Feedback Adversarial Learning: Spatial Feedback for Improving Generative Adversarial Networks

Minyoung Huh*
UC Berkeley
minyoungg@berkeley.edu

Shao-Hua Sun*
University of Southern California
shaohuas@usc.edu

Ning Zhang
Vaitl Inc.
ning@vaitl.ai

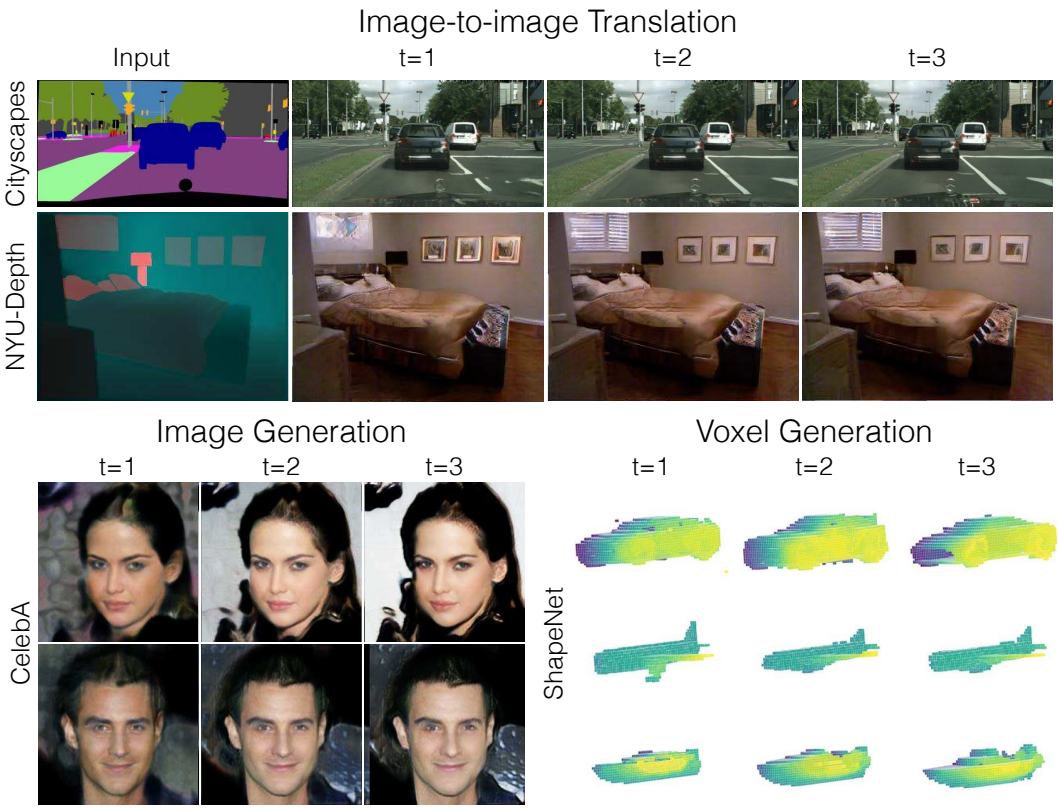

Figure 1: Results using feedback adversarial learning on various generative adversarial learning tasks. Our model learns to utilize the feedback signal from the discriminator and iteratively improve the generation quality with more generation steps.

Abstract

We propose feedback adversarial learning (FAL) framework that can improve existing generative adversarial networks by leveraging spatial feedback from the discriminator. We formulate the generation task as a recurrent framework, in which the discriminator's feedback is integrated into the feedforward path of the generation process. Specifically, the generator conditions on the discriminator's spatial output response, and its previous generation to improve generation quality over time – allowing the generator to attend and fix its previous mistakes. To effectively utilize the feedback, we propose an adaptive spatial transform (AST) layer, which learns to spatially modulate feature maps from its previous generation and the feedback signal from the discriminator. We demonstrate that one can easily adapt our method to improve existing adversarial learning frameworks on a wide range of tasks, including image generation, image-to-image translation, and voxel generation. The project website can be found at https://minyoungg.github.io/feedbackgan.

*Authors contributed equally.

1. Introduction

A masterpiece is not created in a day. Even with countless hours of training, an expert can still make a mistake and learn how to improve. The key to success is an endless cycle of feedback and revision between an artist and a critic, where the artist can refine its existing work together with the critic's feedback – something that is missing in the current generative adversarial network (GAN) training frameworks. In traditional GAN setting, the discriminator acts as a critic, providing only gradient signals to the generator; however, the generator does not get the second chance to look at its own generation along with the feedback from the discriminator to improve upon. The generation task becomes notoriously difficult with increasing data complexity (e.g. image dimension, data variations). Therefore, to alleviate the difficulty of the generation task, we propose feedback adversarial learning (FAL) framework for integrating the discriminator's feedback in the feed-forward path of the generation process, allowing the generator to iteratively improve its generation.

A generative adversarial network consists of 2 networks: a generator (G) and a discriminator (D), where the goal of the generator G is to generate a sample $\hat{y}$ from a latent noise vector $z \in \mathbb{R}^{z_d}$ sampled from a known distribution (e.g. $\mathcal{N}(0, I)$). These generated samples $\hat{y}$ should be indistinguishable to the discriminator from real samples y. Since the introduction of GANs, there has been extensive interest in improving generation quality. Due to the instability of training GANs, different optimization methods [3, 19], normalization [53], and advanced training techniques such as progressive generation [28] have been proposed.

For all adversarial learning paradigms, the discriminator provides gradients as a learning signal for the generator and is discarded during testing time. To successfully train a model, the designer has to find the right equilibrium between the discriminator being too powerful and being an informative signal. Even in successfully trained models, the discriminator easily outperforms the generator, an indication that there exists information in the discriminator that the generator could still utilize. This motivates the idea of allowing the generator to leverage additional information from the discriminator during generation. As shown in Figure 2, the generator looks at the previous generation and the discriminator's response to drive the generation for the next time step. In fact, the idea of using the feedback originates from well-established control theory, where one uses the error signals to propagate adjustments for the input signal. Similarly, we demonstrate that using the discriminator signal as an error propagation signal allows the generator to attend to the regions that look unrealistic and iteratively generate higher quality samples over time.

To effectively leverage the discriminator's feedback, we propose adaptive spatial transform (AST) module, which allows the generator to spatially modulate input features based on the feedback signal. We demonstrate the feasibility and effectiveness of applying feedback adversarial learning to several frameworks on a wide range of tasks, including image generation, image-to-image translation, and voxel generation, as well as provide qualitative and quantitative results on various datasets. As shown in Figure 1, the models trained with FAL learn to improve their generation using the discriminator's feedback. We extensively evaluate the generated samples with a variety of metrics, including FID, segmentation score, depth prediction, LPIPS, and classification accuracy.

2. Related Works

Generative adversarial networks Generative adversarial networks [18] use a discriminator to model the data distribution, which acts as a loss function to provide the generator a learning signal to generate realistic samples. GANs have continued to show impressive results in various applications such as image generation [46, 15, 63, 8], text to image synthesis [47, 26], future frame prediction [40, 55], image editing [67], novel view synthesis [52, 43], domain adaptation [7, 51], 3D modeling [59, 27], video generation [66], video re-targeting [4], text generation [61, 20], audio generation [16, 17, 22], etc.

Image generation Synthesizing images using convolutional neural networks has been popularized by GANs but the history goes far back. The community has explored variational auto-encoders [29], auto-regressive models [1], etc. More recently, [12] demonstrated that using perceptual-loss and coarse-to-fine generation can be used to synthesize photo-realistic images without having a discriminator.

Image-to-image translation [25] demonstrated GANs can also be applied on paired image-to-image translation. This sparked the vision and graphics community to apply adversarial image translation on various tasks. With the difficulty of collecting interesting paired data, many works [68, 35, 60, 34, 6, 48] have proposed alternative methods to translate images. This task is now known as unpaired image-to-image translation — a task of learning a mapping from two arbitrary domains without having any paired images.

Optimization and training frameworks With the difficulty that arises when training GANs, the community has been trying to improve GANs through different methods of optimization and normalization. Few to mention are least-squares [37] and Wasserstein-distance loss [3] and its follow-up work using gradient penalty [19]. Beyond optimization, many have also found that weight normalization [53, 50] helps stabilize training and generate better results. Moreover, many training paradigms have been proposed to stabilize training, where the use of coarse-to-fine

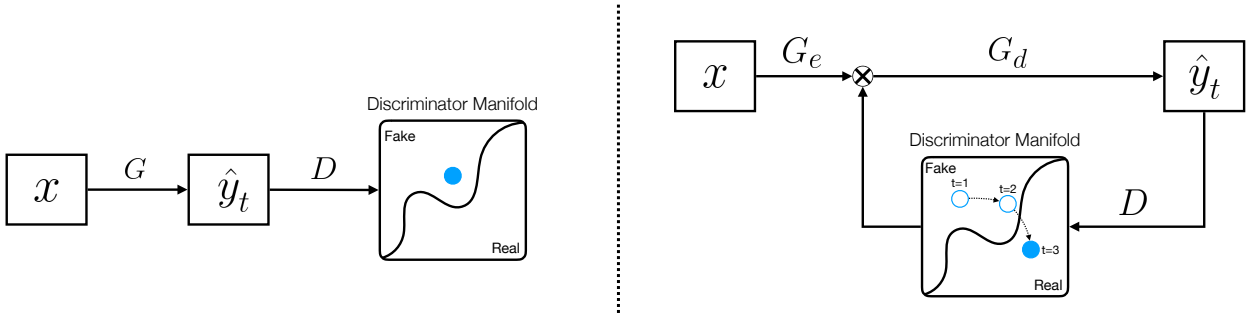

Figure 2: **Feedback adversarial learning:** On the **left** is a typical GAN setup, and on the **right** we have our feedback adversarial learning setup. At each time step, the generator generates a single image. Our method uses the discriminator output decision map and the previously generated image to drive the generation for the next time step. The discriminator manifold is a visualization of the discriminator's belief of whether a sample looks generated or real. The blue circle indicates the generated image in the discriminator's manifold. The trailing empty blue circles are previous generations and the curved lines indicate the discriminator's decision boundary. For the task of generating images from latent vector, input x is replaced with latent code z.

and unrolled predictions [41, 64, 58, 28, 20, 24] have shown promising results.

Feedback learning Leveraging feedback to iteratively improve the performance has been explored on classification [62], object recognition [32], and human pose estimation [9, 5].

In our approach, we propose a simple yet effective method that uses the discriminator's spatial output and the previous generation as a signal for the generator to improve. The output of the discriminator indicates which regions of a sample look real or fake; hence, the generator can attend to those unrealistic regions and improve them. Our method can be applied to any existing architectures and optimization methods to generate higher quality samples.

3. Generative Adversarial Networks

A generative adversarial network (GAN) consists of 2 networks: a generator G and a discriminator D. The goal of the generator is to generate realistic samples from a noise vector z, $G : z \rightarrow \hat{y}$, such that the discriminator cannot disambiguate a real sample y from a generated sample $\hat{y}$. An unconditional GAN can be formulated as:

$$\hat{y} = G(z). \tag{1}$$

In conditional GANs, the generator conditions its generation on additional information x, $G : (z, x) \rightarrow \hat{y}$, where x is a conditional input such as an image (e.g. segmentation map, depth map) or class information (e.g. ImageNet class, face attributes); in the latter case the task is called class-conditional image generation. When the input and the output domains are images, this task is referred to as image-to-image translation. In image-to-image translation, we have the following formulation, although the latent noise vector z is often not used:

$$\hat{y} = G(z, x). \tag{2}$$

The goal of the discriminator is to discriminate generated samples from real samples. Hence, the objective function of the generator is to maximize the log-likelihood of fooling the discriminator with the generated samples. The overall objective can be written as:

$$\min_G \max_D \mathbb{E}_{y \sim q_{data}}[\log D(y)] + \mathbb{E}_{z \sim p_z}[\log(1 - D(G(z))], \tag{3}$$

where q_{data} is the real data distribution and p_z is the sample distribution such as the normal distribution $\mathcal{N}(0, I)$. For the task of image-to-image translation, the sample distribution comes from $x \sim p_x$ and an additional reconstruction loss on $\hat{y}$ is incurred on the generator: $\mathcal{L}_{rec} = \|\hat{y} - y\|_p$ for some norm p. Other works have explored using perceptual loss or cycle-consistency loss instead.

4. Method

In a standard adversarial learning setting, the generator only gets a single attempt to generate the image, and the discriminator only provides gradients as a learning signal for the generator. Instead, we propose a feedback adversarial learning (FAL) framework which leverages the discriminator's spatial output as feedback to allow the generator to locally attend to and improve its previous generation. The proposed method can be easily adapted to any GAN framework on a variety of tasks. We introduce our method in the following sections. First, we decompose the generation procedure as a two-stage process, in Section 4.1. In Section 4.2, we define the formulation of feedback adversarial

learning. In Section 4.3, we propose a method that allows the generator to effectively utilize the spatial feedback information.

4.1. Reformulation

To simplify describing the idea of feedback learning in GANs, we first reformulate a generator G as a 2-part model: an encoder G_e which encodes input information and a decoder G_d which then decodes the intermediate encoding into the target domain. This is well demonstrated in conditional image-to-image translation GANs where an encoder network G_e maps information x (e.g. image) into some encoded features h, $G_e : x \rightarrow h$, and a decoder G_d maps the intermediate representation h back into the image space y, $G_d : h \rightarrow y$. Note that the choice of where to re-define the generator as an encoder and decoder can be chosen arbitrarily. We can write the generation process as:

$$\hat{y} = G(x) = G_d(G_e(x)), \tag{4}$$

where $\hat{y}$ denotes an output image. For the case of unconditional GANs, this can be described as $\hat{y} = G(z) = G_d(G_e(z))$.

4.2. Feedback Adversarial Learning

We now define our feedback adversarial learning framework, where the generator aims to iteratively improve its generations by using discriminator's feedback information. To enable the generator to attend to specific regions of its generation, we utilize local discriminators [30, 25] which output a response map instead of a scalar, where each pixel corresponds to the decision made from a set of input pixels in a local receptive field. We formulate the generation task as a recurrent process, where the generator is trained to fix the mistakes of its previous generation by leveraging the discriminator's response map and produce a better image.

We denote the generated image for some arbitrary time step t as $\hat{y}_t$ and the encoding at time step t as h_t. Then, the discriminator response map of the generated image at time step t can be written as:

$$r_t = D(\hat{y}_t), \tag{5}$$

where $r_t \in \mathbb{R}^{H/c \times W/c}$ is the output of the discriminator with dimension scaling constant c corresponding to the choice of the discriminator architecture. Here, H and W indicate the original image height and width. This can be generalized to other data domains such as voxels, where $r_t \in \mathbb{R}^{H/c \times W/c \times Dep/c}$ with Dep representing depth. This response map is indicative of whether certain regions in the image look fake or real to the discriminator.

To leverage the previous image generation $\hat{y}_{t-1}$ and its discriminator response r_{t-1}, we design a feedback network

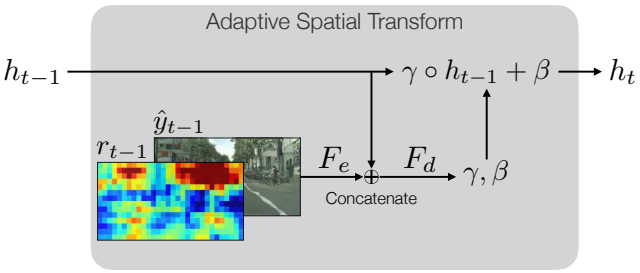

Figure 3: **Adaptive Spatial Transform:** We propose to utilize feedback information by predicting affine parameters γ and β to locally modulate the input feature h_{t_1}. The previously generated image $\hat{y}_{t-1}$ and the discriminator's response r_{t-1} are passed through the feedback encoder F_e, and is concatenated with h_{t-1} to predict affine parameters γ and β using the feedback decoder F_d. The predicted affine parameters have the same dimension as h and is used to scale and bias the existing features per-element.

F, explained in the next section, to inject feedback information into the input encoding h_t. We now redefine the generator Equation 4 for time step t as:

$$y_t = G_d(F(h_{t-1}, y_{t-1}, r_{t-1})) = G_d(h_t). \tag{6}$$

At time step $t = 1$, the input embedding $G_e(x)$ or $G_e(z)$ is computed once to initialize h_0, while y_0 and r_0 are initialized as zero tensors. To train both the generator and the discriminator, we compute the same loss in Equation 3 at every time step and compute the mean across all the time steps.

4.3. Adaptive Spatial Transform

We propose Adaptive Spatial Transform (AST) to effectively utilize the information from the previous time step to modulate the encoded features h. Our method is inspired by [21, 23, 45, 56], which uses external information to predict scalar affine parameters γ and β per channel to linearly transform the features:

$$\hat{h} = \gamma \cdot h + \beta, \tag{7}$$

where $\gamma, \beta \in \mathbb{R}^C$ with C indicating the number of channels. These methods result in a global transformation on the whole feature map. Instead, to allow the generator to modulate features locally, we propose adaptive spatial transform layer, which spatially scales and bias the individual elements as shown in Figure 3. This allows for a controlled spatial transformation. A similar idea has been explored in a concurrent work [44].

To implement this idea, we decompose the feedback network F into 2 sub-networks: a feedback encoder F_e and a feedback decoder F_d. We first use the previously generated

image and the discriminator decision map to predict feedback feature f_{t-1} using the feedback encoder F_e:

$$f_{t-1} = F_e(\hat{y}_{t-1}, r_{t-1}). \tag{8}$$

The encoded feedback information $f_{t-1} \in \mathbb{R}^{H' \times W' \times C}$ has the same dimension as the encoded input feature h_{t-1}, with H' and W' indicating the spatial dimension of the encoding h_{t-1}. Note that the response map r_{t-1} is bilinearly upsampled to match the dimension of the generated image $\hat{y}_{t-1}$ and is concatenated to y_{t-1} across the channel dimension. Finally, the encoded input features and feedback features are concatenated and used to predict transformation parameters using the feedback decoder F_d:

$$\gamma,\ \beta = F_d(h_{t-1}, f_{t-1}). \tag{9}$$

The predicted affine parameters γ, β have the same dimension as h (*i.e.* with spatial dimensions) and are used to spatially scale and bias the input features:

$$h_t = \gamma \circ h_{t-1} + \beta, \tag{10}$$

where $\circ$ and $+$ denote the Hadamard product and an element wise addition. The transformed encoding h_t is then used as an input to the decoder to produce an improved image $y_t = G_d(h_t)$. The scale parameter γ is one-centered and the bias parameter β is zero-centered. We keep track of the transformed input encoding for future feedback generations. We demonstrate the effectiveness of the proposed adaptive spatial transform in Section 5.

5. Experiments

We demonstrate how to leverage the proposed feedback adversarial learning technique on a variety of tasks to improve existing GAN frameworks.

5.1. Experimental Setup

Image generation We first demonstrate our method on the image generation task, where the goal of the generator is to generate an image from a latent vector sampled from a known distribution. We take influences from the recent state-of-the-art architecture BigGAN-deep [8] and constructed our own GAN. We made some modification to make the network feasible to fit on a commercial GPU. Specifically, we removed self-attention layer [57, 63] and reduced the generator and discriminator depth by half. We use 64 filters for both the generator and the discriminator instead of 128, and use instance norm and adaptive instance norm [23] instead of batch norm and conditional batch norm [21]. Furthermore, we do not pool over the last layer to preserve the spatial output of the discriminator. We train the model to optimize the hinge version of the adversarial loss [33, 53, 54] with a batch size of 16. Further architecture details can be found in the appendix.

Image-to-Image translation We further apply our method to the image-to-image translation task, where the goal of the generator is to map images from one domain to another. We use a generator consisting of 9-Residual blocks, identical to the one from [68]. We train the model to optimize the least-squares loss proposed in [38] and scale the reconstruction loss by 10. We made some modifications to improve the overall performance, and further details can be found in the appendix.

Voxel Generation To investigate if the proposed feedback adversarial learning mechanism can generalize beyond 2D images, we demonstrate our method on the task of voxel generation [59, 14, 31]. The goal of the generator is to produce realistic voxels, represented by a binary occupancy cube $V \in \mathbb{R}^{H \times W \times Dep}$, from a randomly sampled latent vector z. Similar to image-generation, the goal of the discriminator is to distinguish between generated voxels from real voxels.

We adopt a similar architecture proposed in VoxelGAN [59], where G consists of 3D-deconvolutional layers and D consists of a stack of 3D-convolutional layers. To produce the spatial output as a feedback signal, the discriminator does not globally pool over the spatial dimension, resulting in a response cube of shape $H/c \times W/c \times Dep/c$. We use Wasserstein loss with gradient penalty for both VoxelGAN trained with and without feedback. The details of architectures and training can be found in the appendix.

5.2. Results

Image generation We train our model on CelebA dataset [36], consiting of over 100K celebrity faces with wide-range of attributes. We use a latent vector of dimension 128 to generate an image of size $128 \times 128 \times 3$. The discriminator outputs a response map of size 8×8. In Figure 4, we show images sampled with and without feedback adversarial learning. In Table 1, we compute the FID score [39] on the last feature layer.

Image-to-image translation We use both the Cityscapes [13] dataset and the NYU-depth-V2 dataset [42]. For Cityscapes, the goal of our network is to generate photos from class segmentation maps. We resize the images to 256×512. In Figure 5, we show qualitative results and in Table 2, we use an image segmentation model [11] to compute the segmentation score on the generated images. We also provide the LPIPS [65] distance from the ground truth image for both the training and validation set. The perceptual score, although indicative of the similarity between the generated image and the ground truth, may penalize images that look realistic but is perceptually different

For NYU-depth-V2, we train our model to generate indoor images. We combine the depth-map, coarse class label

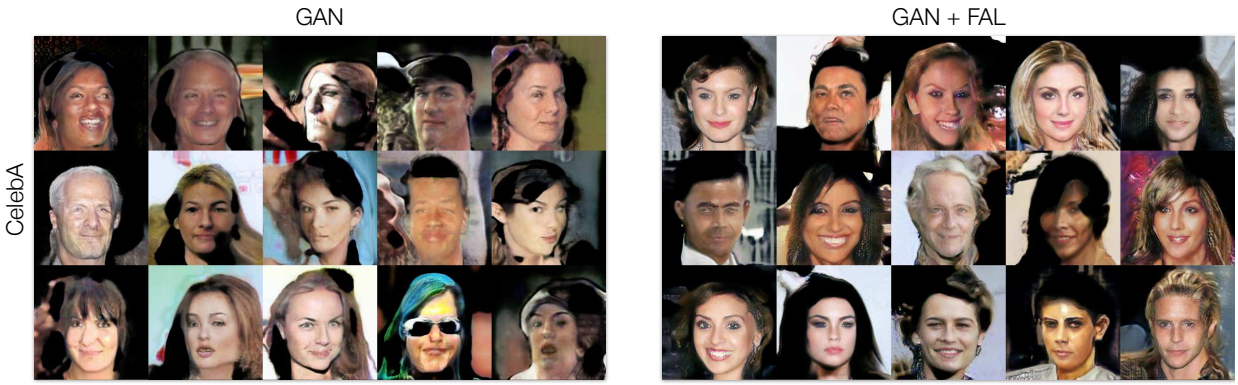

Figure 4: **Image Generation:** Results using feedback adversarial learning on 256×256 CelebA dataset. These images are randomly sampled from truncated $\mathcal{N}(0, I)$.

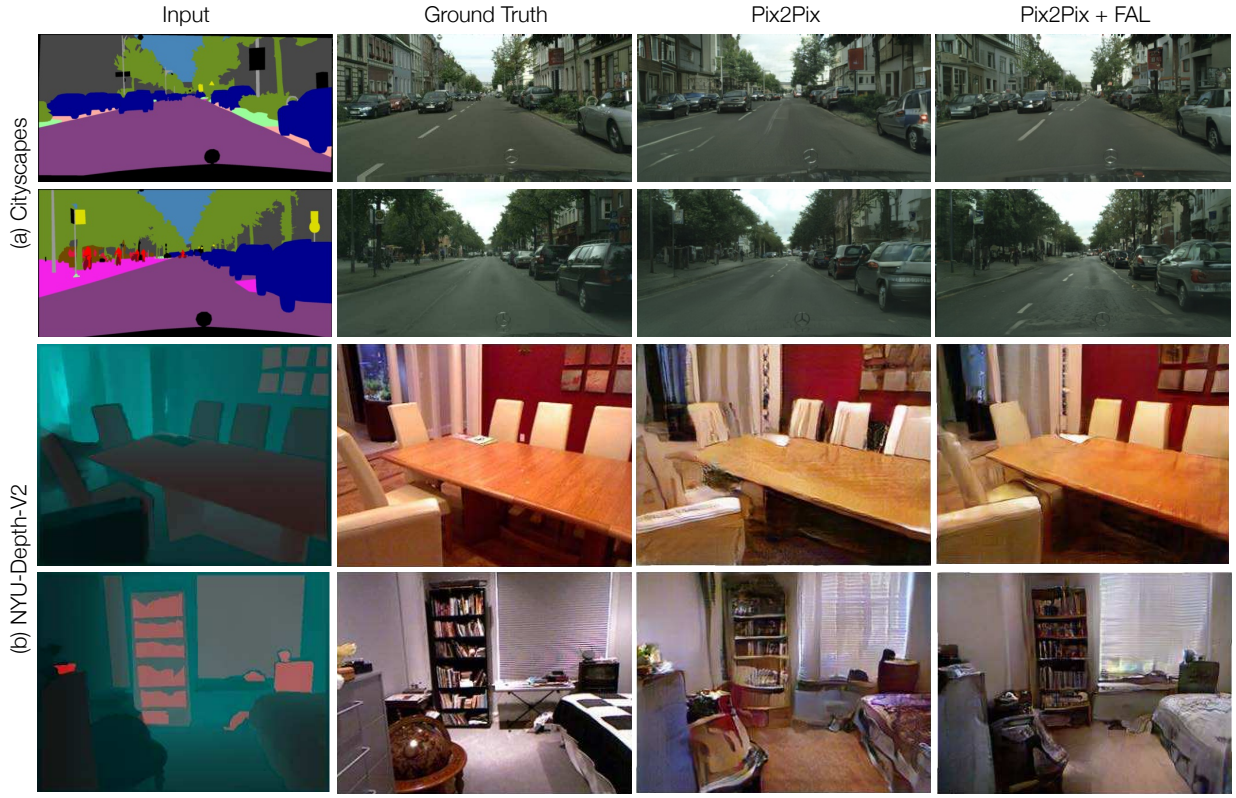

Figure 5: **Image-to-image translation:** Results using feedback adversarial learning. We train the models on 256×512 Cityscapes images that map segmentation map to photos. For NYU-depth-v2, the models are trained to map 240×360 depth, coarse-class, and edges to photo. We train our model with 3 generation steps and show our results on the last generation step.

map to construct a 2-channel input. To create this input data, we labeled the top 37 most occurring classes (out of around 1000 classes) and mapped the classes to the first input channel, where the classes are equidistant from each other. Next, we use the depth map as the second channel of the image. The resulting image is of size $240 \times 320 \times 2$. In Figure 5 we visualize our results, and on Table 3 we quantify our result using a network trained to predict depth from monocular

Model	CelebA-FID ↓
GAN	22.56
GAN w/ Feedback (t=1)	26.49
GAN w/ Feedback (t=2)	20.65
GAN w/ Feedback (t=3)	**18.52**

Table 1: **Image generation (CelebA):** FID score computed between the generated and real CelebA images. We compute the score using the pretrained Inception-V3 model. Lower FID score is better.

Model	Val Cat IOU ↑	Val Cls IOU ↑	Val LPIPS ↓	Train LPIPS ↓
Ground Truth	76.2	0.21	0.0	0.0
Pix2Pix	0.380	0.655	**0.428**	0.320
Pix2Pix + Feedback (t=1)	0.383	0.646	0.431	0.265
Pix2Pix + Feedback (t=2)	0.417	0.687	**0.428**	**0.254**
Pix2Pix + Feedback (t=3)	**0.418**	**0.692**	0.429	**0.254**

Table 2: **Image-to-image translation (Cityscapes):** We use a pretrained segmentation model trained on real images to compute the segmentation score. The pretrained model is trained on real image. We also provide LPIPS distance score using the generated and the ground truth image.

RGB image [2]. We also provide the LPIPS distance to the ground truth image.

Voxel Generation We train the VoxelGAN with and without Feedback on ShapeNet [10]. ShapeNet consists of a large number of synthetic objects where the voxels are generated from. We select three different object categories with varying numbers of voxels: airplane (4k), car (8k), and vessel (2k) and train a model for each category. The generator consists of 7 3D-deconvolutional layers, which produces $64 \times 64 \times 64$ voxels from a sampled latent vector of dimension 100. The discriminator consists of 6 3D-convolutional layers and outputs $4 \times 4 \times 4$ response cube.

To quantitatively evaluate the quality of the generated voxels, we train a voxel classifier – with the assumption that realistically generated voxels will have a higher classification probability, similar to the idea of Inception Score [49]. We use 10 object categories with a sufficient number of voxels to train a 10-way classifier. The trained classifier achieves an overall 95.9% accuracy on the testing set (95.9% on airplanes, 99.6% on cars, and 98.8% on vessels). The details on the voxel classifier can be found in the appendix.

We randomly sample 1k generated voxels and measure the accuracy of the voxels using the trained classifier. The quantitative results are shown in Table. 4 and the qualitative

Model	Val REL↓	Val δ_1 ↑	Val δ_2 ↑	Val LPIPS ↓	Train LPIPS ↓
Ground Truth	0.191	0.846	0.974	0.0	0.0
Pix2Pix	0.191	0.892	0.961	0.483	0.337
Pix2Pix + Feedback ($t=1$)	0.179	0.702	0.904	0.473	0.281
Pix2Pix + Feedback ($t=2$)	**0.178**	**0.706**	0.906	**0.469**	**0.275**
Pix2Pix + Feedback ($t=3$)	0.181	0.701	**0.908**	0.473	0.284

Table 3: **Image-to-image translation (NYU Depth):** Using a pre-trained monocular depth prediction model, we compute the scores on the generated images. The depth prediction model is trained on real images. We also provide LPIPS distance score using the generated and the ground truth image.

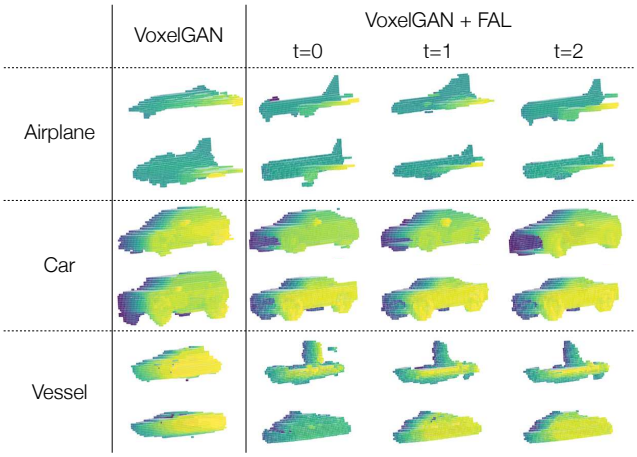

Figure 6: **Voxel generation (ShapeNet):**. Voxels are colorized based on the depth using the Viridis colormap. Our model is able to progressively generate better quality voxels with compared to the baseline.

results are shown in Figure. 6. We demonstrate that the VoxelGAN trained with feedback outperforms the baseline by progressively improving the generated voxels and achieving higher accuracy with more feedback steps. The classification accuracy on real testing voxels is shown in the first row.

5.3. Ablation Study

To investigate the essentials of utilizing both the discriminator response map and previous generation as feedback, we conduct an ablation study where only either of them is fed back to the generator. Furthermore, to verify the effectiveness of the proposed AST layer, we experiment with a variety of ways to merge the input feature h and feedback feature f. These ablation studies can be found in the appendix.

Model	Classification accuracy ↑		
	Airplane	Car	Vessel
Ground Truth	95.9%	99.6%	98.8%
VoxelGAN	93.0%	98.1%	89.2%
VoxelGAN + Feedback ($t = 1$)	93.0%	98.2%	91.0%
VoxelGAN + Feedback ($t = 2$)	94.0%	98.9%	96.2%
VoxelGAN + Feedback ($t = 3$)	**95.6%**	**99.1%**	**97.1%**

Table 4: **Voxel Classification Scores:** Classification score on the generated voxels. The accuracy measures whether the generated voxels are correctly classified to the category that the generator was trained from.

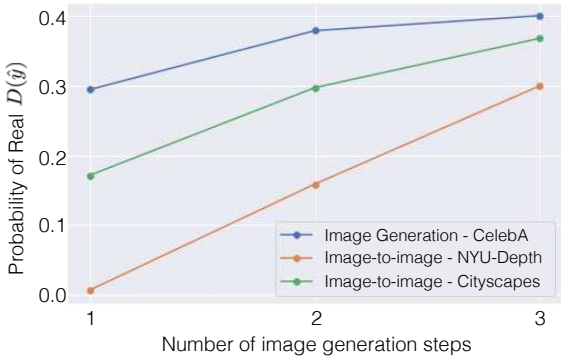

Figure 7: **Fooling likelihood over-time:** We plot the likelihood the discriminator believes the generated sample is correct. We show that the likelihood of fooling the discriminator increases over generation steps.

5.4. Discriminator response visualization

To visualize whether the generator can produce better results that can fool the discriminator, we visualize the discriminator's response across various generation time steps in Figure 7. In Figure 8, we plot the likelihood of fooling the discriminator across various generation time step.

5.5. Generalization to more generation steps

Although we trained our model with 3 generation steps, our model can progressively improve the generation quality with an increasing number of feedback steps. We quantify the output of the discriminator by taking the mean of the response. On Cityscapes, if we take the average of the discriminator output across the whole training set, we have the following fooling probabilities for 5 generation steps (in increasing order of generation): 17.1%, 28.7%, 36.8%, 40.3%, 43.4%. This illustrates that the generator has learned to continually leverage the discriminator output beyond the trained generation steps.

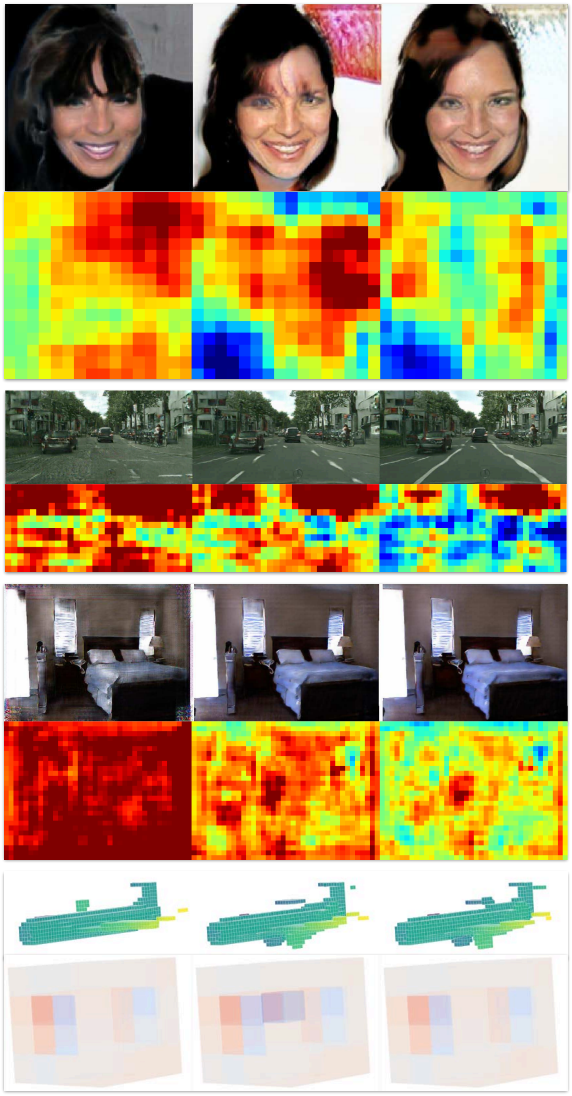

Figure 8: **Response visualization:** We visualize the output of the discriminator overtime for various datasets. **Red** indicating fake and **blue** indicating real. We show that the discriminator predicts more real regions over time.

6. Conclusion

We demonstrated that feedback adversarial learning – leveraging discriminator information into the feed-forward path of the generation process – is a simple yet effective method to improve existing generative adversarial frameworks. We demonstrated that our approach is not restricted to a specific domain by applying it to the tasks of image generation, image-to-image translation, and voxel generation. We extensively evaluated models trained with and without feedback on a variety of datasets with various metrics to verify the effectiveness of our proposed method.

References

[1] O. V. L. E. A. G. K. K. Aaron van den Oord, Nal Kalchbrenner. Conditional image generation with pixelcnn decoders. In *Advances in Neural Information Processing Systems*, 2016.

[2] I. Alhashim and P. Wonka. High quality monocular depth estimation via transfer learning. *arXiv e-prints*, abs/1812.11941, 2018.

[3] M. Arjovsky, S. Chintala, and L. Bottou. Wasserstein generative adversarial networks. In *International Conference on Machine Learning*, 2017.

[4] A. Bansal, S. Ma, D. Ramanan, and Y. Sheikh. Recycle-gan: Unsupervised video retargeting. In *European Conference on Computer Vision*, 2018.

[5] V. Belagiannis and A. Zisserman. Recurrent human pose estimation. In *IEEE International Conference on Automatic Face & Gesture Recognition*, 2017.

[6] S. Benaim and L. Wolf. One-sided unsupervised domain mapping. In *Advances in Neural Information Processing Systems*, 2017.

[7] K. Bousmalis, N. Silberman, D. Dohan, D. Erhan, and D. Krishnan. Unsupervised pixel-level domain adaptation with generative adversarial networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

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

[9] J. Carreira, P. Agrawal, K. Fragkiadaki, and J. Malik. Human pose estimation with iterative error feedback. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[10] A. X. Chang, T. Funkhouser, L. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015.

[11] L. Chen, G. Papandreou, F. Schroff, and H. Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv e-prints*, abs/1706.05587, 2017.

[12] Q. Chen and V. Koltun. Photographic image synthesis with cascaded refinement networks. In *International Conference on Computer Vision*, 2017.

[13] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele. The cityscapes dataset for semantic urban scene understanding. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[14] A. Dai, C. R. Qi, and M. Nießner. Shape completion using 3d-encoder-predictor cnns and shape synthesis. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[15] E. L. Denton, S. Chintala, A. Szlam, and R. Fergus. Deep generative image models using a laplacian pyramid of adversarial networks. In *Advances in Neural Information Processing Systems*, 2015.

[16] C. Donahue, J. McAuley, and M. Puckette. Synthesizing audio with gans, 2018.

[17] C. Donahue, J. McAuley, and M. Puckette. Adversarial audio synthesis. In *International Conference on Learning Representations*, 2019.

[18] 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*, 2014.

[19] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville. Improved training of wasserstein gans. In *Advances in Neural Information Processing Systems*, 2017.

[20] J. Guo, S. Lu, H. Cai, W. Zhang, Y. Yu, and J. Wang. Long text generation via adversarial training with leaked information. 2018.

[21] J. M. H. L. O. P. A. C. Harm de Vries, Florian Strub. Modulating early visual processing by language. In *Advances in Neural Information Processing Systems*, 2017.

[22] W.-N. Hsu, Y. Zhang, R. J. Weiss, Y.-A. Chung, Y. Wang, Y. Wu, and J. Glass. Disentangling correlated speaker and noise for speech synthesis via data augmentation and adversarial factorization. In *NIPS 2018 Interpretability and Robustness for Audio, Speech and Language Workshop*, 2018.

[23] X. Huang and S. Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *International Conference on Computer Vision*, 2017.

[24] D. J. Im, C. D. Kim, H. Jiang, and R. Memisevic. Generating images with recurrent adversarial networks. In *ICLR Workshop*, 2016.

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

[26] L. F.-F. Justin Johnson, Agrim Gupta. Image generation from scene graphs. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[27] A. Kanazawa, M. J. Black, D. W. Jacobs, and J. Malik. End-to-end recovery of human shape and pose. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[28] 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.

[29] D. P. Kingma and M. Welling. Auto-encoding variational bayes. In *International Conference on Learning Representations*, 2014.

[30] C. Li and M. Wand. Precomputed real-time texture synthesis with markovian generative adversarial networks. In *European Conference on Computer Vision*, 2016.

[31] J. Li, K. Xu, S. Chaudhuri, E. Yumer, H. Zhang, and L. Guibas. Grass: Generative recursive autoencoders for shape structures. *ACM Transactions on Graphics*, 2017.

[32] M. Liang and X. Hu. Recurrent convolutional neural network for object recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

[33] J. H. Lim and J. C. Ye. Geometric gan. *arXiv preprint arXiv:1705.02894*, 2017.

[34] M. Liu, T. Breuel, and J. Kautz. Unsupervised image-to-image translation networks. In *Advances in Neural Information Processing Systems*, 2017.

[35] M. Liu and O. Tuzel. Coupled generative adversarial networks. In *Advances in Neural Information Processing Systems*, 2016.

[36] Z. Liu, P. Luo, X. Wang, and X. Tang. Deep learning face attributes in the wild. In *International Conference on Computer Vision*, 2015.

[37] X. Mao, Q. Li, H. Xie, R. Y. Lau, and Z. Wang. Multi-class generative adversarial networks with the l2 loss function. *arXiv preprint arXiv:1611.04076*, 2016.

[38] X. Mao, Q. Li, H. Xie, R. Y. Lau, and Z. Wang. Multi-class generative adversarial networks with the l2 loss function. *arXiv preprint arXiv:1611.04076*, 2016.

[39] T. U.-B. N. Martin Heusel, Hubert Ramsauer. Gans trained by a two time-scale update rule converge to a local nash equilibrium. 2017.

[40] M. Mathieu, C. Couprie, and Y. LeCun. Deep multi-scale video prediction beyond mean square error. In *International Conference on Learning Representations*, 2016.

[41] L. Metz, B. Poole, D. Pfau, and J. Sohl-Dickstein. Unrolled generative adversarial networks. In *International Conference on Learning Representations*, 2017.

[42] P. K. Nathan Silberman, Derek Hoiem and R. Fergus. Indoor segmentation and support inference from rgbd images. In *European Conference on Computer Vision*, 2012.

[43] E. Park, J. Yang, E. Yumer, D. Ceylan, and A. C. Berg. Transformation-grounded image generation network for novel 3d view synthesis. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[44] T. Park, M.-Y. Liu, T.-C. Wang, and J.-Y. Zhu. Semantic image synthesis with spatially-adaptive normalization. *arXiv preprint arXiv:1903.07291*, 2019.

[45] E. Perez, F. Strub, H. De Vries, V. Dumoulin, and A. Courville. Film: Visual reasoning with a general conditioning layer. In *Association for the Advancement of Artificial Intelligence*, 2018.

[46] A. Radford, L. Metz, and S. Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. In *International Conference on Learning Representations*, 2016.

[47] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee. Generative adversarial text-to-image synthesis. In *International Conference on Machine Learning*, 2016.

[48] F. S.-L. Z.-M. Roey Mechrez, Itamar Talmi. The contextual loss. In *European Conference on Computer Vision*, 2018.

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

[50] T. Salimans and D. P. Kingma. Weight normalization: A simple reparameterization to accelerate training of deep neural networks. In *Advances in Neural Information Processing Systems*, 2016.

[51] A. Shrivastava, T. Pfister, O. Tuzel, J. Susskind, W. Wang, and R. Webb. Learning from simulated and unsupervised images through adversarial training. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[52] S.-H. Sun, M. Huh, Y.-H. Liao, N. Zhang, and J. J. Lim. Multi-view to novel view: Synthesizing novel views with self-learned confidence. In *European Conference on Computer Vision*, 2018.

[53] M. K.-Y. Y. Takeru Miyato, Toshiki Kataoka. Spectral normalization for generative adversarial networks. In *International Conference on Learning Representations*, 2018.

[54] D. Tran, R. Ranganath, and D. Blei. Hierarchical implicit models and likelihood-free variational inference. In *Advances in Neural Information Processing Systems 30*, 2017.

[55] C. Vondrick, H. Pirsiavash, and A. Torralba. Generating videos with scene dynamics. In *Advances in Neural Information Processing Systems*, 2016.

[56] R. Vuorio, S.-H. Sun, H. Hu, and J. J. Lim. Toward multimodal model-agnostic meta-learning. *arXiv preprint arXiv:1812.07172*, 2018.

[57] X. Wang, R. Girshick, A. Gupta, and K. He. Non-local neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[58] X. Wang and A. Gupta. Generative image modeling using style and structure adversarial networks. In *European Conference on Computer Vision*, 2016.

[59] J. Wu, C. Zhang, T. Xue, W. T. Freeman, and J. B. Tenenbaum. Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling. In *Advances in Neural Information Processing Systems*, 2016.

[60] Z. Yi, H. Zhang, and M. Gong. Dualgan: Unsupervised dual learning for image-to-image translation. In *International Conference on Computer Vision*, 2017.

[61] L. Yu, W. Zhang, J. Wang, and Y. Yu. Seqgan: Sequence generative adversarial nets with policy gradient. In *Association for the Advancement of Artificial Intelligence*, 2017.

[62] A. R. Zamir, T. Wu, L. Sun, W. B. Shen, J. Malik, and S. Savarese. Feedback network. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[63] H. Zhang, I. J. Goodfellow, D. N. Metaxas, and A. Odena. Self-attention generative adversarial networks. In *Neural Information Processing Systems*, 2018.

[64] H. Zhang, T. Xu, H. Li, S. Zhang, X. Huang, X. Wang, and D. N. Metaxas. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In *International Conference on Computer Vision*, 2017.

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

[66] Y. Zhou and T. L. Berg. Learning temporal transformations from time-lapse videos. In *European Conference on Computer Vision*, 2016.

[67] J.-Y. Zhu, P. Krähenbühl, E. Shechtman, and A. A. Efros. Generative visual manipulation on the natural image manifold. In *European Conference on Computer Vision*, 2016.

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