Collaborative GAN Sampling

Yuejiang Liu [*1] **Parth Kothari** [*1] **Alexandre Alahi** [1]

Abstract

Generative adversarial networks (GANs) have shown great promise in generating complex data such as images. A standard practice in GANs is to discard the discriminator after training and use only the generator for sampling. However, this loses valuable information of real data distribution learned by the discriminator. In this work, we propose a collaborative sampling scheme between the generator and discriminator for improved data generation. Guided by the discriminator, our approach refines generated samples through gradient-based optimization, shifting the generator distribution closer to the real data distribution. Additionally, we present a practical discriminator shaping method that can further improve the sample refinement process. Orthogonal to existing GAN variants, our proposed method offers a new degree of freedom in GAN sampling. We demonstrate its efficacy through experiments on synthetic data and image generation tasks.

1. Introduction

Generative adversarial networks (GAN) (Goodfellow et al., 2014) have achieved great success in various tasks such as image generation (Goodfellow et al., 2014; Radford et al., 2015; Brock et al., 2018), image editing (Ledig et al., 2017), image style transfer (Zhu et al., 2017; Isola et al., 2017; Karras et al., 2018) and representation learning (Chen et al., 2016). Despite these promising results, training GANs poses a challenge in practice. The gradients provided by the discriminator over time are prone to vanish (Salimans et al., 2016) or explode (Gulrajani et al., 2017), often leading to training instability. While a great amount of effort has been devoted to stabilize the GAN training (Wu et al., 2017; Karras et al., 2017; Gulrajani et al., 2017; Kodali et al., 2017; Miyato et al., 2018), obtaining a consistently good generative model remains an open question. In this work, we aim to better approximate the real distribution through an alternate approach - instead of changing the GAN training dynamics, our goal is to improve the model distribution during the sampling process at test time.

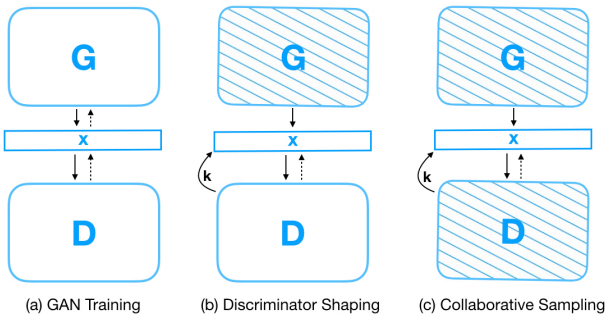


Figure 1. The GAN collaborative sampling framework. (a) The GAN is first trained. (b) Then, the generator parameters are frozen, whereas the discriminator loss landscape is further shaped if needed. (c) Finally, both the generator and discriminator are frozen. During the collaborative sampling process, samples are first produced by the generator and subsequently refined by the discriminator, leading to an improved model distribution.

A standard practice in GAN sampling is to completely discard the discriminator once training completes and use only the generator for sample generation. Recently, (Azadi et al., 2018) shows that it is beneficial to post-process the generator distribution by using the discriminator to reject bad samples. While this method, under strict conditions, can recover the real data distribution, it needs the generator distribution and real distribution to have the same support and suffers from low sampling efficiency. We propose to overcome these limitations and take a step further by modifying, rather than simply rejecting, the generated samples.

In this work, we introduce a collaborative sampling scheme between the generator and discriminator for improved sample generation. Once GAN training completes, we freeze the parameters of the generator and refine the generated samples leveraging on the discriminator gradients. We further propose to shape the discriminator loss landscape using these refined samples. Through sample-wise optimization, our method shifts the model distribution closer to the real data distribution.

[*]Equal contribution
[1]École Polytechnique Fédérale de Lausanne (EPFL). <{yuejiang.liu,parth.kothari,alexandre.alahi}@epfl.ch>.

Preprint. Source code is available at https://github.com/vita-epfl/collaborative-gan-sampling

We demonstrate the potential of our collaborative sampling scheme by conducting experiments on synthetic data, MNIST and CelebA dataset. Through quantitative and qualitative evaluation, we show that our proposed collaborative sampling method can consistently improve the generator distribution. Moreover, we show that in a challenging imbalance setting the standard GAN training easily runs into mode collapse, whereas our proposed framework succeeds in recovering all modes with high sample quality. We finally present empirical evidence that our method is not limited to the task of sample generation by successfully applying it to image denoising. Orthogonal to the extensive research works in tackling the GAN training dynamics, our proposed method is naturally compatible with different GAN variants and offers a new degree of freedom in GAN sampling.

2. Background

2.1. Generative Adversarial Networks

A generative adversarial network (GAN) consists of two neural networks, namely the generator G and the discriminator D, trained together. Informally, it is a two player game where the objective of the generator is to fool the discriminator into believing that the generated samples are real. On the other hand, the discriminator has to correctly classify whether the given sample is real or fake. The generator G takes as input a noise vector z sampled from a given noise distribution p_z and transforms it into a real looking sample $G(z)$. The G essentially maps the noise distribution p_z to a generator distribution p_g. The D tells whether a sample comes from the generator distribution p_g or the real data distribution p_r. The game between the generator G and the discriminator D can be formulated as a minimax objective:

$$\min_G \max_D \mathbb{E}_{x\sim p_r}[log(D(x))] + \mathbb{E}_{x'\sim p_g}[1 - log(D(x'))]. \tag{1}$$

It is shown in (Goodfellow et al., 2014) that under certain conditions, optimizing the above loss function can lead to a generator that exactly recovers the real data distribution. However, this assumption may not hold in practice (Arora et al., 2017), resulting in an inaccurate model distribution.

2.2. Training GANs

The training dynamics of GANs is in general very complex. The generator does not have direct access to the real data distribution and can only learn from the gradient information provided by the discriminator. However, these gradients are prone to vanish (Salimans et al., 2016) or explode (Gulrajani et al., 2017), resulting in notorious training instability. The original GAN (Goodfellow et al., 2014) advocated the use of a non-saturating loss function (NS-GAN) to mitigate the issue of vanishing gradients:

$$\begin{aligned} L_D &= -\mathbb{E}_{x\sim p_r}[\log D(x)] - \mathbb{E}_{z\sim p_z}[1 - \log D(G(z))] \\ L_G &= -\mathbb{E}_{z\sim p_z}[\log D(G(z))]. \end{aligned} \tag{2}$$

To further improve GAN training, recent works have been extensively focused on various training procedures (Karras et al., 2017; Chavdarova & Fleuret, 2017), loss functions (Wu et al., 2017), network architectures (Radford et al., 2015), regularization (Gulrajani et al., 2017; Kodali et al., 2017) and normalization (Miyato et al., 2018) methods. Despite improved stability, these techniques can overly constrain the discriminator, leading to a tradeoff between the training stability and algorithmic performance. Recent work (Brock et al., 2018) achieved the state-of-the-art results by relaxing these stabilizing conditions and allowing training collapse. Moreover, (Fedus et al., 2017; Lucic et al., 2017) have shown that, with proper hyperparameter tuning, the original non-saturating GAN (Eq. 2) can achieve comparable performance to the new variants. In this work, we sidestep these issues in GAN training. Rather, we accept the trained generator distribution at its face value and try to improve the generated sample quality in the sampling process.

2.3. GAN sampling

A standard GAN sampling process draws samples from the generator distribution without the involvement of the discriminator. Recently, (Azadi et al., 2018) proposed a rejection sampling scheme that uses the discriminator to filter out the generated samples that are unlikely to be real. A similar method is introduced in (Turner et al., 2018), which replaced the rejection sampling by the Markov chain Monte Carlo (MCMC) method for better scalability in high-dimensional space. However, both of them rely on an accept-reject principle and inevitably sacrifice sample efficiency. In fact, they have to modulate the rejection threshold (Azadi et al., 2018) or calibrate the discriminator (Turner et al., 2018) to mitigate undesired acceptance rate for practical use. Our work will explore a deeper collaboration scheme between the generator and discriminator that exploits richer information contained in the discriminator to improve the model distribution.

2.4. Collaborative Networks

Collaborative neural networks have been a growing concept and a useful tool to tackle challenging tasks which a single isolated scheme can hardly cope with. Various collaborative mechanisms have been proposed for image recognition (Zhang et al., 2011; Cai et al., 2016), image segmentation (Zheng et al., 2018) as well as recommendation system (Wang et al., 2015; He et al., 2017; Chen et al., 2017; Tay

et al., 2018).

In the context of GANs, (LeCun, 2016) promoted to go beyond adversarial training and explore collaborative generative networks. (Albanie et al., 2017) introduced a conceptually unadversarial paradigm where two networks work as a team to achieve the best performance. Very recently, (Lee et al., 2019) proposed a GAN-based image imputation method that leverages multiple inputs in a collaborative manner to generate realistic missing data. However, existing works lack a concrete and generic solution for close collaboration between the generator and discriminator in either training or sampling. In this work, we will present a collaborative sampling scheme that allows the two networks to jointly participate in the modeling of the real data distribution.

3. Method

A common practice in GANs is to discard the discriminator once the training completes, using only the generator for sample generation. We posit that, in contrast to disregarding the discriminator, it can be beneficial to leverage the information contained in the discriminator to refine generated samples. By doing this, we acquire an additional degree of freedom to shift the generator distribution towards the real one.

In this section, we will first introduce our proposed collaborative sampling scheme and then move on to explain why shaping the discriminator loss landscape can lead to a better refined sample distribution.

3.1. Collaborative Sampling

For the sake of simplicity, let x denote a sample drawn from the generator distribution, *i.e.*, $x \sim p_g$. The discriminator outputs $D(x)$ as an estimate of the probability of being real. We denote the parameters of the discriminator and the generator by θ_d and θ_g respectively. In standard GAN training, gradients are backpropagated through the generated sample x to parameters θ_g:

$$\frac{\partial L_G}{\partial \theta_g} = \frac{\partial L_G}{\partial x}\frac{\partial x}{\partial \theta_g},$$

where L_G is defined according to Eq. 2.

Ideally, we want the generator and discriminator to reach an equilibrium where the discriminator loses its ability to distinguish between p_g and p_r, *i.e.*, $D(x) = 1/2$ and $g_x := \frac{\partial L_G}{\partial x} = 0$. However, such a saddle point can be hardly reached in practice. It has been shown that even after sufficient training, the generator and discriminator may not converge to an equilibrium (Arora et al., 2017), indicating a scope of improvement for the generator distribution.

Practically, let's assume that generator distribution does not exactly match the real data distribution, *i.e.*, the divergence between p_g and p_r is not zero. Rather than sampling from this inexact generator distribution, we leverage the gradient information provided by the discriminator to continuously refine generated samples using the following recursion:

$$x^{k+1} = \mathit{Proj}_{\Omega}(x^k - \lambda_{x^k} g_x^k),$$

where k is the iteration index, λ is the stepsize, *Proj* is the projection operator and Ω is the constraints on x, *e.g.*, the range of valid pixel value. Inspired by (Rolinek & Martius, 2018), we adopt an adaptive stepsize based on the discriminator output:

$$\lambda_{x^k} = \eta(D(x^*) - D(x^k)), \qquad x^* \sim p_r,$$

where η is a constant hyperparameter of learning rate and x^* is a random sample drawn from the real data distribution. $D(x^*) - D(x^k)$ serves as an estimate of the distance between a refined sample and a nearest optimum. Thus, the step-size adaptation accelerates sample optimization for those samples that are distant from the real data distribution.

Algorithm 1 Collaborative sampling

Input : A frozen generator G, a frozen discriminator D, maximum number of refinement iterations K
Output : synthetic sample x
begin
  Randomly sample a real data x^* from data distribution p_r
  Randomly sample a fake data x^0 from generator distribution p_g
  for $k = 0, 1, \ldots, K-1$ **do**
    if $D(x^k) > D(x*)$ **then**
      break
    end
    g^k = GetGradient(D, x^k)
    x^{k+1} = UpdateSample(g^k, x^k)
  end
end

3.2. Discriminator Shaping

We refer to the function defined by the discriminator for classifying a sample as the learned discriminator loss function. Ideally, we want a smooth and monotonic discriminator loss function from the generated samples to their nearest real distribution modes in order to effectively guide the sample refinement process according to Algorithm 1. However, this is not always the case for the standard GAN training. Since the objective of the discriminator is solely to distinguish real and fake samples, the learned discriminator can dramatically overfit to the generator distribution. Consequently, the

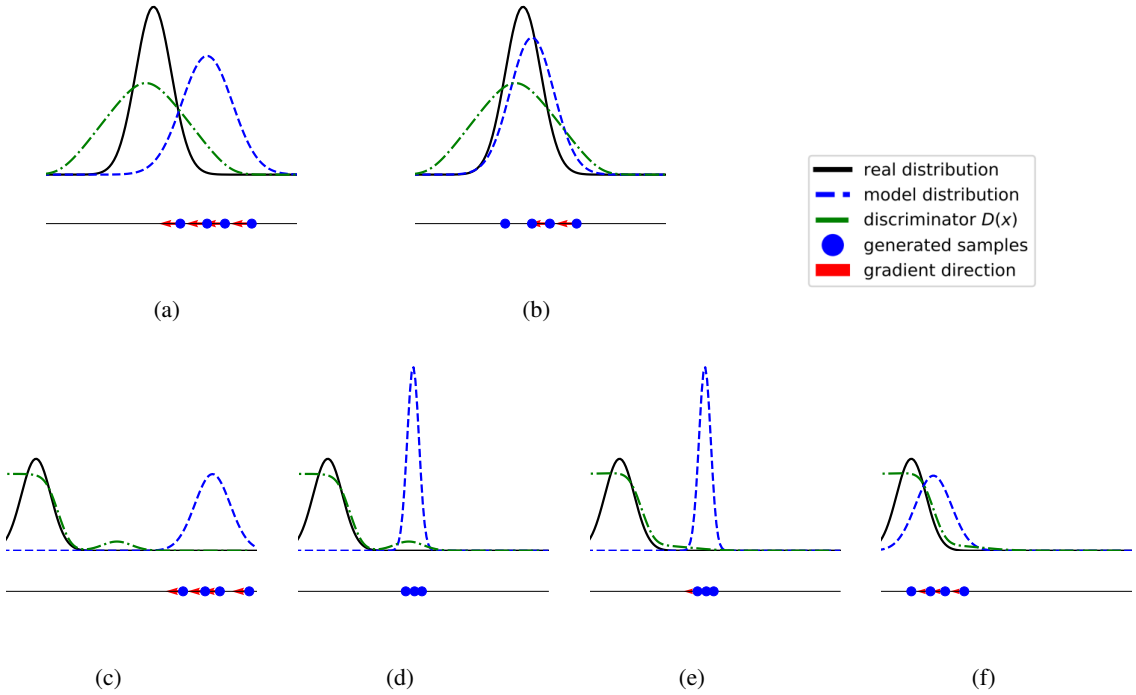


Figure 2. (a)-(b) illustrates the collaborative sampling scheme in the ideal scenario. The default samples from the generator distribution are refined using the gradient information from the discriminator to get the refined distribution. (c)-(f) Illustrates the collaborative sampling scheme in the non-ideal scenario. (c)-(d) Illustrates the refined samples getting stuck in a bad optimum due to highly complex discriminator loss landscape. (e) Discriminator shaping is performed to get rid of the bad optimum. (f) The new loss landscape helps to recover the real distribution from the generator distribution.

discriminator may misclassify a refined fake sample which it has never seen before and lose the ability to suggest improvements. In other words, when the learned discriminator has multiple local optima, it can fail in guiding the generated samples toward the real distribution. From here on, we refer to such local optima as 'bad local optima'.

To resolve this issue, we devise a practical discriminator shaping method, in hope to obtain a discriminator that is not only good at classification but also capable of effectively guiding the sample refinement process. Given a trained generator and discriminator, our method shapes the discriminator according to the following modified objective:

$$L_D = -\mathbb{E}_{x\sim p_r}[\log D(x)] - \mathbb{E}_{x'\sim p_c}[1 - \log D(x')], \quad (3)$$

where x' is a refined sample from Algorithm 1 and p_c is the collaboratively sampled data distribution.

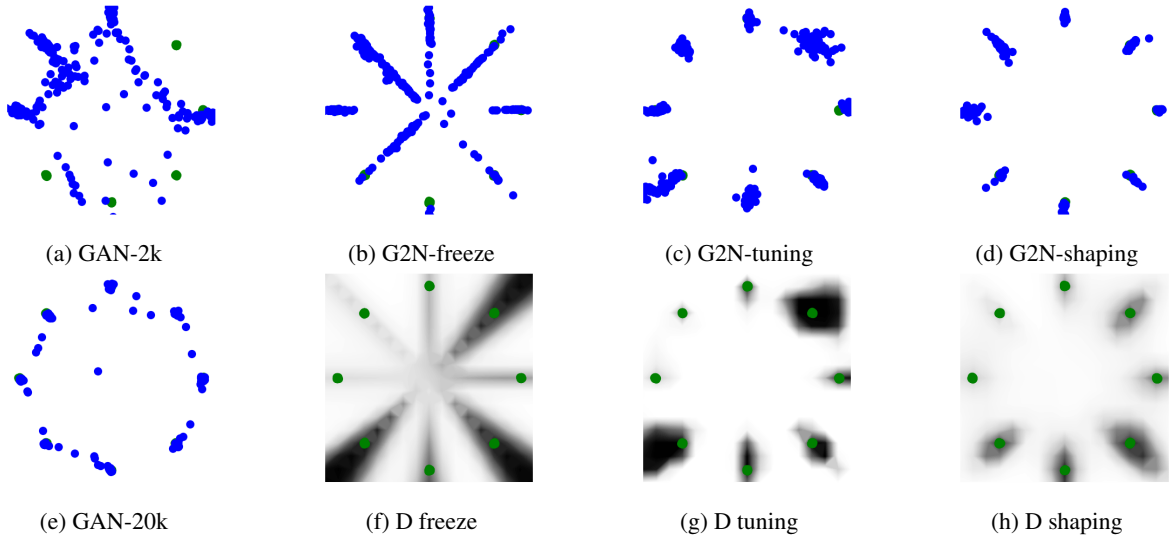
As outlined in Algorithm 2, the discriminator shaping step and sample refinement process are conducted alternatively, which essentially encourages the discriminator to explore the space spanning from the generation distribution p_g to the real data distribution p_r. By doing so, we aim to remove bad local optima in the discriminator loss landscape. It can alternatively be viewed as a self-supervision game, gradually pushing the discriminator to generalize and collaborate with the generator for sample improvement.

Algorithm 2 Discriminator shaping

Input : a frozen generator G, a pre-trained discriminator D
Output : a fine-tuned discriminator $\hat{D}$
begin
  for *number of D shaping iterations* **do**
    Draw m refined samples $\{x_c^{(1)}, \ldots, x_c^{(m)}\}$ from the collaborative data distribution $p_c(x)$ according to Algorithm 1
    Draw m real samples $\{x_r^{(1)}, \ldots, x_r^{(m)}\}$ from the real data distribution $p_r(x)$
    Update discriminator by minimizing the objective function Eq. (3)
  end
end

3.3. Discussion

We summarize our proposed collaborative sampling scheme through an illustration. In the ideal case shown in Fig. (2a)



(a) GAN-2k (b) G2N-freeze (c) G2N-tuning (d) G2N-shaping

(e) GAN-20k (f) D freeze (g) D tuning (h) D shaping

Figure 3. Comparison of various sampling schemes in the mixture of 8 Gaussians task. The GAN models are trained for $2k$ iterations in the top row and $20k$ iterations in (e). Green - Real samples. Blue - Samples obtained from trained GAN models using different sampling schemes. (a) Default generator samples. (b) Refined samples without additional discriminator training. (c) Refined samples with a discriminator fine-tuned using generator samples for $10k$ iterations. (d) Refined samples with a discriminator shaped using refined samples for $10k$ iterations. In (f),(g),(h), the background darkness is proportional to the $D(x)$, illustrating the loss landscape defined by the discriminator.

and Fig. (2b), a trained generator provides a model distribution that is close to, but not exactly the same as, the real data distribution. When the discriminator loss function is smooth and monotonic, the collaborative sampling scheme can easily shift the generator distribution closer to the real one. However, in higher dimensional space, the discriminator loss function can present numerous bad local optima. As shown in Fig. (2c) and Fig. (2d), the refined samples may end up at these bad local optima. Fig. (2e) and Fig. (2f) illustrate how recursively shaping discriminator loss landscape using the refined data distribution can help in recovering the real data distribution.

Propositions 1. *If the discriminator reaches its optimal D^* at each step given the model distribution, the sample refinement process $x_c = R(x_g)$ shifts the model distribution closer to the data distribution*

$$D_{JS}(p_r \parallel p_c) \leq D_{JS}(p_r \parallel p_g)$$

Proof. Consider each pair of generated sample x_g and refined sample x_c, we have $L_G(x_c) \leq L_G(x_g)$ and $D^*(x_c) \leq D^*(x_g)$ for objective functions such as the minimax or non-saturating loss. Applying this inequality to the value function provided by an optimal discriminator, we have

$$\begin{aligned} V(G, D^*) &= \mathbb{E}_{x\sim p_r}[\log D^*(x)] - \mathbb{E}_{x_g\sim p_g}[1 - \log D^*(x_g)] \\ &\geq \mathbb{E}_{x\sim p_r}[\log D^*(x)] - \mathbb{E}_{x_c\sim p_c}[1 - \log D^*(x_c)]. \end{aligned}$$

According to the Theorem 1 in the original GAN paper (Goodfellow et al., 2014), $V(G, D^*) = 2D_{JS}(p_r \parallel p_g) - log(4)$, concluding the proof. □

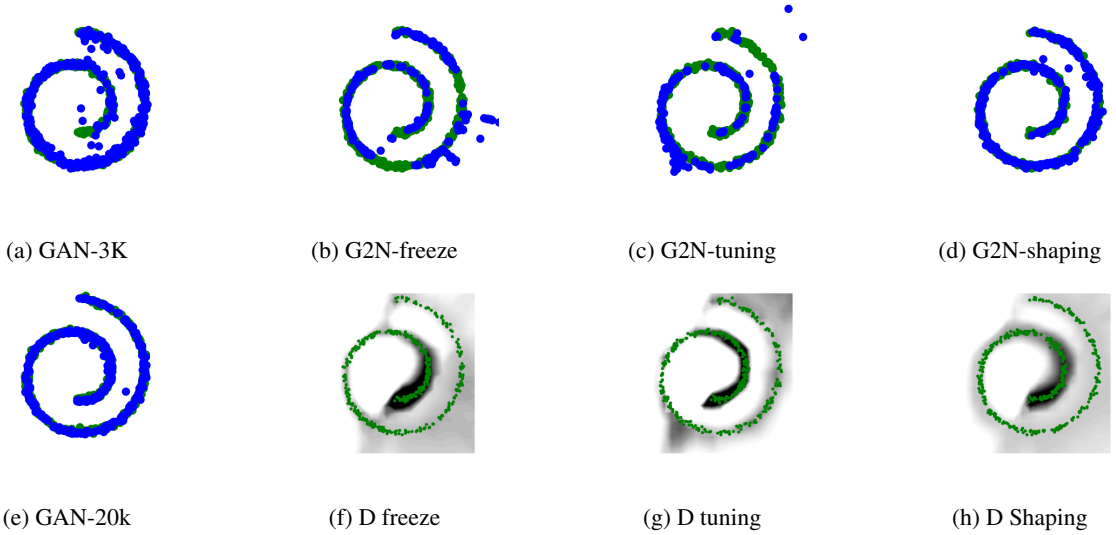
Remark 1. *The proposed collaborative sampling scheme can be naturally integrated with rejection sampling process (Azadi et al., 2018), thereby recovering the exact data distribution under ideal conditions. In particular, our sample refinement method can potentially boost the sample-efficiency of the downstream rejection sampling process.*

4. Experiments

In this section, we will compare our proposed collaborative sampling scheme with the standard GAN framework both qualitatively and quantitatively. First, we will analyze the performance of our method on standard 2D synthetic datasets: 2D Gaussians and Swissroll. Additionally, we will show the strength of our method in a challenging scenario of large data imbalance. Next, we will examine its effectiveness on image generation tasks, including MNIST and CelebA. We will finally show an application of the proposed scheme in image denoising.

4.1. 2D Synthetic Data

In this experiment, we examine our proposed collaborative sampling framework on two standard low-dimensional synthetic datasets typically used in the GAN community: Mixture of eight 2D isotropic Gaussians and Swissroll. These 2D experiments not only demonstrate the ability of our method for improved sample generation but also provide a



(a) GAN-3K (b) G2N-freeze (c) G2N-tuning (d) G2N-shaping

(e) GAN-20k (f) D freeze (g) D tuning (h) D Shaping

Figure 4. Comparison of various sampling schemes in the Swissroll task. The GAN models are trained for $3k$ iterations in the top row and $20k$ iterations in (e). Green - Real samples. Blue - Samples obtained from trained GAN models using different sampling schemes. (a) Default generator samples. (b) Refined samples without additional discriminator training. (c) Refined samples with a discriminator fine-tuned using generator samples for $5k$ iterations. (d) Refined samples with a discriminator shaped using refined samples for $5k$ iterations. In (f),(g),(h), the background darkness is proportional to the $D(x)$, illustrating the loss landscape defined by the discriminator.

clear visualization of the discriminator loss landscape.

The synthetic data is created in a range of 10.0 units with a standard deviation of 0.05 units for 2D Gaussians and 0.25 units for Swissroll. We use a standard fully-connected MLP with 4 hidden layers to model the generator and the discriminator. For sample refinement, we conduct maximum 50 updates (k=50) with a step size of 0.1. We compare the results of the following sample generation schemes in our experiments:

a. *GAN*: Standard sampling from GAN only using the trained generator

b. *G2N-freeze*: Collaborative sampling from both the generator and discriminator, without additional discriminator training

c. *G2N-tuning*: Collaborative sampling from both the generator and discriminator, with discriminator tuning using generated samples

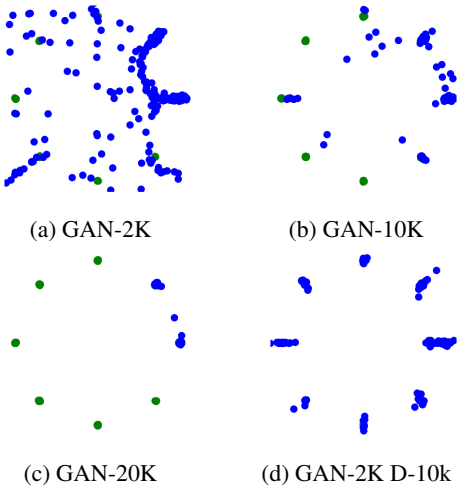
d. *G2N-shaping*: Collaborative sampling from both the generator and discriminator, with discriminator shaping using refined samples

All the schemes are applied on an early terminated GAN model ($2k$ iterations). Figure 3 and Figure 4 show a qualitative comparison of 400 samples produced by the different schemes on the 2D datasets. We can clearly observe that the default generator distribution produces a considerable amount of outliers far away from the real data distribution. Comparing different sample refinement schemes, our proposed G2N-shaping method provides the closest distribution to the real distribution on both two tasks. Furthermore, the loss landscape learned by G2N-shaping is better aligned with the real distribution. In contrast, G2N-freeze provides bad sample accuracy while G2N-tuning causes severe overfitting, misleading the sample refinement process. This result confirms that both the additional discriminator training as well as the use of refined samples for shaping the landscape are an integral part for the success of improved sample generation.

	8 Gaussians		Swissroll	
	Dist.	% Good	Dist.	% Good
GAN-early	0.98	0.23	0.26	0.97
G2N-freeze	1.99	0.07	0.41	0.93
G2N-tuning	0.60	0.36	3.66	0.83
G2N-shaping	**0.16**	**0.66**	**0.17**	**0.99**
GAN-optimal	0.18	0.76	0.16	0.99

Table 1. Quantitative comparison of different sampling schemes. A sample is '*Good*' if the distance between the generated sample and its nearest real sample is smaller than 3 times of the standard deviation. GAN-early are the early terminated GAN models corresponding to Fig. 3a and Fig. 4a, from which different sampling methods generate samples. GAN-optimal are near optimal GAN models trained for $20k$ iterations.

A quantitative comparison is summarized in Table 1. Following (Srivastava et al., 2017), we consider a sample as having high quality if it is within a distance of three times the standard deviation from the nearest real sample. The proposed collaborative sampling method significantly outperforms



(a) GAN-2K (b) GAN-10K

(c) GAN-20K (d) GAN-2K D-10k

Figure 5. Model comparison on imbalanced Gaussian mixture dataset. 80% of the real data (green) are drawn from the right and upper right Gaussian components, whereas the remaining other six clusters account for the rest 20% of data. The standard GAN models are trained for (a) $2k$, (b) $10k$, (c) $20k$ iterations. We observe that GAN training gradually runs into mode collapse. (d) The proposed collaborative sampling method is applied to a GAN model trained for $2k$ iterations. The resulting model distribution achieves highly competitive sample quality without losing any modes.

the standard GAN sampling, increasing the fraction of high-quality samples from 23% to 66% and from 97% to 99% in the Gaussian mixture and Swissroll respectively. We can also notice that without discriminator shaping, the sample refinement process cannot consistently improve the model distribution, which is in line with our qualitative observations above.

One might argue that training GANs for a longer period may recover an even better model distribution. To this extent, we provide an additional comparison between an early terminated generator with our proposed sampling method and a fully trained GAN for $20k$ iterations, which is roughly equivalent to the best performance a model can achieve within the given capacity, as shown in Fig. (3e) and Fig. (4e). The quantitative and qualitative results indicate the near-optimal performance of our method in both of the 2D tasks.

Based on the highly competitive performance on the ideal balanced datasets, we next present a more realistic and imbalanced scenario where standard GAN training is prone to mode collapse. Fig. 5 shows an experiment on an imbalanced mixture of Gaussians, where 80% of the data come from two Gaussian components while the rest 20% are distributed to the other six clusters. As shown in Fig. (5c), training GANs until convergence results in high sample quality but poor diversity. On the other hand, early termination fails to provide realistic samples. By combining our proposed collaborative sampling method with an early terminated GAN model, we can obtain samples of both high quality and diversity, as shown in Fig. (5d). This superior result suggests a high potential of our method in overcoming challenges that a standard GAN encounters.

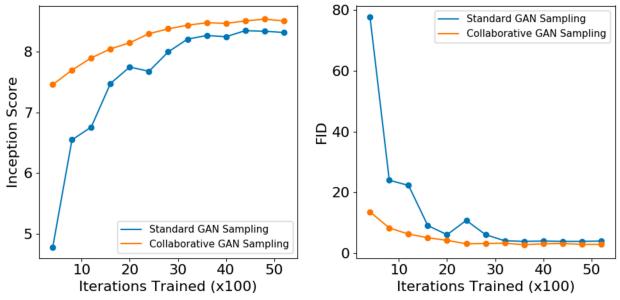


Figure 6. Quantitative comparison of the collaborative sampling scheme v/s default generator sampling. Higher score is better for IS and lower value is better for FID.

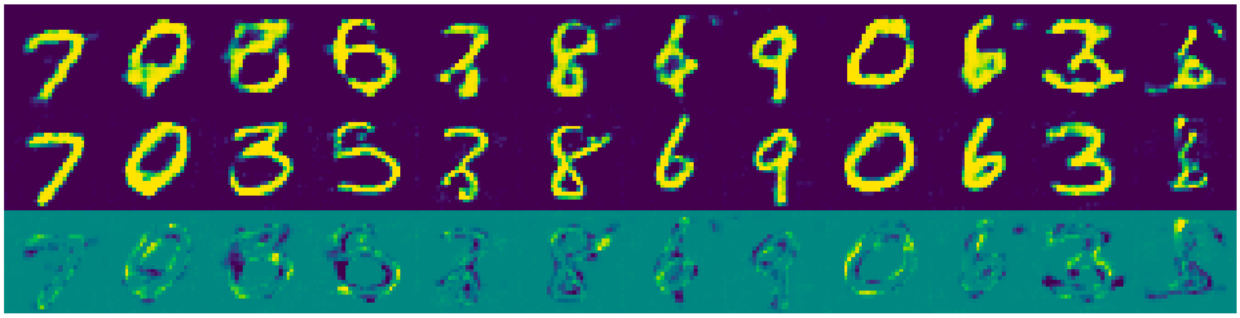
4.2. Image Generation

In our next experiment, we apply the proposed framework to the task of image generation. We first examine the effectiveness of our method on MNIST and CelebA datasets, and subsequently explore its application in denoising.

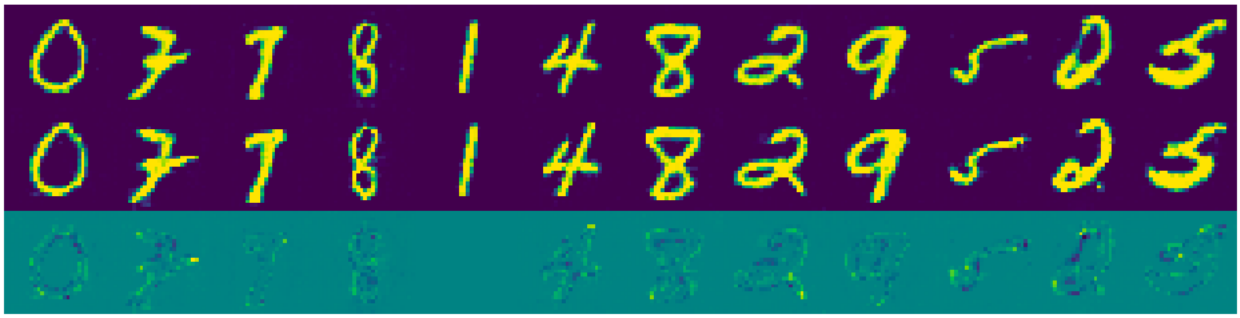
For MNIST, the generator network consists of two fully connected layers followed by two deconvolutional layers. The discriminator consists of two convolutional layers followed by two linear layers. All the layers have leaky ReLU activations except for the last layer which has tanh and sigmoid activation function for the generator and discriminator respectively. We train both the networks using the Adam optimizer (Kingma & Ba, 2014) with a learning rate of 0.0002.

We highlight the performance of the collaborative sampling scheme at two different stages of GAN training. When the GAN training is early terminated, as shown in Fig. (7a), the default generator distribution is far from a good approximation of the real distribution. By collaboratively sampling from this generator and the shaped discriminator, the refined samples become clearly better looking. In particular, the refined samples have much less digit errors as well as background noise. On the other hand, when the GAN is fully trained, as shown in Fig. (7b), both the default generator samples and the refined samples look good qualitatively. However, on careful observation, we notice that the refined distribution corrects subtle digit errors present in the default generator distribution, for example, the '2' digits.

A quantitative comparison between the standard GAN sampling and our proposed collaborative sampling on MNIST is presented in Fig. 6. Our proposed sampling scheme is applied to GAN models trained for various epochs. Given a frozen generator at each stage, the discriminator is shaped



(a) Early terminated NS-GAN after 1 epoch



(b) Well trained NS-GAN after 20 epoch

Figure 7. A comparison between the standard GAN sampling (top row) and the proposed collaborative sampling method (middle row) on MNIST. Image difference between the first two rows is presented in the bottom row. (a) The collaborative sampling method drastically improves the image quality when GAN training is early terminated after the 1st epoch. (b) When the generator is well trained for 20 epochs, the collaborative sampling method can still correct minor image errors, wherever necessary. For example, the digit '2'.

for $1k$ more iterations. The result shows that the proposed collaborative sampling method can consistently improve the IS (Salimans et al., 2016) and FID (Heusel et al., 2017). This fact adds credence to our hypothesis that the sample refinement process guided by the discriminator can offer another degree of freedom to improve the model distribution.

We further assess our method with DCGAN on CelebA dataset. We first train both networks using the Adam optimizer with a learning rate of 0.0002 for 5 epochs. Once GAN training stops, the discriminator is shaped for 2 more epochs. Qualitative results are illustrated in Fig 8. Clear improvements can be observed on most images, which reaffirms the high potential of our proposed method in refining complex data like human faces.

Our proposed framework is not limited to the task of improving sample generation. Fig. 9 shows an application of our method to image denoising. The model architecture is the same as the MNIST case. The input noisy samples are constructed by adding faint generator samples to the real data samples. Given these noisy inputs, we compare the outputs of the following two schemes: sample refinement without discriminator shaping and sample refinement using discriminator shaping. We can observe that the sample refinement process with the shaped discriminator greatly removes the noise on top of the real digits, whereas the refinement process with the default discriminator injects additional noise around the digits. This, once again, demonstrates that the discriminator shaping method helps to remove bad local optima in the learned loss landscape.

5. Conclusion and Discussion

In this work, we introduced a collaborative sampling scheme for GANs. Rather than disregarding the discriminator at test time, we propose to continue using the discriminator gradients to refine the generated samples. This is advantageous in cases where the generator distribution does not exactly match the real distribution, due to practical challenges such as training instability, mode collapse (Fig. 5) or limited model capacity (Fig. 7). Orthogonal to various works on GAN training, our proposed scheme offers another degree of freedom to improve the model distribution empowered by the discriminator.

The potential of our method is not limited to GAN sampling, as noted by its successful application to the task of image denoising (Fig. 9). We hope to further explore its potential in tackling mode collapse, training instability as well as other challenges in the future.

One limitation of the proposed method is the additional computational time for sampling. By introducing the sample-

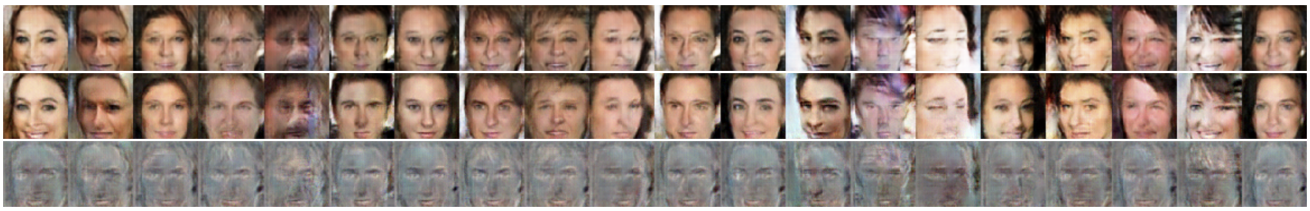


Figure 8. Performance of our collaborative sampling scheme on 64×64 CelebA. A DCGAN is trained for 5 epoch, followed by discriminator shaping for 2 epoch. The refined images (middle row) look more visually realistic than the default generated images (top row). The difference between the two set of samples are shown in the bottom row.

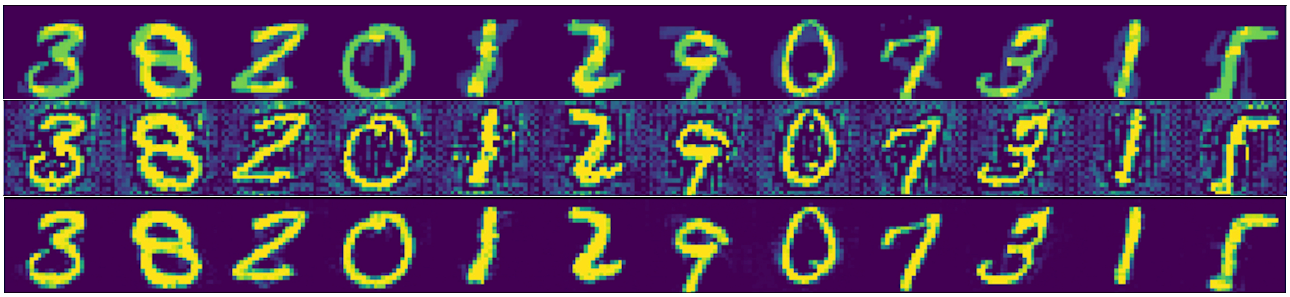


Figure 9. Application of our proposed method to image denoising. The real images are perturbed by other faint digits (top row). Directly applying the sample refinement with the default discriminator gives rise to background noise (middle row). Refining the noisy samples with the shaped discriminator successfully recovers the original images (bottom row).

wise optimization, we gain improved sample quality but at the expense of extra iterations. Finding a time-efficient optimization method for the sample refinement process is another potential avenue for future research.

References

Albanie, S., Ehrhardt, S., and Henriques, J. F. Stopping gan violence: Generative unadversarial networks. *arXiv preprint arXiv:1703.02528*, 2017.

Arora, S., Ge, R., Liang, Y., Ma, T., and Zhang, Y. Generalization and Equilibrium in Generative Adversarial Nets (GANs). *arXiv:1703.00573 [cs, stat]*, March 2017. URL http://arxiv.org/abs/1703.00573. arXiv: 1703.00573.

Azadi, S., Olsson, C., Darrell, T., Goodfellow, I., and Odena, A. Discriminator Rejection Sampling. *arXiv:1810.06758 [cs, stat]*, October 2018. URL http://arxiv.org/abs/1810.06758. arXiv: 1810.06758.

Brock, A., Donahue, J., and Simonyan, K. Large Scale GAN Training for High Fidelity Natural Image Synthesis. *arXiv:1809.11096 [cs, stat]*, September 2018. URL http://arxiv.org/abs/1809.11096. arXiv: 1809.11096.

Cai, S., Zhang, L., Zuo, W., and Feng, X. A probabilistic collaborative representation based approach for pattern classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2950–2959, 2016.

Chavdarova, T. and Fleuret, F. SGAN: An Alternative Training of Generative Adversarial Networks. *arXiv:1712.02330 [cs, stat]*, December 2017. URL http://arxiv.org/abs/1712.02330. arXiv: 1712.02330.

Chen, J., Zhang, H., He, X., Nie, L., Liu, W., and Chua, T.-S. Attentive collaborative filtering: Multimedia recommendation with item-and component-level attention. In *Proceedings of the 40th International ACM SIGIR conference on Research and Development in Information Retrieval*, pp. 335–344. ACM, 2017.

Chen, Y., Hoffman, M. W., Colmenarejo, S. G., Denil, M., Lillicrap, T. P., Botvinick, M., and de Freitas, N. Learning to Learn without Gradient Descent by Gradient Descent. *arXiv:1611.03824 [cs, stat]*, November 2016. URL http://arxiv.org/abs/1611.03824. arXiv: 1611.03824.

Fedus, W., Rosca, M., Lakshminarayanan, B., Dai, A. M., Mohamed, S., and Goodfellow, I. Many Paths to Equilibrium: GANs Do Not Need to Decrease a Divergence At Every Step. *arXiv:1710.08446 [cs, stat]*, October 2017. URL http://arxiv.org/abs/1710.08446. arXiv: 1710.08446.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio,

Y. Generative Adversarial Nets. In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N. D., and Weinberger, K. Q. (eds.), *Advances in Neural Information Processing Systems 27*, pp. 2672–2680. Curran Associates, Inc., 2014. URL http://papers.nips.cc/paper/5423-generative-adversarial-nets.pdf.

Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., and Courville, A. C. Improved Training of Wasserstein GANs. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 30*, pp. 5767–5777. Curran Associates, Inc., 2017. URL http://papers.nips.cc/paper/7159-improved-training-of-wasserstein-gans.pdf.

He, X., Liao, L., Zhang, H., Nie, L., Hu, X., and Chua, T.-S. Neural collaborative filtering. In *Proceedings of the 26th International Conference on World Wide Web*, pp. 173–182. International World Wide Web Conferences Steering Committee, 2017.

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., and Hochreiter, S. GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium. *arXiv:1706.08500 [cs, stat]*, June 2017. URL http://arxiv.org/abs/1706.08500. arXiv: 1706.08500.

Isola, P., Zhu, J., Zhou, T., and Efros, A. A. Image-to-Image Translation with Conditional Adversarial Networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5967–5976, July 2017. doi: 10.1109/CVPR.2017.632.

Karras, T., Aila, T., Laine, S., and Lehtinen, J. Progressive Growing of GANs for Improved Quality, Stability, and Variation. *arXiv:1710.10196 [cs, stat]*, October 2017. URL http://arxiv.org/abs/1710.10196. arXiv: 1710.10196.

Karras, T., Laine, S., and Aila, T. A Style-Based Generator Architecture for Generative Adversarial Networks. *arXiv:1812.04948 [cs, stat]*, December 2018. URL http://arxiv.org/abs/1812.04948. arXiv: 1812.04948.

Kingma, D. P. and Ba, J. Adam: A Method for Stochastic Optimization. *arXiv:1412.6980 [cs]*, December 2014. URL http://arxiv.org/abs/1412.6980. arXiv: 1412.6980.

Kodali, N., Abernethy, J., Hays, J., and Kira, Z. On Convergence and Stability of GANs. *arXiv:1705.07215 [cs]*, May 2017. URL http://arxiv.org/abs/1705.07215. arXiv: 1705.07215.

LeCun, Y. Generative Collaborative Networks. *Twitter*, 2016. URL https://twitter.com/boredyannlecun/status/791115429766766592?lang=en.

Ledig, C., Theis, L., Huszr, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J., Wang, Z., and Shi, W. Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 105–114, July 2017. doi: 10.1109/CVPR.2017.19.

Lee, D., Kim, J., Moon, W.-J., and Ye, J. C. Collagan: Collaborative GAN for Missing Image Data Imputation. *arXiv:1901.09764 [cs, stat]*, January 2019. URL http://arxiv.org/abs/1901.09764. arXiv: 1901.09764.

Lucic, M., Kurach, K., Michalski, M., Gelly, S., and Bousquet, O. Are GANs Created Equal? A Large-Scale Study. *arXiv:1711.10337 [cs, stat]*, November 2017. URL http://arxiv.org/abs/1711.10337. arXiv: 1711.10337.

Miyato, T., Kataoka, T., Koyama, M., and Yoshida, Y. Spectral Normalization for Generative Adversarial Networks. *arXiv:1802.05957 [cs, stat]*, February 2018. URL http://arxiv.org/abs/1802.05957. arXiv: 1802.05957.

Radford, A., Metz, L., and Chintala, S. Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. *arXiv:1511.06434 [cs]*, November 2015. URL http://arxiv.org/abs/1511.06434. arXiv: 1511.06434.

Rolinek, M. and Martius, G. L4: Practical loss-based stepsize adaptation for deep learning. *arXiv:1802.05074 [cs, stat]*, February 2018. URL http://arxiv.org/abs/1802.05074. arXiv: 1802.05074.

Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., and Chen, X. Improved Techniques for Training GANs. *arXiv:1606.03498 [cs]*, June 2016. URL http://arxiv.org/abs/1606.03498. arXiv: 1606.03498.

Srivastava, A., Valkov, L., Russell, C., Gutmann, M. U., and Sutton, C. VEEGAN: Reducing Mode Collapse in GANs using Implicit Variational Learning. *arXiv:1705.07761 [stat]*, May 2017. URL http://arxiv.org/abs/1705.07761. arXiv: 1705.07761.

Tay, Y., Anh Tuan, L., and Hui, S. C. Latent relational metric learning via memory-based attention for collaborative ranking. In *Proceedings of the 2018 World Wide Web*

Conference on World Wide Web, pp. 729–739. International World Wide Web Conferences Steering Committee, 2018.

Turner, R., Hung, J., Saatci, Y., and Yosinski, J. Metropolis-Hastings Generative Adversarial Networks. *arXiv:1811.11357 [cs, stat]*, November 2018. URL http://arxiv.org/abs/1811.11357. arXiv: 1811.11357.

Wang, H., Wang, N., and Yeung, D.-Y. Collaborative deep learning for recommender systems. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1235–1244. ACM, 2015.

Wu, J., Huang, Z., Thoma, J., Acharya, D., and Van Gool, L. Wasserstein Divergence for GANs. *arXiv:1712.01026 [cs]*, December 2017. URL http://arxiv.org/abs/1712.01026. arXiv: 1712.01026.

Zhang, L., Yang, M., and Feng, X. Sparse representation or collaborative representation: Which helps face recognition? In *Computer vision (ICCV), 2011 IEEE international conference on*, pp. 471–478. IEEE, 2011.

Zheng, H., Xie, L., Ni, T., Zhang, Y., Wang, Y.-F., Tian, Q., Fishman, E. K., and Yuille, A. L. Phase collaborative network for multi-phase medical imaging segmentation. *arXiv preprint arXiv:1811.11814*, 2018.

Zhu, J.-Y., Park, T., Isola, P., and Efros, A. A. Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks. *arXiv:1703.10593 [cs]*, March 2017. URL http://arxiv.org/abs/1703.10593. arXiv: 1703.10593.