (MR)*: Images generated by a generator trained on previous tasks are combined with the training images for the current task to form a hybrid training set. (b) *Sequential Fine-tuning (SFT)*: The model is fine-tuned in a sequential manner, with parameters initialized from the model trained/fine-tuned on the previous task. (c) *Joint Learning (JL)*: The model is trained utilizing data from all tasks.

Note that for image-conditioned image generation, we only compare with joint learning and sequential fine-tuning methods, as memory replay based approaches are not applicable without any ground-truth conditional input.

Quantitative Metrics. We use different metrics to evaluate different aspects of the generation. In this work, we use *Acc*, *r-Acc* and *LPIPS* to validate the quality of the generated data. *Acc* is the accuracy of the classifier network



(a) Lifelong GAN (b) Sequential Fine-tuning (c) Joint Learning

Figure 3: Comparison among different approaches for continual learning of MNIST digit segmentations. Lifelong GAN can learn the current task without forgetting the previous ones.

		SF	JL	Ours
MNIST	Acc	58.02	94.25	95.90
	r-Acc	61.56	96.79	96.14
	LPIPS	-	0.150	0.157
Image-to-Image	Acc	39.72	99.26	98.93
	r-Acc	49.88	98.98	99.37
	LPIPS	-	0.442	0.417

Table 1: Quantitative evaluation for image-conditioned generation. For MNIST digit generation, LPIPS for real images is 0.154. For image-to-image translation, LPIPS for real images is 0.472.

trained on real images and evaluated on generated images (higher indicates better generation quality). *r-Acc* is the accuracy of the classifier network trained on generated images and evaluated on real images (higher indicates better generation quality). *LPIPS* [33] is used to quantitatively evaluate the diversity as used in BicycleGAN [35]. Higher LPIPS indicates higher diversity. Furthermore, LPIPS closer to the ones of real images indicates more realistic generation.

4.1. Image-conditioned Image Generation

Digit Generation. We divide the digits in MNIST [14] into 3 groups: {0,1,2}, {3,4,5}, and {6,7,8,9}[1]. The digits in each group are dyed with a signature color as shown in Figure 3. Given a dyed image, the task is to generate a foreground segmentation mask for the digit (i.e. generate a foreground segmentation given a dyed image as condition). The three groups give us three tasks for sequential learning.

Generated images from the last task for all approaches are shown in Figure 3. We can see that sequential fine-tuning suffers from catastrophic forgetting (it is unable to segment digits 0-5 from the previous tasks), while our approach can learn to generate segmentation masks for the current task without forgetting the previous tasks.

[1]group {a,b,c} contains digits with label a, b and c. This applies to all experiments on MNIST.

		SFT	JL	MR	Ours
MNIST	Acc	21.59	98.08	97.54	97.52
	r-Acc	21.21	87.72	85.57	87.77
	LPIPS	-	0.125	0.120	0.119
Flower	Acc	20.0	96.4	87.6	98.4
	r-Acc	19.6	83.6	60.4	85.6
	LPIPS	-	0.413	0.319	0.399

Table 2: Quantitative evaluation for label-conditioned image generation tasks. For MNIST digit generation, LPIPS for real images is 0.155. For flower image generation, LPIPS for real images is 0.479.

Image-to-image Translation. We also apply Lifelong GAN to more challenging domains and datasets with large variation for higher resolution images. The first task is image-to-image translation of edges → shoes photos [31, 29]. The second task is image-to-image translation of segmentations → facades [22]. The goal of this experiment is to learn the task of semantic segmentations → facades without forgetting the task edges → shoe photos. We sample ~20000 image pairs for the first task and use all images for the second task.

Generated images for all approaches are shown in Figure 4. For both Lifelong GAN and sequential fine-tuning, the model of *Task2* is initialized from the same model trained on *Task1*. We show the generation results of each task for Lifelong GAN. For sequential fine-tuning, we show the generation results of the last task. It is clear that the sequentially fine-tuned model completely forgets the previous task and can only generate incoherent facade-like patterns. In contrast, Lifelong GAN learns the current generative task while remembering the previous task. It is also observed that Lifelong GAN is capable of maintaining the diversity of generated images of the previous task.

4.2. Label-conditioned Image Generation

Digit Generation. We divide the MNIST [14] digits into 4 groups, {0,1,2}, {3,4}, {5,6,7} and {8,9}, resulting in four tasks for sequential learning. Each task is to generate binary MNIST digits given labels (one-hot encoded labels) as conditional inputs.

Visual results for all methods are shown in Figure 5, where we also include outputs of the generator after each task for our approach and memory replay. Sequential fine-tuning results in catastrophic forgetting, as shown by this baseline's inability to generate digits from any previous tasks; when given a previous label, it will either generate something similar to the current task or simply unrecognizable patterns. Meanwhile, both our approach and memory replay are visually similar to joint training results, indicating that both are able to address the forgetting issue in

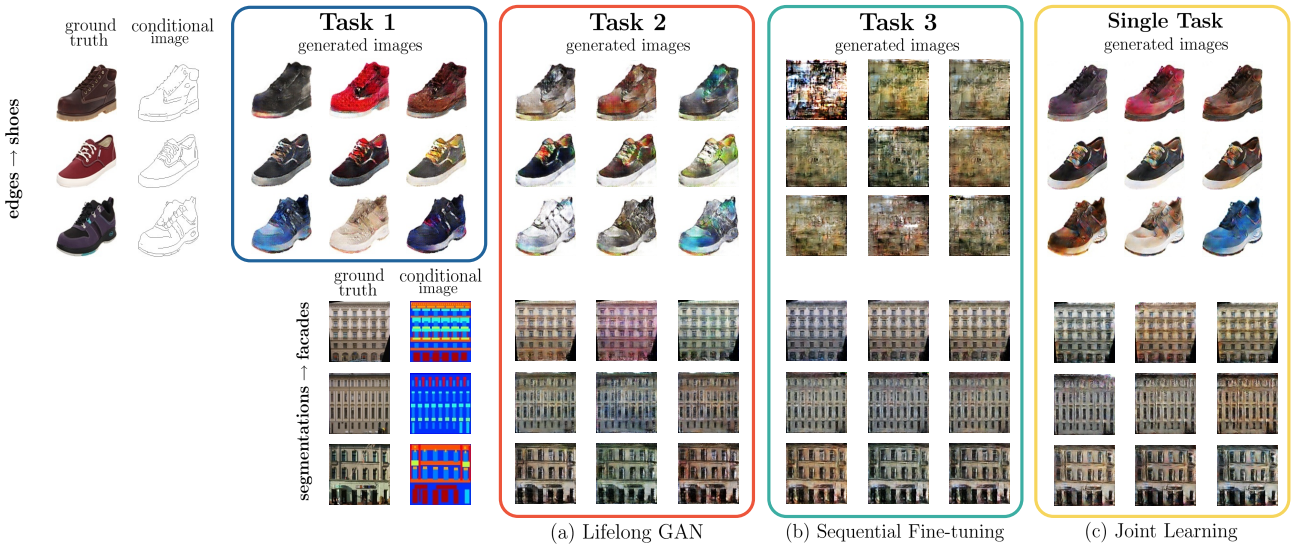


Figure 4: Comparison among different approaches for continual learning of image to image translation tasks. Given the same model trained for the task *edges* → *shoes*, we train Lifelong GAN and sequential fine-tuning model on the task *segmentations* → *facades*. Sequential fine-tuning suffers from severe catastrophic forgetting. In contrast, Lifelong GAN can learn the current task while remembering the old task.

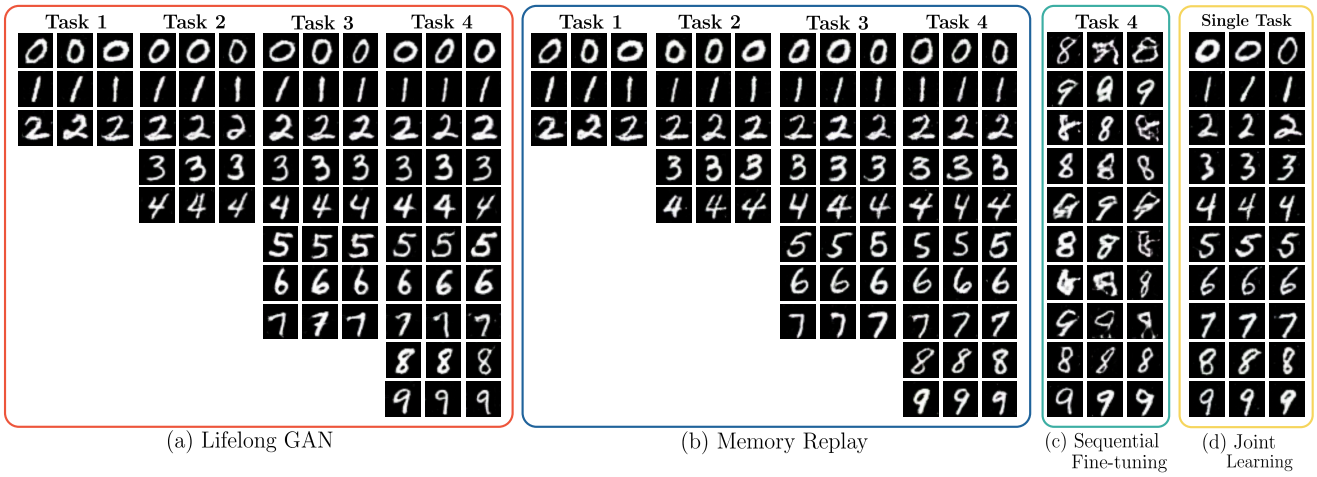


Figure 5: Comparison among different approaches for continual learning of MNIST digit generation conditioned on label. We demonstrate some intermediate results during different tasks of continual learning for our distillation based approach and memory replay. Sequential fine-tuning suffers from severe forgetting issues while other methods give visually similar results compared to the joint learning results.

this task. Quantitatively, our method achieves comparable classification accuracy to memory replay, and outperforms memory replay in terms of reverse classification accuracy.

Flower Generation. We also demonstrate Lifelong GAN on a more challenging dataset, which contains higher resolution images from five categories of the Flower dataset [20]. The experiment consists of a sequence of five tasks in the order of *sunflower*, *daisy*, *iris*, *daffodil*, *pansy*. Each task involves learning a new category.

Generated images for all approaches are shown in Figure 6. We show the generation results of each task for both Lifelong GAN and memory replay to better analyze these two methods. For sequential fine-tuning, we show the generation results of the last task which is enough to show that the model suffers from catastrophic forgetting.

Figure 6 gives useful insights into the comparison between Lifelong GAN and memory replay. Both methods can learn to generate images for new tasks while remembering previous ones. However, memory replay is more sensitive to generation artifacts appearing in the intermediate

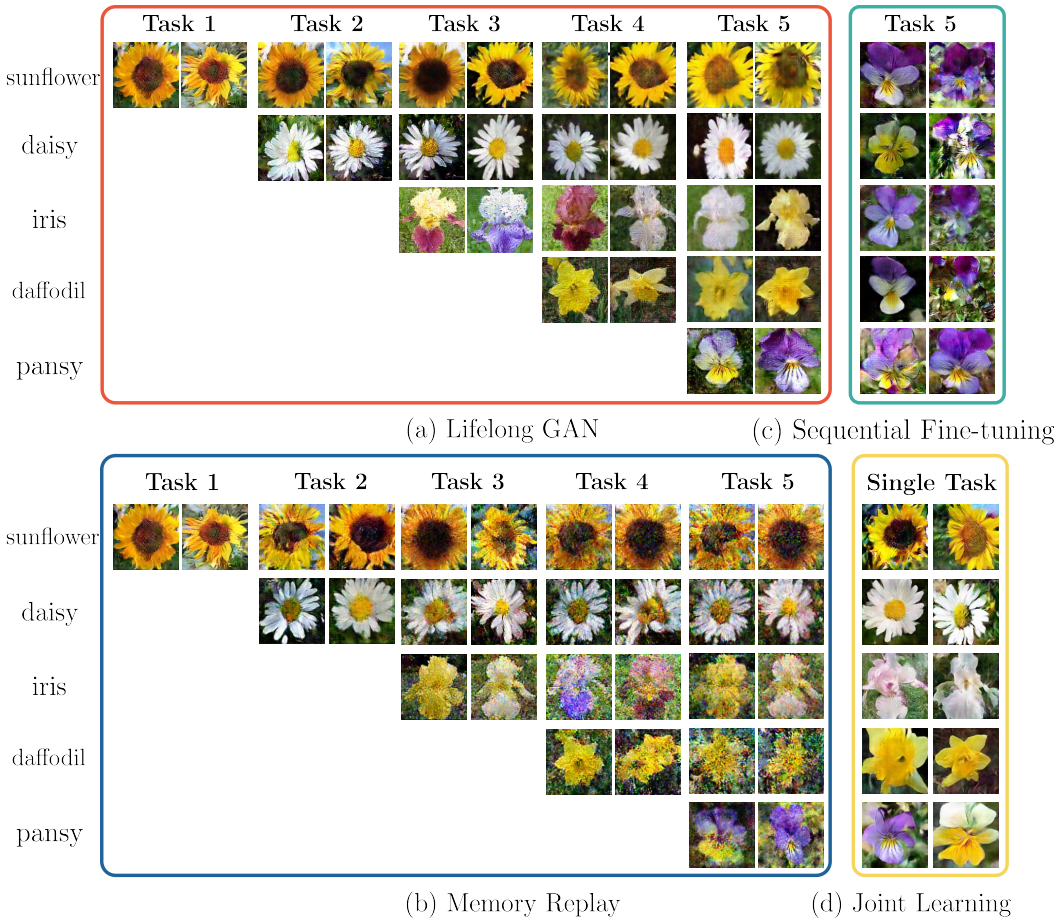


Figure 6: Comparison among different approaches for continual learning of flower image generation tasks. Given the same model trained for category *sunflower*, we train Lifelong GAN, memory replay and sequential fine-tuning model for other tasks. Sequential fine-tuning suffers from severe catastrophic forgetting, while both Lifelong GAN and memory replay can learn to perform the current task while remembering the old tasks. Lifelong GAN is more robust to artifacts in the generated images of the middle tasks, while memory replay is much more sensitive and all later tasks are severely impacted by these artifacts.

tasks of sequential learning. While training *Task3* (category iris), both Lifelong GAN and memory replay show some artifacts in the generated images. For memory replay, the artifacts are reinforced during the training of later tasks and gradually spread over all categories. In contrast, Lifelong GAN is more robust to the artifacts and later tasks are much less sensitive to intermediate tasks. Lifelong GAN treats previous tasks and current tasks separately, trying to learn the distribution of new tasks while mimicking the distribution of the old tasks.

Table 2 shows the quantitative results. Lifelong GAN outperforms memory replay by 10% in terms of classification accuracy and 25% in terms of reverse classification accuracy. We also observed visually and quantitatively that memory replay tends to lose diversity during the sequential learning, and generates images with little diversity for the final task.

5. Conclusion

We study the problem of lifelong learning for generative networks and propose a distillation based continual learning framework enabling a single network to be extended to new tasks without forgetting previous tasks with only supervision for the current task. Unlike previous methods that adopt memory replay to generate images from previous tasks as training data, we employ knowledge distillation to transfer learned knowledge from previous networks to the new network. Our generic framework enables a broader range of generation tasks including image-to-image translation, which is not possible using memory replay based methods. We validate Lifelong GAN for both image-conditioned and label-conditioned generation tasks, and both qualitative and quantitative results illustrate the generality and effectiveness of our method.

References

[1] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, et al. Tensorflow: A system for large-scale machine learning. In *Symposium on Operating Systems Design and Implementation (OSDI)*, 2016.

[2] J. Ba and R. Caruana. Do deep nets really need to be deep? In *Advances in neural information processing systems (NIPS)*, 2014.

[3] F. M. Castro, M. J. Marín-Jiménez, N. Guil, C. Schmid, and K. Alahari. End-to-end incremental learning. In *European Conference on Computer Vision (ECCV)*, 2018.

[4] D. Chen, L. Yuan, J. Liao, N. Yu, and G. Hua. Stylebank: An explicit representation for neural image style transfer. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[5] X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *Advances in Neural Information Processing Systems (NIPS)*, 2016.

[6] L. Chongxuan, T. Xu, J. Zhu, and B. Zhang. Triple generative adversarial nets. In *Advances in Neural Information Processing Systems (NIPS)*, 2017.

[7] W. Churchill and P. Newman. Practice makes perfect? managing and leveraging visual experiences for lifelong navigation. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2012.

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

[9] G. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network. In *NIPS workshop on Deep Learning and Representation Learning*, 2015.

[10] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[11] J. Johnson, A. Alahi, and L. Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European Conference on Computer Vision (ECCV)*, 2016.

[12] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015.

[13] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska, D. Hassabis, C. Clopath, D. Kumaran, and R. Hadsell. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences of the United States of America*, 2017.

[14] Y. LeCun. The mnist database of handwritten digits. *http://yann. lecun. com/exdb/mnist/*, 1998.

[15] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[16] F. Luan, S. Paris, E. Shechtman, and K. Bala. Deep photo style transfer. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[17] L. Ma, X. Jia, Q. Sun, B. Schiele, T. Tuytelaars, and L. Van Gool. Pose guided person image generation. In *Advances in Neural Information Processing Systems (NIPS)*, 2017.

[18] M. McCloskey and N. J. Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of Learning and Motivation*. 1989.

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

[20] M.-E. Nilsback and A. Zisserman. A visual vocabulary for flower classification. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2006.

[21] A. Polino, R. Pascanu, and D. Alistarh. Model compression via distillation and quantization. In *International Conference on Learning Representations (ICLR)*, 2018.

[22] R. Š. Radim Tyleček. Spatial pattern templates for recognition of objects with regular structure. In *German Conference on Pattern Recognition (GCPR)*, 2013.

[23] A. Seff, A. Beatson, D. Suo, and H. Liu. Continual learning in generative adversarial nets. In *Advances in Neural Information Processing Systems (NIPS)*, 2017.

[24] K. Shmelkov, C. Schmid, and K. Alahari. Incremental learning of object detectors without catastrophic forgetting. In *IEEE International Conference on Computer Vision*, 2017.

[25] V. Vapnik and R. Izmailov. Learning using privileged information: similarity control and knowledge transfer. *Journal of Machine Learning Research (JMLR)*, 2015.

[26] R. Villegas, J. Yang, Y. Zou, S. Sohn, X. Lin, and H. Lee. Learning to generate long-term future via hierarchical prediction. In *International Conference on Machine Learning (ICML)*, 2017.

[27] X. Wang, R. Zhang, Y. Sun, and J. Qi. Kdgan: Knowledge distillation with generative adversarial networks. In *Advances in Neural Information Processing Systems (NIPS)*, 2018.

[28] C. Wu, L. Herranz, X. Liu, Y. Wang, J. van de Weijer, and B. Raducanu. Memory replay gans: learning to generate images from new categories without forgetting. In *Advances in Neural Information Processing Systems (NIPS)*, 2018.

[29] S. Xie and Z. Tu. Holistically-nested edge detection. In *IEEE International Conference on Computer Vision (ICCV)*, 2015.

[30] R. A. Yeh, C. Chen, T. Yian Lim, A. G. Schwing, M. Hasegawa-Johnson, and M. N. Do. Semantic image inpainting with deep generative models. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[31] A. Yu and K. Grauman. Fine-Grained Visual Comparisons with Local Learning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[32] M. Zhai, R. Deng, J. Chen, L. Chen, Z. Deng, and G. Mori. Adaptive appearance rendering. In *British Machine Vision Conference (BMVC)*, 2018.

[33] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[34] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *IEEE International Conference on Computer Vision (ICCV)*, 2017.

[35] J.-Y. Zhu, R. Zhang, D. Pathak, T. Darrell, A. A. Efros, O. Wang, and E. Shechtman. Toward multimodal image-to-image translation. In *Advances in Neural Information Processing Systems (NIPS)*, 2017.