arXiv:2001.06937v1 [cs.LG] 20 Jan 2020

A Review on Generative Adversarial Networks: Algorithms, Theory, and Applications

Jie Gui, Zhenan Sun, Yonggang Wen, Dacheng Tao, Jieping Ye

Abstract—Generative adversarial networks (GANs) are a hot research topic recently. GANs have been widely studied since 2014, and a large number of algorithms have been proposed. However, there is few comprehensive study explaining the connections among different GANs variants, and how they have evolved. In this paper, we attempt to provide a review on various GANs methods from the perspectives of algorithms, theory, and applications. Firstly, the motivations, mathematical representations, and structure of most GANs algorithms are introduced in details. Furthermore, GANs have been combined with other machine learning algorithms for specific applications, such as semi-supervised learning, transfer learning, and reinforcement learning. This paper compares the commonalities and differences of these GANs methods. Secondly, theoretical issues related to GANs are investigated. Thirdly, typical applications of GANs in image processing and computer vision, natural language processing, music, speech and audio, medical field, and data science are illustrated. Finally, the future open research problems for GANs are pointed out.

Index Terms—Deep Learning, Generative Adversarial Networks, Algorithm, Theory, Applications.

1 Introduction

GENERATIVE adversarial networks (GANs) have become a hot research topic recently. Yann LeCun, a legend in deep learning, said in a Quora post "GANs are the most interesting idea in the last 10 years in machine learning." There are a large number of papers related to GANs according to Google scholar. For example, there are about 11,800 papers related to GANs in 2018. That is to say, there are about 32 papers everyday and more than one paper every hour related to GANs in 2018.

GANs consist of two models: a generator and a discriminator. These two models are typically implemented by neural networks, but they can be implemented with any form of differentiable system that maps data from one space to the other. The generator tries to capture the distribution of true examples for new data example generation. The discriminator is usually a binary classifier, discriminating generated examples from the true examples as accurately as possible. The optimization of GANs is a minimax optimization problem. The optimization terminates at a saddle point that is a minimum with respect to the generator and a maximum with respect to the discriminator. That is, the optimization goal is to reach Nash equilibrium [1]. Then, the generator can be thought to have captured the real distribution of true examples.

Some previous work has adopted the concept of making two neural networks compete with each other. The most relevant work is predictability minimization [2]. The connections between predictability minimization and GANs can be found in [3], [4].

The popularity and importance of GANs have led to several previous reviews. The difference from previous work is summarized in the following.

1) *GANs for specific applications*: There are surveys of using GANs for specific applications such as image synthesis and editing [5], audio enhancement and synthesis [6].
2) *General survey*: The earliest relevant review was probably the paper by Wang et al. [7] which mainly introduced the progress of GANs before 2017. References [8], [9] mainly introduced the progress of GANs prior to 2018. The reference [10] introduced the architecture-variants and loss-variants of GANs only related to computer vision. Other related work can be found in [11]–[13].

As far as we know, this paper is the first to provide a comprehensive survey on GANs from the algorithm, theory, and application perspectives which introduces the latest progress. Furthermore, our paper focuses on applications not only to image processing and computer vision, but also sequential data such as natural language processing, and other related areas such as medical field.

The remainder of this paper is organized as follows: The related work is discussed in Section 2. Sections 3-5 introduce GANs from the algorithm, theory, and applications perspectives, respectively. Tables 1 and 2 show GANs' algorithms and applications which will be discussed in Sections 3 and 5, respectively. The open research problems are discussed in Section 6 and Section 7 concludes the survey.

2 Related work

GANs belong to generative algorithms. Generative algorithms and discriminative algorithms are two categories of

J. Gui is with the Department of Computational Medicine and Bioinformatics, University of Michigan, USA (e-mail: guijie@ustc.edu).
Z. Sun is with the Center for Research on Intelligent Perception and Computing, Chinese Academy of Sciences, Beijing 100190, China (e-mail: znsun@nlpr.ia.ac.cn).
Y. Wen is with the School of Computer Science and Engineering, Nanyang Technological University, Singapore 639798 (e-mail: ygwen@ntu.edu.sg).
D. Tao is with the UBTECH Sydney Artificial Intelligence Center and the School of Information Technologies, the Faculty of Engineering and Information Technologies, the University of Sydney, Australia. (e-mail: Dacheng.Tao@Sydney.edu.au).
J. Ye is with DiDi AI Labs, P.R. China and University of Michigan, Ann Arbor.(e-mail:jpye@umich.edu).

TABLE 1: A overview of GANs' algorithms discussed in Section 3

GANs' Representative variants		InfoGAN [14], cGANs [15], CycleGAN [16], f-GAN [17], WGAN [18], WGAN-GP [19], LS-GAN [20]
GANs training	Objective function	LSGANs [21], [22], hinge loss based GAN [23]–[25], MDGAN [26], unrolled GAN [27], SN-GANs [23], RGANs [28]
	Skills	ImprovedGANs [29], AC-GAN [30]
	Structure	LAPGAN [31], DCGANs [32], PGGAN [33], StackedGAN [34], SAGAN [35], BigGANs [36], StyleGAN [37], hybrids of autoencoders and GANs (EBGAN [38], BEGAN [39], BiGAN [40]/ALI [41], AGE [42]), multi-discriminator learning (D2GAN [43], GMAN [44]), multi-generator learning (MGAN [45], MAD-GAN [46]), multi-GAN learning (CoGAN [47])
Task driven GANs	Semi-supervised learning	CatGANs [48], feature matching GANs [29], VAT [49], Δ-GAN [50], Triple-GAN [51]
	Transfer learning	DANN [52], CycleGAN [53], DiscoGAN [54], DualGAN [55], StarGAN [56], CyCADA [57], ADDA [58], [59], FCAN [60], unsupervised pixel-level domain adaptation (PixelDA) [61]
	Reinforcement learning	GAIL [62]

TABLE 2: Applications of GANs discussed in Section 5

Field	Subfield	Method
Image processing and computer vision	Super-resolution	SRGAN [63], ESRGAN [64], Cycle-in-Cycle GANs [65], SRDGAN [66], TGAN [67]
	Image synthesis and manipulation	DR-GAN [68], TP-GAN [69], PG^2 [70], PSGAN [71], APDrawingGAN [72], IGAN [73], introspective adversarial networks [74], GauGAN [75]
	Texture synthesis	MGAN [76], SGAN [77], PSGAN [78]
	Object detection	Segan [79], perceptual GAN [80], MTGAN [81]
	Video	VGAN [82], DRNET [83], Pose-GAN [84], video2video [85], MoCoGan [86]
Sequential data	Natural language processing (NLP)	RankGAN [87], IRGAN [88], [89], TAC-GAN [90]
	Music	RNN-GAN (C-RNN-GAN) [91], ORGAN [92], SeqGAN [93], [94]

machine learning algorithms. If a machine learning algorithm is based on a fully probabilistic model of the observed data, this algorithm is generative. Generative algorithms have become more popular and important due to their wide practical applications.

2.1 Generative algorithms

Generative algorithms can be classified into two classes: explicit density model and implicit density model.

2.1.1 Explicit density model

An explicit density model assumes the distribution and utilizes true data to train the model containing the distribution or fit the distribution parameters. When finished, new examples are produced utilizing the learned model or distribution. The explicit density models include maximum likelihood estimation (MLE), approximate inference [95], [96], and Markov chain method [97]–[99]. These explicit density models have an explicit distribution, but have limitations. For instance, MLE is conducted on true data and the parameters are updated directly based on the true data, which leads to an overly smooth generative model. The generative model learned by approximate inference can only approach the lower bound of the objective function rather than directly approach the objective function, because of the difficulty in solving the objective function. The Markov chain algorithm can be used to train generative models, but it is computationally expensive. Furthermore, the explicit density model has the problem of computational tractability. It may fail to represent the complexity of true data distribution and learn the high-dimensional data distributions [100].

2.1.2 Implicit density model

An implicit density model does not directly estimate or fit the data distribution. It produces data instances from the distribution without an explicit hypothesis [101] and utilizes the produced examples to modify the model. Prior to GANs, the implicit density model generally needs to be trained utilizing either ancestral sampling [102] or Markov chain-based sampling, which is inefficient and limits their practical applications. GANs belong to the directed implicit density model category. A detailed summary and relevant papers can be found in [103].

2.1.3 The comparison between GANs and other generative algorithms

GANs were proposed to overcome the disadvantages of other generative algorithms. The basic idea behind adversarial learning is that the generator tries to create as realistic examples as possible to deceive the discriminator. The discriminator tries to distinguish fake examples from true examples. Both the generator and discriminator improve through adversarial learning. This adversarial process gives GANs notable advantages over other generative algorithms. More specifically, GANs have advantages over other generative algorithms as follows:

1) GANs can parallelize the generation, which is impossible for other generative algorithms such as

PixelCNN [104] and fully visible belief networks (FVBNs) [105], [106].
2) The generator design has few restrictions.
3) GANs are subjectively thought to produce better examples than other methods.

Refer to [103] for more detailed discussions about this.

2.2 Adversarial idea

The adversarial idea has been successfully applied to many areas such as machine learning, artificial intelligence, computer vision and natural language processing. The recent event that AlphaGo [107] defeats world's top human player engages public interest in artificial intelligence. The intermediate version of AlphaGo utilizes two networks competing with each other.

Adversarial examples [108]–[117] have the adversarial idea, too. Adversarial examples are those examples which are very different from the real examples, but are classified into a real category very confidently, or those that are slightly different than the real examples, but are classified into a wrong category. This is a very hot research topic recently [112], [113]. To be against adversarial attacks [118], [119], references [120], [121] utilize GANs to conduct the right defense.

Adversarial machine learning [122] is a minimax problem. The defender, who builds the classifier that we want to work correctly, is searching over the parameter space to find the parameters that reduce the cost of the classifier as much as possible. Simultaneously, the attacker is searching over the inputs of the model to maximize the cost.

The adversarial idea exists in adversarial networks, adversarial learning, and adversarial examples. However, they have different objectives.

3 ALGORITHMS

In this section, we first introduce the original GANs. Then, the representative variants, training, evaluation of GANs, and task-driven GANs are introduced.

3.1 Generative Adversarial Nets (GANs)

The GANs framework is straightforward to implement when the models are both neural networks. In order to learn the generator's distribution p_g over data x, a prior on input noise variables is defined as $p_z(z)$ [3] and z is the noise variable. Then, GANs represent a mapping from noise space to data space as $G(z, \theta_g)$, where G is a differentiable function represented by a neural network with parameters θ_g. Other than G, the other neural network $D(x, \theta_d)$ is also defined with parameters θ_d and the output of $D(x)$ is a single scalar. $D(x)$ denotes the probability that x was from the data rather than the generator G. The discriminator D is trained to maximize the probability of giving the correct label to both training data and fake samples generated from the generator G. G is trained to minimize $\log(1 - D(G(z)))$ simultaneously .

3.1.1 Objective function

Different objective functions can be used in GANs.

3.1.1.1 Original minimax game:
The objective function of GANs [3] is

$$\min_G \max_D V(D,G) = E_{x\sim p_{data}(x)}[\log D(x)] + E_{z\sim p_z(z)}[\log(1 - D(G(z)))]. \quad (1)$$

$\log D(x)$ is the cross-entropy between $[1 \quad 0]^T$ and $[D(x) \quad 1 - D(x)]^T$. Similarly, $\log(1 - D(G(z)))$ is the cross-entropy between $[0 \quad 1]^T$ and $[D(G(z)) \quad 1 - D(G(z))]^T$. For fixed G, the optimal discriminator D is given by [3]:

$$D_G^*(x) = \frac{p_{data}(x)}{p_{data}(x) + p_g(x)}. \quad (2)$$

The minmax game in (1) can be reformulated as:

$$\begin{aligned} C(G) &= \max_D V(D,G) \\ &= E_{x\sim p_{data}}[\log D_G^*(x)] + E_{z\sim p_z}[\log(1 - D_G^*(G(z)))] \\ &= E_{x\sim p_{data}}[\log D_G^*(x)] + E_{x\sim p_g}[\log(1 - D_G^*(x))] \\ &= E_{x\sim p_{data}}\left[\log \frac{p_{data}(x)}{\frac{1}{2}(p_{data}(x)+p_g(x))}\right] + E_{x\sim p_g}\left[\frac{p_g(x)}{\frac{1}{2}(p_{data}(x)+p_g(x))}\right] - 2\log 2. \end{aligned} \quad (3)$$

The definition of KullbackLeibler (KL) divergence and Jensen-Shannon (JS) divergence between two probabilistic distributions $p(x)$ and $q(x)$ are defined as

$$KL(p\|q) = \int p(x) \log \frac{p(x)}{q(x)} dx, \quad (4)$$

$$JS(p\|q) = \frac{1}{2}KL(p\|\frac{p+q}{2}) + \frac{1}{2}KL(q\|\frac{p+q}{2}). \quad (5)$$

Therefore, (3) is equal to

$$\begin{aligned} C(G) &= KL(p_{data}\|\tfrac{p_{data}+p_g}{2}) + KL(p_g\|\tfrac{p_{data}+p_g}{2}) - 2\log 2 \\ &= 2JS(p_{data}\|p_g) - 2\log 2. \end{aligned} \quad (6)$$

Thus, the objective function of GANs is related to both KL divergence and JS divergence.

3.1.1.2 Non-saturating game:
It is possible that the Equation (1) cannot provide sufficient gradient for G to learn well in practice. Generally speaking, G is poor in early learning and samples are clearly different from the training data. Therefore, D can reject the generated samples with high confidence. In this situation, $\log(1 - D(G(z)))$ saturates. We can train G to maximize $\log(D(G(z)))$ rather than minimize $\log(1 - D(G(z)))$. The cost for the generator then becomes

$$\begin{aligned} J^{(G)} &= E_{z\sim p_z(z)}[-\log(D(G(z)))] \\ &= E_{x\sim p_g}[-\log(D(x))]. \end{aligned} \quad (7)$$

This new objective function results in the same fixed point of the dynamics of D and G but provides much larger gradients early in learning. The non-saturating game is heuristic, not being motivated by theory. However, the non-saturating game has other problems such as unstable numerical gradient for training G. With optimal D_G^*, we have

$$\begin{aligned} &E_{x\sim p_g}[-\log(D_G^*(x))] + E_{x\sim p_g}[\log(1 - D_G^*(x))] \\ &= E_{x\sim p_g}\left[\log \frac{(1-D_G^*(x))}{D_G^*(x)}\right] = E_{x\sim p_g}\left[\log \frac{p_g(x)}{p_{data}(x)}\right] \\ &= KL(p_g\|p_{data}). \end{aligned} \quad (8)$$

Therefore, $E_{x\sim p_g}\left[-\log\left(D_G^*\left(x\right)\right)\right]$ is equal to

$$\begin{aligned} &E_{x\sim p_g}\left[-\log\left(D_G^*\left(x\right)\right)\right] \\ &= KL(p_g\|p_{data}) - E_{x\sim p_g}\left[\log\left(1-D_G^*\left(x\right)\right)\right]. \end{aligned} \tag{9}$$

From (3) and (6), we have

$$\begin{aligned} &E_{x\sim p_{data}}\left[\log D_G^*\left(x\right)\right] + E_{x\sim p_g}\left[\log\left(1-D_G^*\left(x\right)\right)\right] \\ &= 2JS(p_{data}\|p_g) - 2\log 2. \end{aligned} \tag{10}$$

Therefore, $E_{x\sim p_g}\left[\log\left(1-D_G^*\left(x\right)\right)\right]$ equals

$$\begin{aligned} &E_{x\sim p_g}\left[\log\left(1-D_G^*\left(x\right)\right)\right] \\ &= 2JS(p_{data}\|p_g) - 2\log 2 - E_{x\sim p_{data}}\left[\log D_G^*\left(x\right)\right]. \end{aligned} \tag{11}$$

By substituting (11) into (9), (9) reduces to

$$\begin{aligned} &E_{x\sim p_g}\left[-\log\left(D_G^*\left(x\right)\right)\right] \\ &= KL(p_g\|p_{data}) - 2JS(p_{data}\|p_g) + \\ &E_{x\sim p_{data}}\left[\log D_G^*\left(x\right)\right] + 2\log 2. \end{aligned} \tag{12}$$

From (12), we can see that the optimization of the alternative G loss in the non-saturating game is contradictory because the first term aims to make the divergence between the generated distribution and the real distribution as small as possible while the second term aims to make the divergence between these two distributions as large as possible due to the negative sign. This will bring unstable numerical gradient for training G. Furthermore, KL divergence is not a symmetrical quantity, which is reflected from the following two examples

- If $p_{data}\left(x\right) \to 0$ and $p_g\left(x\right) \to 1$, we have $KL(p_g\|p_{data}) \to +\infty$.
- If $p_{data}\left(x\right) \to 1$ and $p_g\left(x\right) \to 0$, we have $KL(p_g\|p_{data}) \to 0$.

The penalizations for two errors made by G are completely different. The first error is that G produces implausible samples and the penalization is rather large. The second error is that G does not produce real samples and the penalization is quite small. The first error is that the generated samples are inaccurate while the second error is that generated samples are not diverse enough. Based on this, G prefers producing repeated but safe samples rather than taking risk to produce different but unsafe samples, which has the mode collapse problem.

3.1.1.3 Maximum likelihood game:

There are many methods to approximate (1) in GANs. Under the assumption that the discriminator is optimal, minimizing

$$\begin{aligned} J^{(G)} &= E_{z\sim p_z(z)}\left[-\exp\left(\sigma^{-1}\left(D\left(G\left(z\right)\right)\right)\right)\right] \\ &= E_{z\sim p_z(z)}\left[-D\left(G\left(z\right)\right)/\left(1-D\left(G\left(z\right)\right)\right)\right], \end{aligned} \tag{13}$$

where σ is the logistic sigmoid function, equals minimizing (1) [123]. The demonstration of this equivalence can be found in Section 8.3 of [103]. Furthermore, there are other possible ways of approximating maximum likelihood within the GANs framework [17]. A comparison of original zero-sum game, non-saturating game, and maximum likelihood game is shown in Fig. 1.

Three observations can be obtained from Fig. 1.

- First, when the sample is possible to be fake, that is on the left end of the figure, both the maximum likelihood game and the original minimax game suffer from gradient vanishing. The heuristically motivated non-saturating game does not have this problem.
- Second, maximum likelihood game also has the problem that almost all of the gradient is from the right end of the curve, which means that a rather small number of samples in each minibatch dominate the gradient computation. This demonstrates that variance reduction methods could be an important research direction for improving the performance of GANs based on maximum likelihood game.
- Third, the heuristically motivated non-saturating game has lower sample variance, which is the possible reason that it is more successful in real applications.

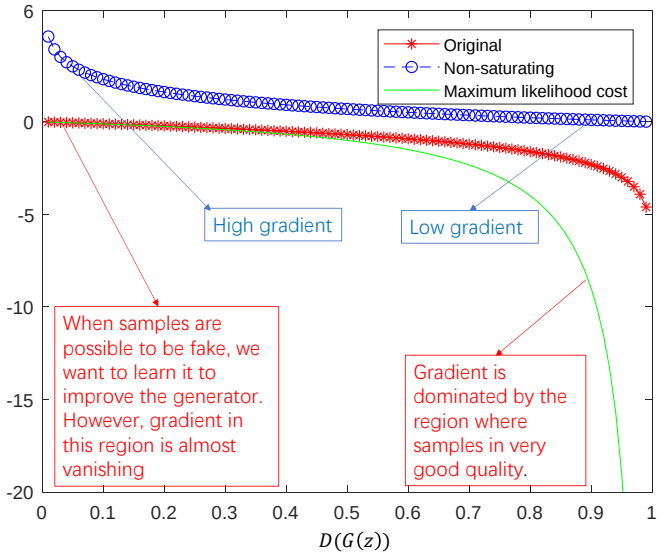


Fig. 1: The three curves of "Original", "Non-saturating", and "Maximum likelihood cost" denotes $\log\left(1-D\left(G\left(z\right)\right)\right)$, $-\log\left(D\left(G\left(z\right)\right)\right)$, and $-D(G(z))/(1-D(G(z)))$ in (1), (7), and (13), respectively. The cost that the generator has for generating a sample $G(z)$ is only decided by the discriminator's response to that generated sample. The larger probability the discriminator gives the real label to the generated sample, the less cost the generator gets. This figure is reproduced from [103], [123].

GAN Lab [124] is proposed as the interactive visualization tool designed for non-experts to learn and experiment with GANs. Bau et al. [125] present an analytic framework to visualize and understand GANs.

3.2 GANs' representative variants

There are many papers related to GANs [126]–[131] such as CSGAN [132] and LOGAN [133]. In this subsection, we will introduce GANs' representative variants.

3.2.1 InfoGAN

Rather than utilizing a single unstructured noise vector z, InfoGAN [14] proposes to decompose the input noise vector into two parts: z, which is seen as incompressible noise; c, which is called the latent code and will target the significant

structured semantic features of the real data distribution. InfoGAN [14] aims to solve

$$\min_G \max_D V_I(D,G) = V(D,G) - \lambda I(c; G(z,c)), \quad (14)$$

where $V(D,G)$ is the objective function of original GAN, $G(z,c)$ is the generated sample, I is the mutual information, and λ is the tunable regularization parameter. Maximizing $I(c;G(z,c))$ means maximizing the mutual information between c and $G(z,c)$ to make c contain as much important and meaningful features of the real samples as possible. However, $I(c;G(z,c))$ is difficult to optimize directly in practice since it requires access to the posterior $P(c|x)$. Fortunately, we can have a lower bound of $I(c;G(z,c))$ by defining an auxiliary distribution $Q(c|x)$ to approximate $P(c|x)$. The final objective function of InfoGAN [14] is

$$\min_G \max_D V_I(D,G) = V(D,G) - \lambda L_I(c;Q), \quad (15)$$

where $L_I(c;Q)$ is the lower bound of $I(c;G(z,c))$. InfoGAN has several variants such as causal InfoGAN [134] and semi-supervised InfoGAN (ss-InfoGAN) [135].

3.2.2 Conditional GANs (cGANs)

GANs can be extended to a conditional model if both the discriminator and generator are conditioned on some extra information y. The objective function of conditional GANs [15] is:

$$\begin{aligned}\min_G \max_D V(D,G) = &E_{x\sim p_{data}(x)}[\log D(x|y)] \\ &+E_{z\sim p_z(z)}[\log(1-D(G(z|y)))].\end{aligned} \quad (16)$$

By comparing (15) and (16), we can see that the generator of InfoGAN is similar to that of cGANs. However, the latent code c of InfoGAN is not known, and it is discovered by training. Furthermore, InfoGAN has an additional network Q to output the conditional variables $Q(c|x)$.

Based on cGANs, we can generate samples conditioning on class labels [30], [136], text [34], [137], [138], bounding box and keypoints [139]. In [34], [140], text to photo-realistic image synthesis is conducted with stacked generative adversarial networks (SGAN) [141]. cGANs have been used for convolutional face generation [142], face aging [143], image translation [144], synthesizing outdoor images having specific scenery attributes [145], natural image description [146], and 3D-aware scene manipulation [147]. Chrysos et al. [148] proposed robust cGANs. Thekumparampil et al. [149] discussed the robustness of conditional GANs to noisy labels. Conditional CycleGAN [16] uses cGANs with cyclic consistency. Mode seeking GANs (MSGANs) [150] proposes a simple yet effective regularization term to address the mode collapse issue for cGANs.

The discriminator of original GANs [3] is trained to maximize the log-likelihood that it assigns to the correct source [30]:

$$\begin{aligned}L = &E[\log P(S=real|X_{real})] \\ &+E[\log(P(S=fake|X_{fake}))],\end{aligned} \quad (17)$$

which is equal to (1). The objective function of the auxiliary classifier GAN (AC-GAN) [30] has two parts: the loglikelihood of the correct source, L_S, and the loglikelihood of the correct class label, L_C. L_S is equivalent to L in (17). L_C is defined as

$$\begin{aligned}L_C = &E[\log P(C=c|X_{real})] \\ &+E[\log(P(C=c|X_{fake}))].\end{aligned} \quad (18)$$

The discriminator and generator of AC-GAN is to maximize $L_C + L_S$ and $L_C - L_S$, respectively. AC-GAN was the first variant of GANs that was able to produce recognizable examples of all the ImageNet [151] classes.

Discriminators of most cGANs based methods [31], [41], [152]–[154] feed conditional information y into the discriminator by simply concatenating (embedded) y to the input or to the feature vector at some middle layer. cGANs with projection discriminator [155] adopts an inner product between the condition vector y and the feature vector.

Isola et al. [156] used cGANs and sparse regularization for image-to-image translation. The corresponding software is called pix2pix. In GANs, the generator learns a mapping from random noise z to $G(z)$. In contrast, there is no noise input in the generator of pix2pix. A novelty of pix2pix is that the generator of pix2pix learns a mapping from an observed image y to output image $G(y)$, for example, from a grayscale image to a color image. The objective of cGANs in [156] can be expressed as

$$\begin{aligned}L_{cGANs}(D,G) = &E_{x,y}[\log D(x,y)] \\ &+E_y[\log(1-D(y,G(y)))].\end{aligned} \quad (19)$$

Furthermore, l_1 distance is used:

$$L_{l_1}(G) = E_{x,y}[\|x - G(y)\|_1]. \quad (20)$$

The final objective of [156] is

$$L_{cGANs}(D,G) + \lambda L_{l_1}(G), \quad (21)$$

where λ is the free parameter. As a follow-up to pix2pix, pix2pixHD [157] used cGANs and feature matching loss for high-resolution image synthesis and semantic manipulation. With the discriminators, the learning problem is a multi-task learning problem:

$$\min_G \max_{D_1,D_2,D_3} \sum_{k=1,2,3} L_{GAN}(G,D_k). \quad (22)$$

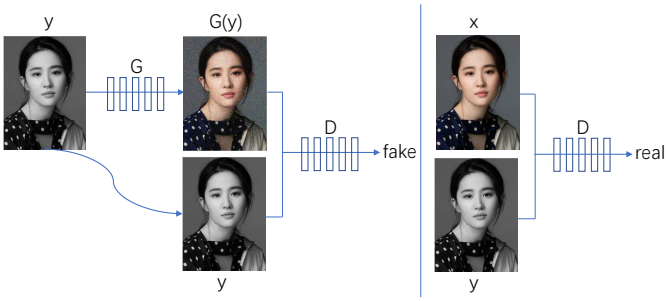


Fig. 2: Illustration of pix2pix: Training a conditional GANs to map grayscale→color. The discriminator D learns to classify between real grayscale, color tuples and fake (synthesized by the generator). The generator G learns to fool the discriminator. Different from the original GANs, both the generator and discriminator observe the input grayscale image and there is no noise input for the generator of pix2pix.

The training set is given as a set of pairs of corresponding images $\{(s_i, x_i)\}$, where x_i is a natural photo and s_i is a corresponding semantic label map. The ith-layer feature extractor of discriminator D_k is denoted as $D_k^{(i)}$ (from input to the ith layer of D_k). The feature matching loss $L_{FM}(G, D_k)$ is:

$$L_{FM}(G, D_k) = E_{(s,x)} \sum_{i=1}^{T} \frac{1}{N_i} \left[\left\| D_k^{(i)}(s, x) - D_k^{(i)}(s, G(s)) \right\|_1 \right], \quad (23)$$

where N_i is the number of elements in each layer and T denotes the total number of layers. The final objective function of [157] is

$$\min_G \max_{D_1, D_2, D_3} \sum_{k=1,2,3} \left(L_{GAN}(G, D_k) + \lambda L_{FM}(G, D_k) \right). \quad (24)$$

3.2.3 CycleGAN

Image-to-image translation is a class of graphics and vision problems where the goal is to learn the mapping between an output image and an input image using a training set of aligned image pairs. When paired training data is available, reference [156] can be used for these image-to-image translation tasks. However, reference [156] can not be used for unpaired data (no input/output pairs), which was well solved by Cycle-consistent GANs (CycleGAN) [53]. CycleGAN is an important progress for unpaired data. It is proved that cycle-consistency is an upper bound of the conditional entropy [158]. CycleGAN can be derived as a special case within the proposed variational inference (VI) framework [159], naturally establishing its relationship with approximate Bayesian inference methods.

The basic idea of DiscoGAN [54] and CycleGAN [53] is nearly the same. Both of them were proposed separately nearly at the same time. The only difference between CycleGAN [53] and DualGAN [55] is that DualGAN uses the loss format advocated by Wasserstein GAN (WGAN) rather than the sigmoid cross-entropy loss used in CycleGAN.

3.2.4 f-GAN

As we know, Kullback-Leibler (KL) divergence measures the difference between two given probability distributions. A large class of assorted divergences are the so called Ali-Silvey distances, also known as the f-divergences [160]. Given two probability distributions P and Q which have, respectively, an absolutely continuous density function p and q with regard to a base measure dx defined on the domain X, the f-divergence is defined,

$$D_f(P \| Q) = \int_X q(x) f\left(\frac{p(x)}{q(x)} \right) dx. \quad (25)$$

Different choices of f recover popular divergences as special cases of f-divergence. For example, if $f(a) = a \log a$, f-divergence becomes KL divergence. The original GANs [3] is a special case of f-GAN [17] which is based on f-divergence. The reference [17] shows that any f-divergence can be used for training GAN. Furthermore, the reference [17] discusses the advantages of different choices of divergence functions on both the quality of the produced generative models and training complexity. Im et al. [161] quantitatively evaluated GANs with divergences proposed for training. Uehara et al. [162] extend the f-GAN further, where the f-divergence is directly minimized in the generator step and the ratio of the distributions of real and generated data are predicted in the discriminator step.

3.2.5 Integral Probability Metrics (IPMs)

Denoting $\mathcal{P}$ the set of all Borel probability measures on a topological space $(M, \mathcal{A})$. The integral probability metric (IPM) [163] between two probability distributions $P \in \mathcal{P}$ and $Q \in \mathcal{P}$ is defined as

$$\gamma_{\mathcal{F}}(P, Q) = \sup_{f \in \mathcal{F}} \left| \int_M f dP - \int_M f dQ \right|, \quad (26)$$

where $\mathcal{F}$ is a class of real-valued bounded measurable functions on M. Nonparametric density estimation and convergence rates for GANs under Besov IPM Losses is discussed in [164]. IPMs include such as RKHS-induced maximum mean discrepancy (MMD) as well as the Wasserstein distance used in Wasserstein GANs (WGAN).

3.2.5.1 Maximum Mean Discrepancy (MMD):
The maximum mean discrepancy (MMD) [165] is a measure of the difference between two distributions P and Q given by the supremum over a function space $\mathcal{F}$ of differences between the expectations with regard to two distributions. The MMD is defined by:

$$MMD(\mathcal{F}, P, Q) = \sup_{f \in \mathcal{F}} \left(E_{X \sim P}[f(X)] - E_{Y \sim Q}[f(Y)] \right). \quad (27)$$

MMD has been used for deep generative models [166]–[168] and model criticism [169].

3.2.5.2 Wasserstein GAN (WGAN):
WGAN [18] conducted a comprehensive theoretical analysis of how the Earth Mover (EM) distance behaves in comparison with popular probability distances and divergences such as the total variation (TV) distance, the Kullback-Leibler (KL) divergence, and the Jensen-Shannon (JS) divergence utilized in the context of learning distributions. The definition of the EM distance is

$$W(p_{data}, p_g) = \inf_{\gamma \in \Pi(p_{data}, p_g)} E_{(x,y) \in \gamma} [\|x - y\|], \quad (28)$$

where $\Pi(p_{data}, p_g)$ denotes the set of all joint distributions $\gamma(x, y)$ whose marginals are p_{data} and p_g, respectively. However, the infimum in (28) is highly intractable. The reference [18] uses the following equation to approximate the EM distance

$$\max_{w \in \mathcal{W}} E_{x \sim p_{data}(x)}[f_w(x)] - E_{z \sim p_z(z)}[f_w(G(z))], \quad (29)$$

where there is a parameterized family of functions $\{f_w\}_{w \in \mathcal{W}}$ that are all K-Lipschitz for some K and f_w can be realized by the discriminator D. When D is optimized, (29) denotes the approximated EM distance. Then the aim of G is to minimize (29) to make the generated distribution as close to the real distribution as possible. Therefore, the overall objective function of WGAN is

$$\min_G \max_{w \in \mathcal{W}} E_{x \sim p_{data}(x)}[f_w(x)] - E_{z \sim p_z(z)}[f_w(G(z))] = \min_G \max_D E_{x \sim p_{data}(x)}[D(x)] - E_{z \sim p_z(z)}[D(G(z))]. \quad (30)$$

By comparing (1) and (30), we can see three differences between the objective function of original GANs and that of WGAN:

- First, there is no log in the objective function of WGAN.
- Second, the D in original GANs is utilized as a binary classifier while D utilized in WGAN is to approximate the Wasserstein distance, which is a regression task. Therefore, the sigmoid in the last layer of D is not used in the WGAN. The output of the discriminator of the original GANs is between zero and one while there is no constraint for that of WGAN.
- Third, the D in WGAN is required to be K-Lipschitz for some K and therefore WGAN uses weight clipping.

Compared with traditional GANs training, WGAN can improve the stability of learning and provide meaningful learning curves useful for hyperparameter searches and debugging. However, it is a challenging task to approximate the K-Lipschitz constraint which is required by the Wasserstein-1 metric. WGAN-GP [19] is proposed by utilizing gradient penalty for restricting K-Lipschitz constraint and the objective function is

$$\begin{aligned} L = &-E_{x\sim p_{data}}\left[D\left(x\right)\right] + E_{\tilde{x}\sim p_g}\left[D\left(\tilde{x}\right)\right] \\ &+\lambda E_{\hat{x}\sim p_{\hat{x}}}\left[\left(\left\|\nabla_{\hat{x}} D\left(\hat{x}\right)\right\|_2 - 1\right)^2\right] \end{aligned} \tag{31}$$

where the first two terms are the objective function of WGAN and $\hat{x}$ is sampled from the distribution $p_{\hat{x}}$ which samples uniformly along straight lines between pairs of points sampled from the real data distribution p_{data} and the generated distribution p_g. There are some other methods closely related to WGAN-GP such as DRAGAN [170]. Wu et al. [171] propose a novel and relaxed version of Wasserstein-1 metric: Wasserstein divergence (W-div), which does not require the K-Lipschitz constraint. Based on W-div, Wu et al. [171] introduce a Wasserstein divergence objective for GANs (WGAN-div), which can faithfully approximate W-div by optimization. CramerGAN [172] argues that the Wasserstein distance leads to biased gradients, suggesting the Cramr distance between two distributions. Other papers related to WGAN can be found in [173]–[178].

3.2.6 Loss Sensitive GAN (LS-GAN)

Similar to WGAN, LS-GAN [20] also has a Lipschitz constraint. It is assumed in LS-GAN that p_{data} lies in a set of Lipschitz densities with a compact support. In LS-GAN , the loss function $L_\theta\left(x\right)$ is parameterized with θ and LS-GAN assumes that a generated sample should have larger loss than a real one. The loss function can be trained to satisfy the following constraint:

$$L_\theta\left(x\right) \leq L_\theta\left(G\left(z\right)\right) - \Delta\left(x, G\left(z\right)\right) \tag{32}$$

where $\Delta\left(x, G\left(z\right)\right)$ is the margin measuring the difference between generated sample $G(z)$ and real sample x. The objective function of LS-GAN is

$$\begin{aligned} &\min_D \mathcal{L}_D = E_{x\sim p_{data}(x)}\left[L_\theta\left(x\right)\right] \\ &+\lambda E_{\substack{x\sim p_{data}(x), \\ z\sim p_z(z)}}\left[\Delta\left(x, G\left(z\right)\right) + L_\theta\left(x\right) - L_\theta\left(G\left(z\right)\right)\right]_+, \end{aligned} \tag{33}$$

$$\min_G \mathcal{L}_G = E_{z\sim p_z(z)}\left[L_\theta\left(G\left(z\right)\right)\right], \tag{34}$$

where $[y]^+ = \max(0, y)$, λ is the free tuning-parameter, and θ is the paramter of the discriminator D.

3.2.7 Summary

There is a website called "The GAN Zoo" (https://github.com/hindupuravinash/the-gan-zoo) which lists many GANs' variants. Please refer to this website for more details.

3.3 GANs Training

Despite the theoretical existence of unique solutions, GANs training is hard and often unstable for several reasons [29], [32], [179]. One difficulty is from the fact that optimal weights for GANs correspond to saddle points, and not minimizers, of the loss function.

There are many papers on GANs training. Yadav et al. [180] stabilized GANs with prediction methods. By using independent learning rates, [181] proposed a two time-scale update rule (TTUR) for both discriminator and generator to ensure that the model can converge to a stable local Nash equilibrium. Arjovsky [179] made theoretical steps towards fully understanding the training dynamics of GANs; analyzed why GANs was hard to train; studied and proved rigorously the problems including saturation and instability that occurred when training GANs; examined a practical and theoretically grounded direction to mitigate these problems; and introduced new tools to study them. Liang et al. [182] think that GANs training is a continual learning problem [183].

One method to improve GANs training is to assess the empirical "symptoms" that might occur in training. These symptoms include: the generative model collapsing to produce very similar samples for diverse inputs [29]; the discriminator loss converging quickly to zero [179], providing no gradient updates to the generator; difficulties in making the pair of models converge [32].

We will introduce GANs training from three perspectives: objective function, skills, and structure.

3.3.1 Objective function

As we can see from Subsection 3.1, utilizing the original objective function in equation (1) will have the gradient vanishing problem for training G and utilizing the alternative G loss (12) in non-saturating game will get the mode collapse problem. These problems are caused by the objective function and cannot be solved by changing the structures of GANs. Re-designing the objective function is a natural solution to mitigate these problems. Based on the theoretical flaws of GANs, many objective function based variants have been proposed to change the objective function of GANs based on theoretical analyses such as least squares generative adversarial networks [21], [22].

3.3.1.1 Least squares generative adversarial networks (LSGANs) :
LSGANs [21], [22] are proposed to overcome the vanishing gradient problem in the original GANs. This work shows that the decision boundary for D of original GAN penalizes very small error to update G for those generated samples which are far from the decision boundary. LSGANs adopt the least squares loss rather than the cross-entropy loss in the original GANs. Suppose that the a-b coding is used for the LSGANs' discriminator [21], where a and b are the labels for generated sample and real sample, respectively. The LSGANs' discriminator loss $V_{LSGAN}(D)$ and generator loss $V_{LSGAN}(G)$ are defined as:

$$\begin{aligned}\min_D V_{LSGAN}(D) = E_{x\sim p_{data}(x)}\left[(D(x)-b)^2\right] \\ +E_{z\sim p_z(z)}\left[(D(G(z))-a)^2\right],\end{aligned} \tag{35}$$

$$\min_G V_{LSGAN}(G) = E_{z\sim p_z(z)}\left[(D(G(z))-c)^2\right], \tag{36}$$

where c is the value that G hopes for D to believe for generated samples. The reference [21] shows that there are two advantages of LSGANs in comparison with the original GANs:

- The new decision boundary produced by D penalizes large error to those generated samples which are far from the decision boundary, which makes those "low quality" generated samples move toward the decision boundary. This is good for generating higher quality samples.
- Penalizing the generated samples far from the decision boundary can supply more gradient when updating the G, which overcomes the vanishing gradient problems in the original GANs.

3.3.1.2 Hinge loss based GAN:
Hinge loss based GAN is proposed and used in [23]–[25] and its objective function is $V(D,G)$:

$$\begin{aligned}V_D\left(\hat{G},D\right) = E_{x\sim p_{data}(x)}\left[\min(0,-1+D(x))\right] \\ +E_{z\sim p_z(z)}\left[\min(0,-1-D\left(\hat{G}(z)\right))\right].\end{aligned} \tag{37}$$

$$V_D\left(G,\hat{D}\right) = -E_{z\sim p_z(z)}\left[\hat{D}(G(z))\right]. \tag{38}$$

The softmax cross-entropy loss [184] is also used in GANs.

3.3.1.3 Energy-based generative adversarial network (EBGAN):
EBGAN's discriminator is seen as an energy function, giving high energy to the fake ("generated") samples and lower energy to the real samples. As for the energy function, please refer to [185] for the corresponding tutorial. Given a positive margin m , the loss functions for EBGAN can be defined as follows:

$$\mathcal{L}_D(x,z) = D(x) + [m - D(G(z))]^+, \tag{39}$$

$$\mathcal{L}_G(z) = D(G(z)), \tag{40}$$

where $[y]^+ = \max(0,y)$ is the rectified linear unit (ReLU) function. Note that in the original GANs, the discriminator D give high score to real samples and low score to the generated ("fake") samples. However, the discriminator in EBGAN attributes low energy (score) to the real samples and higher energy to the generated ones. EBGAN has more stable behavior than original GANs during training.

3.3.1.4 Boundary equilibrium generative adversarial networks (BEGAN):
Similar to EBGAN [38], dual-agent GAN (DA-GAN) [186], [187], and margin adaptation for GANs (MAGANs) [188], BEGAN also uses an auto-encoder as the discriminator. Using proportional control theory, BEGAN proposes a novel equilibrium method to balance generator and discriminator in training, which is fast, stable, and robust to parameter changes.

3.3.1.5 Mode regularized generative adversarial networks (MDGAN) :
Che et al. [26] argue that GANs' unstable training and model collapse is due to the very special functional shape of the trained discriminators in high dimensional spaces, which can make training stuck or push probability mass in the wrong direction, towards that of higher concentration than that of the real data distribution. Che et al. [26] introduce several methods of regularizing the objective, which can stabilize the training of GAN models. The key idea of MDGAN is utilizing an encoder $E(x) : x \to z$ to produce the latent variable z for the generator G rather than utilizing noise. This procedure has two advantages:

- Encoder guarantees the correspondence between z ($E(x)$) and x, which makes G capable of covering diverse modes in the data space. Therefore, it prevents the mode collapse problem.
- Because the reconstruction of encoder can add more information to the generator G, it is not easy for the discriminator D to distinguish between real samples and generated ones.

The loss function for the generator and the encoder of MDGAN is

$$\begin{aligned}\mathcal{L}_G = -E_{z\sim p_z(z)}\left[\log(D(G(z)))\right] \\ +E_{x\sim p_{data}(x)}\left[\begin{array}{l}\lambda_1 d(x, G\circ E(x)) \\ +\lambda_2 \log D(G\circ E(x))\end{array}\right],\end{aligned} \tag{41}$$

$$\mathcal{L}_E = E_{x\sim p_{data}(x)}\left[\begin{array}{l}\lambda_1 d(x, G\circ E(x)) \\ +\lambda_2 \log D(G\circ E(x))\end{array}\right], \tag{42}$$

where both λ_1 and λ_2 are free tuning parameters, d is the distance metric such as Euclidean distance, and $G\circ E(x) = G(E(x))$.

3.3.1.6 Unrolled GAN:
Metz et al. [27] introduce a technique to stabilize GANs by defining the generator objective with regard to an unrolled optimization of the discriminator. This allows training to be adjusted between utilizing the current value of the discriminator, which is usually unstable and leads to poor solutions, and utilizing the optimal discriminator solution in the generator's objective, which is perfect but infeasible in real applications. Let $f(\theta_G,\theta_D)$ denote the objective function of the original GANs.

A local optimal solution of the discriminator parameters θ_D^* can be expressed as the fixed point of an iterative optimization procedure,

$$\theta_D^0 = \theta_D, \tag{43}$$

$$\theta_D^{k+1} = \theta_D^k + \eta^k \frac{df\left(\theta_G, \theta_D^k\right)}{d\theta_D^k}, \tag{44}$$

$$\theta_D^*\left(\theta_G\right) = \lim_{k\to\infty} \theta_D^k, \tag{45}$$

where η^k is the learning rate. By unrolling for K steps, a surrogate objective for the update of the generator is created

$$f_K\left(\theta_G, \theta_D\right) = f\left(\theta_G, \theta_D^K\left(\theta_G, \theta_D\right)\right). \tag{46}$$

When $K = 0$, this objective is the same as the standard GAN objective. When $K \to \infty$, this objective is the true generator objective function $f\left(\theta_G, \theta_D^*\left(G\right)\right)$. By adjusting the number of unrolling steps K, we are capable of interpolating between standard GAN training dynamics with their related pathologies, and more expensive gradient descent on the true generator loss. The generator and discriminator parameter updates of unrolled GAN using this surrogate loss are

$$\theta_G = \theta_G - \eta \frac{df_K\left(\theta_G, \theta_D\right)}{d\theta_G}, \tag{47}$$

$$\theta_D = \theta_D + \eta \frac{df\left(\theta_G, \theta_D\right)}{d\theta_D}. \tag{48}$$

Metz et al. [27] show how this method solves mode collapse, stabilizes training of GANs, and increases diversity and coverage of the generated distribution by the generator.

3.3.1.7 Spectrally normalized GANs (SN-GANs):
SN-GANs [23] propose a novel weight normalization method named spectral normalization to make the training of the discriminator stable. This new normalization technique is computationally efficient and easy to be integrated into existing methods. The spectral normalization [23] uses a simple method to make the weight matrix W satisfy the Lipschitz constraint $\sigma\left(W\right) = 1$:

$$\bar{W}_{SN}\left(W\right) := W/\sigma\left(W\right), \tag{49}$$

where W is the weight matrix of each layer in D and $\sigma\left(W\right)$ is the spectral norm of W. It is shown that [23] SN-GANs can generate images of equal or better quality in comparison with the previous training stabilization methods. In theory, spectral normalization is capable of being applied to all GANs variants. Both BigGANs [36] and SAGAN [35] use the spectral normalization and have good performances on the Imagenet.

3.3.1.8 Relativistic GANs (RGANs):
In the original GANs, the discriminator can be defined, according to the non-transformed layer $C(x)$, as $D(x) = sigmoid(C(x))$. A simple way to make discriminator relativistic (i.e., making the output of D depend on both real and generated samples) [28] is to sample from real/generated data pairs $\tilde{x} = (x_r, x_g)$ and define it as

$$D\left(\tilde{x}\right) = sigmoid\left(C\left(x_r\right) - C\left(x_g\right)\right). \tag{50}$$

This modification can be understood in the following way [28]: D estimates the probability that the given real sample is more realistic than a randomly sampled generated sample. Similarly, $D_{rev}\left(\tilde{x}\right) = sigmoid\left(C\left(x_g\right) - C\left(x_r\right)\right)$ can be defined as the probability that the given generated sample is more realistic than a randomly sampled real sample. The discriminator and generator loss functions of the Relativistic Standard GAN (RSGAN) is:

$$L_D^{RSGAN} = -E_{(x_r, x_g)}\left[\log\left(sigmoid\left(C\left(x_r\right) - C\left(x_g\right)\right)\right)\right], \tag{51}$$

$$L_G^{RSGAN} = -E_{(x_r, x_g)}\left[\log\left(sigmoid\left(C\left(x_g\right) - C\left(x_r\right)\right)\right)\right]. \tag{52}$$

Most GANs can be parametrized:

$$L_D^{GAN} = E_{x_r}\left[f_1\left(C\left(x_r\right)\right)\right] + E_{x_g}\left[f_2\left(C\left(x_g\right)\right)\right], \tag{53}$$

$$L_G^{GAN} = E_{x_r}\left[g_1\left(C\left(x_r\right)\right)\right] + E_{x_g}\left[g_2\left(C\left(x_g\right)\right)\right], \tag{54}$$

where f_1, f_2, g_1, g_2 are scalar-to-scalar functions. If we adopt a relativistic discriminator, the loss functions of these GANs become:

$$\begin{array}{c} L_D^{RGAN} = E_{(x_r, x_g)}\left[f_1\left(C\left(x_r\right) - C\left(x_g\right)\right)\right] \\ +E_{(x_r, x_g)}\left[f_2\left(C\left(x_g\right) - C\left(x_r\right)\right)\right] \end{array}, \tag{55}$$

$$\begin{array}{c} L_G^{RGAN} = E_{(x_r, x_g)}\left[g_1\left(C\left(x_r\right) - C\left(x_g\right)\right)\right] \\ +E_{(x_r, x_g)}\left[g_2\left(C\left(x_g\right) - C\left(x_r\right)\right)\right]. \end{array} \tag{56}$$

3.3.2 Skills

NIPS 2016 held a workshop on adversarial training, with an invited talk by Soumith Chintala named "How to train a GAN." This talk has an assorted tips and tricks. For example, this talk suggests that if you have labels, training the discriminator to also classify the examples: ACGAN [30]. Refer to the GitHub repository associated with Soumith's talk: https://github.com/soumith/ganhacks for more advice.

Salimans et al. [29] proposed very useful and improved techniques for training GANs (ImprovedGANs), such as feature matching, minibatch discrimination, historical averaging, one-sided label smoothing, and virtual batch normalization.

3.3.3 Structure

The original GANs utilized multi-layer perceptron (MLP). Specific type of structure may be good for specific applications e.g., recurrent neural network (RNN) for time series data and convolutional neural network (CNN) for images.

3.3.3.1 The original GANs:
The original GANs used MLP as the generator G and discriminator D. MLP can be only used for small datasets such as CIFAR-10 [189], MNIST [190], and the Toronto Face Database (TFD) [191]. However, MLP does not have good generalization on more complex images [10].

3.3.3.2 Laplacian generative adversarial networks (LAPGAN) and SinGAN:
LAPGAN [31] is proposed for producing higher resolution images in comparison with the original GANs. LAPGAN uses a cascade of CNN within a Laplacian pyramid framework [192] to generate high quality images.

SinGAN [193] learns a generative model from a single natural image. SinGAN has a pyramid of fully convolutional GANs, each of which learns the patch distribution at a different scale of the image. Similar to SinGAN, InGAN [194] also learns a generative model from a single natural image.

3.3.3.3 Deep convolutional generative adversarial networks (DCGANs):
In original GANs, G and D are defined by MLP. Because CNN are better at images than MLP, G and D are defined by deep convolutional neural networks (DCNNs) in DCGANs [32], which have better performance. Most current GANs are at least loosely based on the DCGANs architecture [32]. Three key features of the DCGANs architecture are listed as follows:

- First, the overall architecture is mostly based on the all-convolutional net [195]. This architecture has neither pooling nor "unpooling" layers. When G needs to increase the spatial dimensionality of the representation, it uses transposed convolution (deconvolution) with a stride greater than 1.
- Second, utilize batch normalization for most layers of both G and D. The last layer of G and first layer of D are not batch normalized, in order that the neural network can learn the correct mean and scale of the data distribution.
- Third, utilize the Adam optimizer instead of SGD with momentum.

3.3.3.4 Progressive GAN:
In Progressive GAN (PGGAN) [33], a new training methodology for GAN is proposed. The structure of Progressive GAN is based on progressive neural networks that is first proposed in [196]. The key idea of Progressive GAN is to grow both the generator and discriminator progressively: starting from a low resolution, adding new layers that model increasingly fine details as training progresses.

3.3.3.5 Self-Attention Generative Adversarial Network (SAGAN):
SAGAN [35] is proposed to allow attention-driven, long-range dependency modeling for image generation tasks. Spectral normalization technique has only been applied to the discriminator [23]. SAGAN uses spectral normalization for both generator and discriminator and it is found that this improves training dynamics. Furthermore, it is confirmed that the two time-scale update rule (TTUR) [181] is effective in SAGAN.

Note that AttnGAN [197] utilizes attention over word embeddings within an input sequence rather than self-attention over internal model states.

3.3.3.6 BigGANs and StyleGAN:
Both BigGANs [36] and StyleGAN [37], [198] made great advances in the quality of GANs.

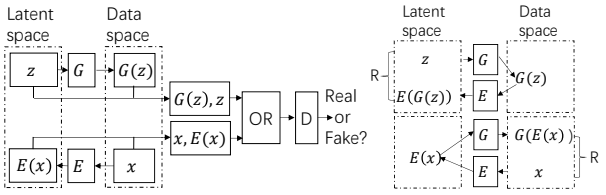
BigGANs [36] is a large scale TPU implementation of GANs, which is pretty similar to SAGAN but scaled up greatly. BigGANs successfully generates images with quite high resolution up to 512 by 512 pixels. If you do not have enough data, it can be a challenging task to replicate the BigGANs results from scratch. Lucic et al. [199] propose to train BigGANs quality model with fewer labels. BigBiGAN [200], based on BigGANs, extends it to representation learning by adding an encoder and modifying the discriminator. BigBiGAN achieve the state of the art in both unsupervised representation learning on ImageNet and unconditional image generation.

In the original GANs [3], G and D are defined by MLP. Karras et al. [37] proposed a StyleGAN architecture for GANs, which wins the CVPR 2019 best paper honorable mention. StyleGAN's generator is a really high-quality generator for other generation tasks like generating faces. It is particular exciting because it allows to separate different factors such as hair, age and sex that are involved in controlling the appearance of the final example and we can then control them separately from each other. StyleGAN [37] has also been used in such as generating high-resolution fashion model images wearing custom outfits [201].

3.3.3.7 Hybrids of autoencoders and GANs:
An autoencoder is a type of neural networks used for learning efficient data codings in an unsupervised way. The autoencoder has an encoder and a decoder. The encoder aims to learn a representation (encoding) for a set of data, $z = E(x)$, typically for dimensionality reduction. The decoder aims to reconstruct the data $\hat{x} = g(z)$. That is to say, the decoder tries to generate from the reduced encoding a representation as close as possible to its original input x.

GANs with an autoencoder: Adversarial autoencoder (AAE) [202] is a probabilistic autoencoder based on GANs. Adversarial variational Bayes (AVB) [203], [204] is proposed to unify variational autoencoders (VAEs) and GANs. Sun et al. [205] proposed a UNsupervised Image-to-image Translation (UNIT) framework that are based on GANs and VAEs. Hu et al. [206] aimed to establish formal connections between GANs and VAEs through a new formulation of them. By combining a VAE with a GAN, Larsen et al. [207] utilize learned feature representations in the GAN discriminator as basis for the VAE reconstruction. Therefore, element-wise errors is replaced with feature-wise errors to better capture the data distribution while offering invariance towards such as translation. Rosca et al. [208] proposed variational approaches for auto-encoding GANs. By adopting an encoder-decoder architecture for the generator, disentangled representation GAN (DR-GAN) [68] addresses pose-invariant face recognition, which is a hard problem due to the drastic changes in an image for each diverse pose.

GANs with an encoder: References [40], [42] only add an encoder to GANs. The original GANs [3] can not learn the inverse mapping - projecting data back into the latent space. To solve this problem, Donahue et al. [40] proposed Bidirectional GANs (BiGANs), which can learn this inverse mapping through the encoder, and show that the resulting learned feature representation is useful. Similarly, Dumoulin et al. [41] proposed the adversarially learned inference (ALI) model, which also utilizes the encoder to learn the latent feature distribution. The structure of BiGAN and ALI is shown in Fig. 3(a). Besides the discriminator and generator, BiGAN also has an encoder, which is used for mapping the



(a) BiGAN/ALI (b) AGE

Fig. 3: The structures of (a) BiGAN and ALI and (b) AGE.

data back to the latent space. The input of discriminator is a pair of data composed of data and its corresponding latent code. For real data x, the pair are $x, E(x)$ where $E(x)$ is obtained from the encoder E. For the generated data $G(z)$, this pair is $G(z), z$ where z is the noise vector generating the data $G(z)$ through the generator G. Similar to (1), the objective function of BiGAN is

$$\begin{aligned}\min_{G,E}\max_{D}\ V(D,E,G) &= E_{x\sim p_{data}(x)}\left[\log D\left(x, E(x)\right)\right] \\ &+E_{z\sim p_z(z)}\left[\log\left(1-D\left(G\left(z\right),z\right)\right)\right].\end{aligned} \tag{57}$$

The generator of [40], [42] can be seen the decoder since the generator map the vectors from latent space to data space, which performs the same function as the decoder.

Similar to utilizing an encoding process to model the distribution of latent samples, Gurumurthy et al. [209] model the latent space as a mixture of Gaussians and learn the mixture components that maximize the likelihood of generated samples under the data generating distribution.

In an encoding-decoding model, the output (also known as a reconstruction), ought to be similar to the input in the ideal case. Generally, the fidelity of reconstructed samples synthesized utilizing a BiGAN/ALI is poor. With an additional adversarial cost on the distribution of data samples and their reconstructions [158], the fidelity of samples may be improved. Other related methods include such as variational discriminator bottleneck (VDB) [210] and MDGAN (detailed in Paragraph 3.3.1.5).

Combination of a generator and an encoder: Different from previous hybrids of autoencoders and GANs, Adversarial Generator-Encoder (AGE) Network [42] is set up directly between the generator and the encoder, and no external mappings are trained in the learning process. The structure of AGE is shown in Fig. 3(b) which R is the reconstruction loss function. In AGE, there are two reconstruction losses: the latent variable z and $E(G(z))$, the data x and $G(E(x))$. AGE is similar to CycleGAN. However, there are two differences between them:

- CycleGAN [53] is used for two modalities of the image such as grayscale and color. AGE acts between latent space and true data space.
- There is a discriminator for each modality of CycleGAN and there is no discriminator in AGE.

3.3.3.8 Multi-discriminator learning:
GANs have a discriminator together with a generator. Different from GANs, dual discriminator GAN (D2GAN) [43] has **a generator and two binary discriminators**. D2GAN is analogous to a minimax game, wherein one discriminator gives high scores for samples from generated distribution whilst the other discriminator, conversely, favoring data from the true distribution, and the generator generates data to fool both discriminators. The reference [43] develops theoretical analysis to show that, given the optimal discriminators, optimizing the generator of D2GAN is minimizing both KL and reverse KL divergences between true distribution and the generated distribution, thus effectively overcoming the mode collapsing problem. Generative multi-adversarial network (GMAN) [44] further extends GANs to a generator and multiple discriminators. Albuquerque et al. [211] performed multi-objective training of GANs with multiple discriminators.

3.3.3.9 Multi-generator learning:
Multi-generator GAN (MGAN) [45] is proposed to train the GANs with a mixture of generators to avoid the mode collapsing problem. More specially, MGAN has **one binary discriminator, K generators, and a multi-class classifier**. The distinguishing feature of MGAN is that the generated samples are produced from multiple generators, and then only one of them will be randomly selected as the final output similar to the mechanism of a probabilistic mixture model. The classifier shows which generator a generated sample is from.

The most closely related to MGAN is multi-agent diverse GANs (MAD-GAN) [46]. The difference between MGAN and MAD-GAN can be found in [45]. SentiGAN [212] uses a mixture of generators and a multi-class discriminator to generate sentimental texts.

3.3.3.10 Multi-GAN learning:
Coupled GAN (CoGAN) [47] is proposed for learning a joint distribution of two-domain images. CoGAN is composed of a pair of GANs - GAN1 and GAN2, each of which synthesizes images in one domain. Two GANs respectively considering structure and style are proposed in [213] based on cGANs. Causal fairness-aware GANs (CFGAN) [214] used two generators and two discriminators for generating fair data. The structures of GANs, D2GAN, MGAN, and CoGAN are shown in Fig. 4.

3.3.3.11 Summary:
There are many GANs' variants and milestone ones are shown in Fig. 5. Due to space limitation, only limited number of variants are shown in Fig. 5.

GANs' objective function based variants can be generalized to structure variants. Compared with other objective function based variants, both SN-GANs and RGANs show the stronger generalization ability. These two objective function based variants can be generalized to the other objective function based variants. Spectral normalization is capable of being generalized to any type of GANs' variants while RGANs is able to be generalized to any IPM-based GANs.

3.4 Evaluation metrics for GANs

In this subsection, we show evaluation metrics [215], [216] that are used for GANs.

3.4.1 Inception Score (IS)

Inception score (IS) is proposed in [29], which uses the Inception model [217] for every generated image to get the conditional label distribution $p\left(y|x\right)$. Images that have

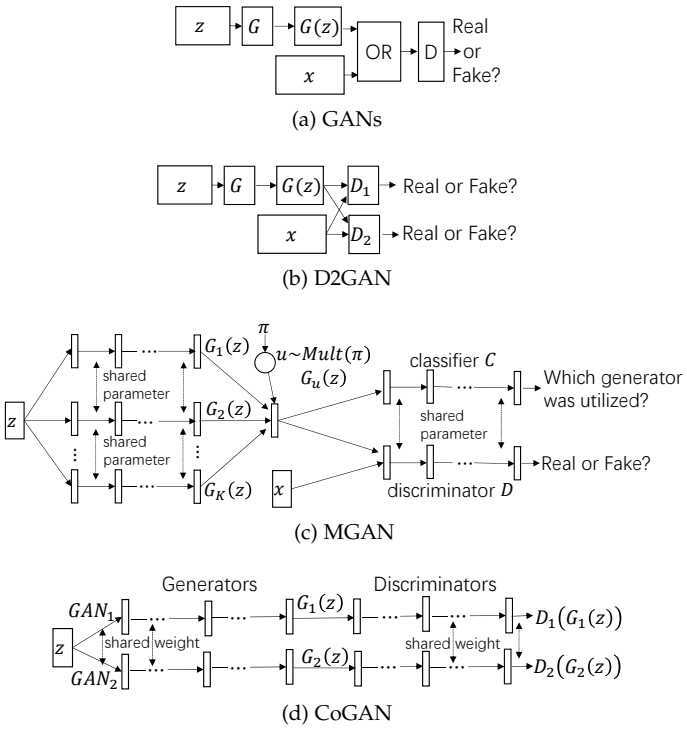


Fig. 4: The structures of GANs [3], D2GAN [43], MGAN [45], and CoGAN [47].

meaningful objects ought to have a conditional label distribution $p(y|x)$ with low entropy. Furthermore, the model is expected to produce diverse images. Therefore, the marginal $\int p(y|x = G(z))dz$ ought to have high entropy. In combination of these two requirements, the IS is:

$$\exp(E_x KL(p(y|x)||p(y))), \tag{58}$$

where exponentiating results is for easy comparison of the values.

A higher IS indicates that the generative model can produce high quality samples and the generated samples are also diverse. However, the IS also has disadvantages. If the generative model falls into mode collapse, the IS might be still good while the real case is pretty bad. To address this issue, an independent Wasserstein critic [218] is proposed to be trained independently for the validation dataset to measure mode collapse and overfitting.

3.4.2 Mode score (MS)

The mode score (MS) [26], [219] is an improved version of the IS. Different from IS, MS can measure the dissimilarity between the real distribution and generated distribution.

3.4.3 Fréchet Inception Distance (FID)

FID was also proposed [181] to evaluate GANs. For a suitable feature function ϕ (the default one is the Inception network's convolutional feature), FID models $\phi(p_{data})$ and $\phi(p_g)$ as Gaussian random variables with empirical means $\mu_r,\ \mu_g$ and empirical covariance $C_r,\ C_g$ and computes

$$\begin{aligned} FID(p_{data}, p_g) &= \|\mu_r - \mu_g\| \\ &+ tr\left(C_r + C_g - 2(C_r C_g)^{1/2}\right), \end{aligned} \tag{59}$$

which is the Fréchet distance (also called the Wasserstein-2 distance) between the two Gaussian distributions. However, the IS and FID cannot well handle the overfitting problem. To mitigate this problem, the kernel inception distance (KID) was proposed in [220].

3.4.4 Multi-scale structural similarity (MS-SSIM)

Structural similarity (SSIM) [221] is proposed to measure the similarity between two images. Different from single scale SSIM measure, the MS-SSIM [222] is proposed for multi-scale image quality assessment. It quantitatively evaluates image similarity by attempting to predict human perceptual similarity judgment. The range of MS-SSIM values is between 0.0 and 1.0 and lower MS-SSIM values means perceptually more dissimilar images. References [30], [223] used MS-SSIM to measure the diversity of fake data. Reference [224] suggested that MS-SSIM should only be taken into account together with the FID and IS metrics for testing sample diversity.

3.4.5 Summary

How to select a good evaluation metric for GANs is still a hard problem [225]. Xu et al. [219] proposed an empirical study on evaluation metrics of GANs. Karol Kurach [224] conducted a large-scale study on regularization and normalization in GANs. There are some other comparative study of GANs such as [226]. Reference [227] presented several measures as meta-measures to guide researchers to select quantitative evaluation metrics. An appropriate evaluation metric ought to differentiate true samples from fake ones, verify mode drop, mode collapse, and detect overfitting. It is hoped that there will be better methods to evaluate the quality of the GANs model in the future.

3.5 Task driven GANs

The focus of this paper is on GANs. There are closely associated fields for specific tasks with an enormous volume of literature.

3.5.1 Semi-Supervised Learning

A research field where GANs are very successful is the application of generative models to semi-supervised learning [228], [229], as proposed but not shown in the first GANs paper [3].

GANs have been successfully used for semi-supervised learning at least since CatGANs [48]. Feature matching GANs [29] got good performance with a small number of labels on datasets such as MNIST, SVHN, and CIFAR-10.

Odena [230] extends GANs to the semi-supervised learning by forcing the discriminator network to output class labels. Generally speaking, when we train GANs, we actually do not use the discriminator in the end. The discriminator is only used to guide the learning process, but the discriminator is not used to generate the data after we have trained the generator. We only use the generator to generate the data and abandon discriminator at last. For traditional GANs, the discriminator is a two-class classifier, which outputs category one for real data and category two for generated data. In semi-supervised learning, the discriminator is upgraded to be a multi-class classifier. At the end of the training, the

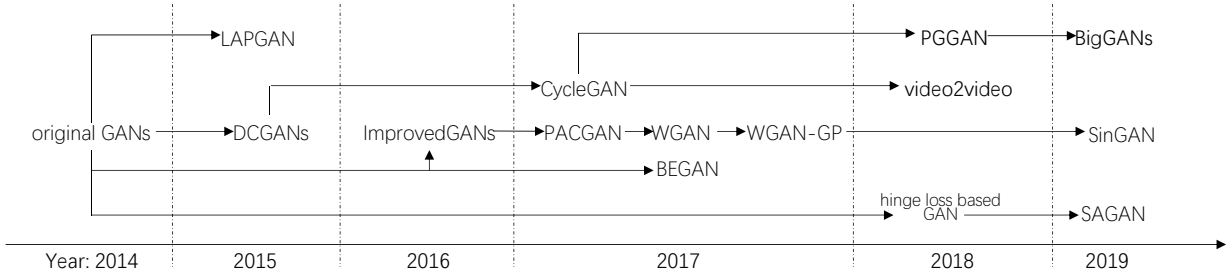


Fig. 5: A road map of GANs. Milestone variants are shown in this figure.

classifier is the thing that we are interested in. For semi-supervised learning, if we want to learn an N class classifier, we make GANs with a discriminator which can predict N+1 classes the input is from, where an extra class corresponds to the outputs of G. Therefore, suppose that we want to learn to classify two classes, apples and oranges. We can make a classifier which has three different labels: one - the class of real apples, two - the class of real oranges, and three - the class of generated data. The system learns on three kinds of data: real labeled data, unlabeled real data, and fake data.

Real labeled data: We can tell the discriminator to maximize the probability of the correct class. For example, if we have an apple photo and it is labeled as an apple, we can maximize the probability of the apple class in this discriminator.

Unlabeled real data: Suppose we have a photo, we do not know whether it is an apple or an orange but we know that it is a real photo. In this situation, we train the discriminator to maximize the sum of the probabilities over all the real classes.

Fake data: When we obtain a generated example from the generator, we train the discriminator to classify it as a fake example.

Miyato et al. [49] proposed virtual adversarial training (VAT): a regularization method for both supervised and semi-supervised learning. Dai et al. [231] show that given the discriminator objective, good semi-supervised learning indeed requires a bad generator from the theory perspective, and propose the definition of a preferred generator. A triangle GAN (Δ-GAN) [50] is proposed for semi-supervised cross-domain joint distribution matching and Δ-GAN is closely related to Triple-GAN [51]. Madani et al. [232] used semi-supervised learning with GANs for chest X-ray classification.

Future improvements to GANs can be expected to simultaneously produce further improvements to semi-supervised learning and unsupervised learning such as self-supervised learning [233].

3.5.2 Transfer learning

Ganin et al. [234] introduce a domain-adversarial training approach of neural networks for domain adaptation, where training data and test data are from similar but different distributions. The Professor Forcing algorithm [235] uses adversarial domain adaptation for training recurrent network. Shrivastava et al. [236] used GANs for simulated training data. A novel extension of pixel-level domain adaptation named GraspGAN [237] was proposed for robotic grasping [238], [239]. By using synthetic data and domain adaptation [237], the number of real-world examples needed to achieve a given level of performance is reduced by up to 50 times, utilizing only randomly generated simulated objects.

Recent studies have shown remarkable success in image-to-image translation [240]–[245] for two domains. However, existing methods such as CycleGAN [53], DiscoGAN [54], and DualGAN [55], cannot be used directly for more than two domains, since different approaches should be built independently for every pair of domains. StarGAN [56] well solve this problem which can conduct image-to-image translations for multiple domains using only a single model. Other related works can be found in [246], [247]. CoGAN [47] can be also used for multiple domains.

Learning fair representations is a closely related problem to domain transfer. Note that different formulations of adversarial objectives [248]–[251] achieve different notations of fairness.

Domain adaptation [61], [252] can be seen as a subset of transfer learning [253]. Recent **visual domain adaptation (VDA)** methods include: **visual appearance adaptation, representation adaptation**, and **output adaptation**, which can be thought of as using domain adaptation based on the original input, features, and outputs of the domains, respectively.

Visual appearance adaptation: CycleGAN [53] is a representative method in this category. CyCADA [57] is proposed for visual appearance adaptation based on CycleGAN.

Representation adaptation: The key of adversarial discriminative domain adaptation (ADDA) [58], [59] is to learn feature representations that a discriminator cannot differentiate which domain they belong to. Sankaranarayanan et al. [254] focused on adapting the representations learned by segmentation networks across real and synthetic domains based on GANs. Fully convolutional adaptation networks (FCAN) [60] is proposed for semantic segmentation which combines visual appearance adaptation and representation adaptation.

Output adaptation: Tsai [255] made the outputs of the source and target images have a similar structure so that the discriminator cannot differentiate them.

Other transfer learning based GANs can be found in [52], [256]–[262].

3.5.3 Reinforcement learning

Generative models can be integrated into reinforcement learning (RL) [107] in different ways [103], [263]. Reference

[264] has already discussed connections between GANs and actor-critic methods. The connections among GANs, inverse reinforcement learning (IRL), and energy-based models is studied in [265]. These connections to RL are possible to be useful for the development of both GANs and RL. Furthermore, GANs were combined with reinforcement learning for synthesizing programs for images [266]. The competitive multi-agent learning framework proposed in [267] is also related to GANs and works on learning robust grasping policies by an adversary.

Imitation Learning: The connection between imitation learning and EBGAN is discussed in [268]. Ho and Ermon [269] show that an instantiation of their framework draws an analogy between GANs and imitation learning, from which they derive a model-free imitation learning method that has significant performance gains over existing model-free algorithms in imitating complex behaviors in large and high-dimensional environments. Song et al. [62] proposed multi-agent generative adversarial imitation learning (GAIL) and Guo et al. [270] proposed generative adversarial self-imitation learning. A multi-agent GAIL framework is used in a deconfounded multi-agent environment reconstruction (DEMER) approach [271] to learn the environment. DEMER is tested in the real application of Didi Chuxing and has achieved good performances.

3.5.4 Multi-modal learning

Generative models, especially GANs, make machine learning be able to work with multi-modal outputs. In many tasks, an input may correspond to multiple diverse correct outputs, each of which is an acceptable answer. Traditional ways of training machine learning methods, such as minimizing the mean squared error (MSE) between the model's predicted output and a desired output, are not capable of training models that can produce many different correct outputs. One instance of such a case is predicting the next frame in a video sequence, as shown in Fig. 6. Multi-modal image-to-image translation related works can be found in [272]–[275].

3.5.5 Other task driven GANs

GANs have been used for feature learning such as feature selection [277], hashing [278]–[285], and metric learning [286].

MisGAN [287] was proposed to learn from incomplete data with GANs. Evolutionary GANs are proposed in [288]. Ponce et al. [289] combined GANs and genetic algorithms to evolve images for visual neurons. GANs have also been used in other machine learning tasks [290] such as active learning [291], [292], online learning [293], ensemble learning [294], zero-shot learning [295], [296], and multi-task learning [297].

4 THEORY

In this section, we first introduce maximum likelihood estimation. Then, we introduce mode collapse. Finally, other theoretical issues such as inverse mapping and memorization are discussed.

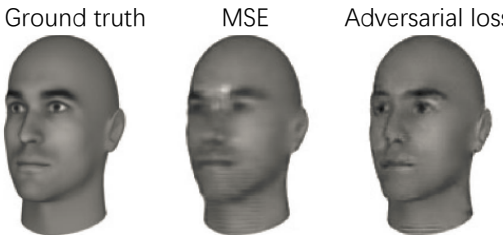


Fig. 6: Lotter et al. [276] show an excellent description of the significance of being capable of modeling multi-modal data. In this instance, a model is trained to predict the next frame in a video. The video describes a computer rendering of a moving 3D model of a man's head. The image on the left is the ground truth, an instance of an actual frame of a video, which the model would like to predict. The image in the middle is what happens when the model is trained using mean squared error (MSE) between the model's predicted next frame and the actual next frame. The model is forced to select only one answer for what the next frame will be. Since there are multiple possible answers, corresponding to slightly diverse positions of the head, the answer that the model selects is an average over multiple slightly diverse images. This causes the faces to have a blurring effect. Utilizing an additional GANs loss, the image on the right is capable of knowing that there are multiple possible outputs, each of which is recognizable and clear as a realistic, satisfying image. Images are from [276].

4.1 Maximum likelihood estimation (MLE)

Not all generative models use maximum likelihood estimation (MLE). Some generative models do not utilize MLE, but can be made to do so (GANs belong to this category). It can be simply proved that minimizing the Kullback-Leibler Divergence (KLD) between $p_{data}(x)$ and $p_g(x)$ is equal to maximizing the log likelihood as the number of samples m increases:

$$\begin{aligned}\theta^* &= \arg\min_{\theta} KLD\left(p_{data} \| p_g\right) \\ &= \arg\min_{\theta} -\int p_{data}\left(x\right) \log \frac{p_g(x)}{p_{data}(x)} dx \\ &= \arg\min_{\theta} \int p_{data}\left(x\right) \log p_{data}\left(x\right) dx \\ &\quad - \int p_{data}\left(x\right) \log p_g\left(x\right) dx \\ &= \arg\max_{\theta} \int p_{data}\left(x\right) \log p_g\left(x\right) dx \\ &= \arg\max_{\theta} \lim_{m\to\infty} \frac{1}{m} \sum_{i=1}^{m} \log p_g\left(x_i\right).\end{aligned} \tag{60}$$

The model probability distribution $p_\theta\left(x\right)$ is replaced with $p_g\left(x\right)$ for notation consistency. Refer to Chapter 5 of [298] for more information on MLE and other statistical estimators.

4.2 Mode collapse

GANs are hard to train, and it has been observed [26], [29] that they often suffer from mode collapse [299], [300], in which the generator learns to generate samples from only a few modes of the data distribution but misses many other modes, even if samples from the missing modes exist throughout the training data. In the worst case, the generator produces simply a single sample (complete collapse)

[179], [301]. In this subsection, we will first introduce two viewpoints of GANs mode collapse. Then, we will introduce methods that propose new objective functions or new structures to solve mode collapse.

4.2.1 Two viewpoints: divergence and algorithmic

We can resolve and understand GANs mode collapse and instability from two viewpoints: divergence and algorithmic.

Divergence viewpoint: Roth et al. [302] stabilizd training of GANs and their variants such as f-divergence based GANs (f-GAN) through regularization.

Algorithmic viewpoint: The numerics of common algorithms for training GANs are analyzed and a new algorithm that has better convergence properties is proposed in [303]. Mescheder et al. [304] showed that which training methods for GANs do actually converge.

4.2.2 Methods overcoming mode collapse

Objective function based methods: Deep regret analytic GAN (DRAGAN) [170] suggests that the mode collapse exists due to the occurence of a fake local Nash equilibrium in the nonconvex problem. DRAGAN solves this problem by constraining gradients of the discriminator around the real data manifold. It adds a gradient penalizing term which biases the discriminator to have a gradient norm of 1 around the real data manifold. Other methods such as EBGAN and Unrolled GAN (detailed in Section 3.3) also belong to this category.

Structure based methods: Representative methods in this category include such as MAD-GAN [46] and MRGAN [26] (detailed in Section 3.3)

There are also other methods to reduce mode collapse in GANs. For example, PACGAN [305] eases the pain of mode collapse by changing input to the discriminator.

4.3 Other theoretical issues

4.3.1 Do GANs actually learn the distribution?

References [41], [301], [306] have both empirically and theoretically brought the concern to light that distributions learned by GANs suffer from mode collapse. In contrast, Bai et al. [307] show that GANs can in principle learn distributions in Wasserstein distance (or KL-divergence in many situations) with polynomial sample complexity, if the discriminator class has strong discriminating power against the particular generator class (instead of against all possible generators). Liang et al. [308] studied how well GANs learn densities, including nonparametric and parametric target distributions. Singh et al. [309] further studied nonparametric density estimation with adversarial losses.

4.3.2 Divergence/Distance

Arora et al. [301] show that training of GAN may not have good generalization properties; e.g., training may look successful but the generated distribution may be far from real data distribution in standard metrics. The popular distances such as Wasserstein and Jensen-Shannon (JS) may not generalize. However, generalization does occur by introducing a novel notion of distance between distributions, the neural net distance. Are there other useful divergences?

4.3.3 Inverse mapping

GANs cannot learn the inverse mapping - projecting data back into the latent space. BiGANs [40] (detailed in Section 3.3.3.7) is proposed as a way of learning this inverse mapping. Dumoulin et al. [41] introduce the adversarially learned inference (ALI) model (detailed in Section 3.3.3.7), which jointly learns an inference network and a generation network utilizing an adversarial process. Arora et al. [310] show the theoretical limitations of Encoder-Decoder GAN architectures such as BiGANs [40] and ALI [41]. Creswell et al. [311] invert the generator of GANs.

4.3.4 Mathematical perspective such as optimization

Mohamed et al. [312] frame GANs within the algorithms for learning in implicit generative models that only specify a stochastic procedure with which to generate data. Gidel et al. [313] looked at optimization approaches designed for GANs and casted GANs optimization problems in the general variational inequality framework. The convergence and robustness of training GANs with regularized optimal transport is disscussed in [314].

4.3.5 Memorization

As for "memorization of GANs", Nagarajan et al. [315] argue that making the generator "learn to memorize" the training data is a more difficult task than making it "learn to output realistic but unseen data".

5 Applications

As discussed earlier, GANs are a powerful generative model which can generate realistic-looking samples with a random vector z. We neither need to know an explicit true data distribution nor have any mathematical assumptions. These advantages allow GANs to be widely applied to many areas such as image processing and computer vision, sequential data.

5.1 Image processing and computer vision

The most successful applications of GANs are in image processing and computer vision, such as image super-resolution, image synthesis and manipulation, and video processing.

5.1.1 Super-resolution (SR)

SRGAN [63], GANs for SR, is the first framework able to infer photo-realistic natural images for upscaling factors. To further improve the visual quality of SRGAN, Wang et al. [64] thoroughly study three key components of SRGAN and improve each of them to derive an Enhanced SRGAN (ESRGAN). For example, ESRGAN uses the idea from relativistic GANs [28] to have the discriminator predict relative realness rather than the absolute value. Benefiting from these improvements, ESRGAN won the first place in the PIRM2018-SR Challenge (region 3) [316] and got the best perceptual index. Based on CycleGAN [53], the Cycle-in-Cycle GANs [65] is proposed for unsupervised image SR. SRDGAN [66] is proposed to learn the noise prior for SR with DualGAN [55]. Deep tensor generative adversarial nets (TGAN) [67] is proposed to generate large high-quality

images by exploring tensor structures. There are methods specific for face SR [317]–[319]. Other related methods can be found in [320]–[323].

5.1.2 Image synthesis and manipulation

5.1.2.1 Face:

Pose related: Disentangled representation learning GAN (DR-GAN) [324] is proposed for pose-invariant face recognition. Huang et al. [69] proposed a Two-Pathway GAN (TP-GAN) for photorealistic frontal view synthesis by simultaneously perceiving local details and global structures. Ma et al. [70] proposed the novel Pose Guided Person Generation Network (PG^2) that synthesizes person images in arbitrary poses, based on a novel pose and an image of that person. Cao et al. [325] proposed a high fidelity pose invariant model for high-resolution face frontalization based on GANs. Siarohin et al. [326] proposed deformable gans for pose-based human image generation. Pose-robust spatial-aware GAN (PSGAN) for customizable makeup transfer is proposed in [71].

Portrait related: APDrawingGAN [72] is proposed to generate artistic portrait drawings from face photos with hierarchical GANs. APDrawingGAN has a software based on wechat and the results are shown in Fig. 8. GANs have also been used in other face related applications such as facial attribute changes [327] and portrait editing [328]–[331].

Face generation: The quality of generated faces by GANs is improved year by year, which can be found in Sebastian Nowozin's GAN lecture materials[1]. As we can see from Figure 7, the generated faces based on original GANs [3] are of low visual quality and can only serves as a proof of concept. Radford et al. [32] used better neural network architectures: deep convolutional neural networks for generating faces. Roth et al. [302] addressed the instability problems of GAN training, allowing for larger architectures such as the ResNet to be utilized. Karras et al. [33] utilized multiscale training, allowing megapixel face image generation at high fidelity.

Face generation [332]–[340] is somewhat easy because there is only one class of object. Every object is a face and most face data sets tend to be composed of people looking straight into the camera. Most people have been registered in terms of putting nose and eyes and other landmarks in consistent locations.

5.1.2.2 General object:

It is a little harder to have GANs work on assorted data sets like ImageNet [151] which has a thousand different object classes. However, we have seen rapid progress over the recent few years. The quality of these images has been improved year by year [304].

While most papers use GANs to synthesize images in two dimensions [341], [342], Wu et al. [343] synthesized three-dimensional (3-D) samples using GANs and volumetric convolutions. Wu et al. [343] synthesized novel objects including cars, chairs, sofa, and tables. Im et al. [344] generated images with recurrent adversarial networks. Yang et al. [345] proposed layered recursive GANs (LR-GAN) for image generation.

1. https://github.com/nowozin/mlss2018-madrid-gan

5.1.2.3 Interaction between a human being and an image generation process:

There are many applications that involve interaction between a human being and an image generation process. Realistic image manipulation is difficult because it requires modifying the image in a user-controlled way, while making it appear realistic. If the user does not have efficient artistic skill, it is easy to deviate from the manifold of natural images while editing. Interactive GAN (IGAN) [73] defines a class of image editing operations, and constrain their output to lie on that learned manifold at all times. Introspective adversarial networks [74] also offer this capability to perform interactive photo editing and have demonstrated their results mostly in face editing. GauGAN [75] can turn doodles into stunning, photorealistic landscapes.

5.1.3 Texture synthesis

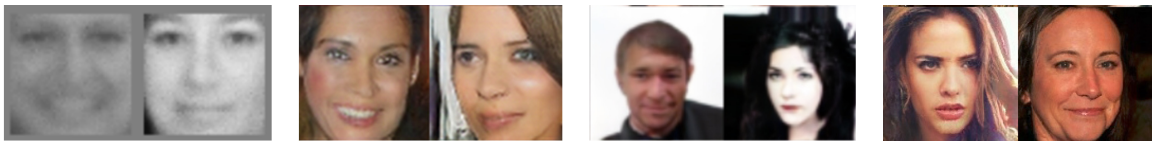
Texture synthesis is a classical problem in image field. Markovian GANs (MGAN) [76] is a texture synthesis method based on GANs. By capturing the texture data of Markovian patches, MGAN can generate stylized videos and images very quickly, in order to realize real-time texture synthesis. Spatial GAN (SGAN) [77] was the first to apply GANs with fully unsupervised learning in texture synthesis. Periodic spatial GAN (PSGAN) [78] is a variant of SGAN, which can learn periodic textures from a single image or complicated big dataset.

5.1.4 Object detection

How can we learn an object detector that is invariant to deformations and occlusions? One way is using a data-driven strategy - collect large-scale datasets which have object examples under different conditions. We hope that the final classifier can use these instances to learn invariances. Is it possible to see all the deformations and occlusions in a dataset? Some deformations and occlusions are so rare that they hardly happen in practical applications; yet we want to learn a method invariant to such situations. Wang et al. [346] used GANs to generate instances with deformations and occlusions. The aim of the adversary is to generate instances that are difficult for the object detector to classify. By using a segmentor and GANs, Segan [79] detected objects occluded by other objects in an image. To deal with the small object detection problem, Li et al. [80] proposed perceptual GAN and Bai et al. [81] proposed an end-to-end multi-task GAN (MTGAN).

5.1.5 Video applications

Reference [82] is the first paper to use GANs for video generation. Villegas et al. [347] proposed a deep neural network for the prediction of future frames in natural video sequences using GANs. Denton and Birodkar [83] proposed a new model named disentangled representation net (DR-NET) that learns disentangled image representations from video based on GANs. A novel video-to-video synthesis approach (video2video) under the generative adversarial learning framework was proposed in [85]. MoCoGan [86] is proposed to decompose motion and content for video generation [348]–[350].



[Goodfellow et al., 2014] University of Montreal | [Radford et al., 2015] Facebook AI Research | [Roth et al., 2017] Microsoft and ETHZ | [Karras et al., 2018] NVIDIA

Fig. 7: Face image synthesis.

(a) photo (b) portrait drawings

Fig. 8: Given a photo such as (a), APDrawingGAN can produce the corresponding artistic portrait drawings (b).

GANs have also been used in other video applications such as video prediction [84], [351], [352] and video retargeting [353].

5.1.6 Other image and vision applications

GANs have been utilized in other image processing and computer vision tasks [354]–[357] such as object transfiguration [358], [359], semantic segmentation [360], visual saliency prediction [361], object tracking [362], [363], image dehazing [364]–[366], natural image matting [367], image inpainting [368], [369], image fusion [370], image completion [371], and image classification [372].

Creswell et al. [373] show that the representations learned by GANs can also be used for retrieval. GANs have also been used for anticipating where people will look [374], [375].

5.2 Sequential data

GANs also have achievements in sequential data such as natural language, music, speech, voice [376], [377], and time series [378]–[381].

Natural language processing (NLP): IRGAN [88], [89] is proposed for information retrieval (IR). Li et al. [382] used adversarial learning for neural dialogue generation. GANs have also been used in text generation [87], [383]–[385] and speech language processing [94]. Kbgan [386] is proposed to generate high-quality negative examples and used in knowledge graph embeddings. Adversarial REward Learning (AREL) [387] is proposed for visual storytelling. DSGAN [388] is proposed for distant supervision relation extraction. ScratchGAN [389] is proposed to train a language GAN from scratch – without maximum likelihood pretraining.

Qiao et al. [90] learn text-to-image generation by redescription and text conditioned auxiliary classifier GAN (TAC-GAN) [390] is also proposed for text to image. GANs have been widely used in image-to-text (image caption) [391], [392], too.

Furthermore, GANs have been widely utilized in other NLP applications such as question answer selection [393], [394], poetry generation [395], talent-job fit [396], and review detection and geneneration [397], [398].

Music: GANs have been used for generating music such as continuous RNN-GAN (C-RNN-GAN) [91], Object-Reinforced GAN (ORGAN) [92], and SeqGAN [93], [94].

Speech and Audio: GANs have been used for speech and audio analysis such as synthesis [399]–[401], enhancement [402], and recognition [403], .

5.3 Other applications

Medical field: GANs have been widely utilized in medical field such as generating and designing DNA [404], [405], drug discovery [406], generating multi-label discrete patient records [407], medical image processing [408]–[415], dental restorations [416], and doctor recommendation [417].

Data science: GANs have been used in data generating [214], [418]–[426], neural networks generating [427], data augmentation [428], [429], spatial representation learning [430], network embedding [431], heterogeneous information networks [432], and mobile user profiling [433].

GANs have been widely used in many other areas such as malware detection [434], chess game playing [435], steganography [436]–[439], privacy-preserving [440]–[442], social robot [443], and network pruning [444], [445].

6 Open Research Problems

Because GANs have become popular throughout the deep learning area, its limitations have recently been improved [446], [447]. There are still open research problems for GANs.

GANs for discrete data: GANs rely on the generated samples being completely differentiable with respect to the generative parameters. Therefore, GANs cannot produce discrete data directly, such as hashing code and one-hot word. Solving this problem is very important since it could unlock the potential of GANs for NLP and hashing. Goodfellow [103] suggested three ways to solve this problem: using Gumbel-softmax [448], [449] or the concrete distribution [450]; utilizing the REINFORCE algorithm [451]; training the generator to sample continuous values that can be transformed to discrete ones (such as sampling word embeddings directly).

There are other methods towards this research direction. Song et al. [278] used a continuous function to approximate the sign function for hashing code. Gulrajani et al. [19]

modelled discrete data with a continuous generator. Hjelm et al. [452] introduced an algorithm for training GANs with discrete data that utilizes the estimated difference measure from the discriminator to compute importance weights for generated samples, and thus providing a policy gradient for training the generator. Other related work can be found in [453], [454]. More work needs to be done in this interesting area.

New Divergences: New families of Integral Probability Metrics (IPMs) for training GANs such as Fisher GAN [455], [456], mean and covariance feature matching GAN (McGan) [457], and Sobolev GAN [458], have been proposed. Are there any other interesting classes of divergences? This deserves further study.

Estimation uncertainty: Generally speaking, as we have more data, uncertainty estimation reduces. GANs do not give the distribution that generated the training examples and GANs aim to generate new samples that come from the same distribution of the training examples. Therefore, GANs have neither a likelihood nor a well-defined posterior. There are early attempts towards this research direction such as Bayesian GAN [459]. Although we can use GANs to generate data, how can we measure the uncertainty of the well-trained generator? This is another interesting future issue.

Theory: As for generalization, Zhang et al. [460] developed generalization bounds between the true distribution and learned distribution under different evaluation metrics. When evaluated with neural distance, the bounds in [460] show that generalization is guaranteed as long as the discriminator set is small enough, regardless of the size of the hypothesis set or generator. Arora et al. [306] proposed a novel test for estimating support size using the birthday paradox of discrete probability and show that GAN does suffer mode collapse even when images are of higher visual quality. More deep theoretical research is well worth studying. How do we test for generalization empirically? Useful theory should enable choice of model class, capacity, and architectures. This is an interesting issue to be investigated in future work.

Others: There are other important research problems for GANs such as evaluation (detailed in Subsection 3.4) and mode collapse (detailed in Subsection 4.2)

7 Conclusions

This paper presents a comprehensive review of various aspects of GANs. We elaborate on several perspectives, i.e., algorithm, theory, applications, and open research problems. We believe this survey will help readers to gain a thorough understanding of the GANs research area.

Acknowledgments

The authors would like to thank the NetEase course taught by Shuang Yang, Ian Good fellow's invited talk at AAAI 19, CVPR 2018 tutorial on GANs, Sebastian Nowozin's MLSS 2018 GAN lecture materials. The authors also would like to thank the helpful discussions with group members of Umich Yelab and Foreseer research group.

References

[1] L. J. Ratliff, S. A. Burden, and S. S. Sastry, "Characterization and computation of local nash equilibria in continuous games," in *Annual Allerton Conference on Communication, Control, and Computing*, pp. 917–924, 2013.

[2] J. Schmidhuber, "Learning factorial codes by predictability minimization," *Neural Computation*, vol. 4, no. 6, pp. 863–879, 1992.

[3] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Neural Information Processing Systems*, pp. 2672–2680, 2014.

[4] J. Schmidhuber, "Unsupervised minimax: Adversarial curiosity, generative adversarial networks, and predictability minimization," *arXiv preprint arXiv:1906.04493*, 2019.

[5] X. Wu, K. Xu, and P. Hall, "A survey of image synthesis and editing with generative adversarial networks," *Tsinghua Science and Technology*, vol. 22, no. 6, pp. 660–674, 2017.

[6] N. Torres-Reyes and S. Latifi, "Audio enhancement and synthesis using generative adversarial networks: A survey," *International Journal of Computer Applications*, vol. 182, no. 35, pp. 27–31, 2019.

[7] K. Wang, C. Gou, Y. Duan, Y. Lin, X. Zheng, and F.-Y. Wang, "Generative adversarial networks: introduction and outlook," *IEEE/CAA Journal of Automatica Sinica*, vol. 4, no. 4, pp. 588–598, 2017.

[8] Y. Hong, U. Hwang, J. Yoo, and S. Yoon, "How generative adversarial networks and their variants work: An overview," *ACM Computing Surveys*, vol. 52, no. 1, pp. 1–43, 2019.

[9] A. Creswell, T. White, V. Dumoulin, K. Arulkumaran, B. Sengupta, and A. A. Bharath, "Generative adversarial networks: An overview," *IEEE Signal Processing Magazine*, vol. 35, no. 1, pp. 53–65, 2018.

[10] Z. Wang, Q. She, and T. E. Ward, "Generative adversarial networks: A survey and taxonomy," *arXiv preprint arXiv:1906.01529*, 2019.

[11] S. Hitawala, "Comparative study on generative adversarial networks," *arXiv preprint arXiv:1801.04271*, 2018.

[12] M. Zamorski, A. Zdobylak, M. Zieba, and J. Swiatek, "Generative adversarial networks: recent developments," in *International Conference on Artificial Intelligence and Soft Computing*, pp. 248–258, Springer, 2019.

[13] Z. Pan, W. Yu, X. Yi, A. Khan, F. Yuan, and Y. Zheng, "Recent progress on generative adversarial networks (gans): A survey," *IEEE Access*, vol. 7, pp. 36322–36333, 2019.

[14] X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel, "Infogan: Interpretable representation learning by information maximizing generative adversarial nets," in *Neural Information Processing Systems*, pp. 2172–2180, 2016.

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

[16] Y. Lu, Y.-W. Tai, and C.-K. Tang, "Conditional cyclegan for attribute guided face image generation," *arXiv preprint arXiv:1705.09966*, 2017.

[17] S. Nowozin, B. Cseke, and R. Tomioka, "f-gan: Training generative neural samplers using variational divergence minimization," in *Neural Information Processing Systems*, pp. 271–279, 2016.

[18] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," in *International Conference on Machine Learning*, pp. 214–223, 2017.

[19] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville, "Improved training of wasserstein gans," in *Neural Information Processing Systems*, pp. 5767–5777, 2017.

[20] G.-J. Qi, "Loss-sensitive generative adversarial networks on lipschitz densities," *International Journal of Computer Vision*, pp. 1–23, 2019.

[21] X. Mao, Q. Li, H. Xie, R. Y. Lau, Z. Wang, and S. Paul Smolley, "Least squares generative adversarial networks," in *IEEE International Conference on Computer Vision*, pp. 2794–2802, 2017.

[22] X. Mao, Q. Li, H. Xie, R. Y. K. Lau, Z. Wang, and S. P. Smolley, "On the effectiveness of least squares generative adversarial networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 12, pp. 2947–2960, 2019.

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

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

[25] D. Tran, R. Ranganath, and D. M. Blei, "Deep and hierarchical implicit models," *arXiv preprint arXiv:1702.08896*, 2017.

[26] T. Che, Y. Li, A. P. Jacob, Y. Bengio, and W. Li, "Mode regularized generative adversarial networks," in *International Conference on Learning Representations*, 2017.

[27] L. Metz, B. Poole, D. Pfau, and J. Sohl-Dickstein, "Unrolled generative adversarial networks," *arXiv preprint arXiv:1611.02163*, 2016.

[28] A. Jolicoeur-Martineau, "The relativistic discriminator: a key element missing from standard gan," in *International Conference on Learning Representation*, 2019.

[29] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training gans," in *Neural Information Processing Systems*, pp. 2234–2242, 2016.

[30] A. Odena, C. Olah, and J. Shlens, "Conditional image synthesis with auxiliary classifier gans," in *International Conference on Machine Learning*, pp. 2642–2651, 2017.

[31] E. L. Denton, S. Chintala, R. Fergus, *et al.*, "Deep generative image models using a laplacian pyramid of adversarial networks," in *Neural Information Processing Systems*, pp. 1486–1494, 2015.

[32] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," *arXiv preprint arXiv:1511.06434*, 2015.

[33] 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.

[34] H. Zhang, T. Xu, H. Li, S. Zhang, X. Wang, X. Huang, and D. N. Metaxas, "Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks," in *IEEE International Conference on Computer Vision*, pp. 5907–5915, 2017.

[35] H. Zhang, I. Goodfellow, D. Metaxas, and A. Odena, "Self-attention generative adversarial networks," in *International Conference on Machine Learning*, pp. 7354–7363, 2019.

[36] A. Brock, J. Donahue, and K. Simonyan, "Large scale gan training for high fidelity natural image synthesis," in *International Conference on Learning representations*, 2019.

[37] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4401–4410, 2019.

[38] J. Zhao, M. Mathieu, and Y. LeCun, "Energy-based generative adversarial network," in *International Conference on Learning Representations*, 2017.

[39] D. Berthelot, T. Schumm, and L. Metz, "Began: Boundary equilibrium generative adversarial networks," *arXiv preprint arXiv:1703.10717*, 2017.

[40] J. Donahue, P. Krähenbühl, and T. Darrell, "Adversarial feature learning," *arXiv preprint arXiv:1605.09782*, 2016.

[41] V. Dumoulin, I. Belghazi, B. Poole, O. Mastropietro, A. Lamb, M. Arjovsky, and A. Courville, "Adversarially learned inference," *arXiv preprint arXiv:1606.00704*, 2016.

[42] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "It takes (only) two: Adversarial generator-encoder networks," in *AAAI Conference on Artificial Intelligence*, pp. 1250–1257, 2018.

[43] T. Nguyen, T. Le, H. Vu, and D. Phung, "Dual discriminator generative adversarial nets," in *Neural Information Processing Systems*, pp. 2670–2680, 2017.

[44] I. Durugkar, I. Gemp, and S. Mahadevan, "Generative multi-adversarial networks," in *International Conference on Learning Representations*, 2017.

[45] Q. Hoang, T. D. Nguyen, T. Le, and D. Phung, "Multi-generator generative adversarial nets," *arXiv preprint arXiv:1708.02556*, 2017.

[46] A. Ghosh, V. Kulharia, V. P. Namboodiri, P. H. Torr, and P. K. Dokania, "Multi-agent diverse generative adversarial networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 8513–8521, 2018.

[47] M.-Y. Liu and O. Tuzel, "Coupled generative adversarial networks," in *Neural Information Processing Systems*, pp. 469–477, 2016.

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

[49] T. Miyato, S.-i. Maeda, S. Ishii, and M. Koyama, "Virtual adversarial training: a regularization method for supervised and semi-supervised learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 8, pp. 1979–1993, 2019.

[50] Z. Gan, L. Chen, W. Wang, Y. Pu, Y. Zhang, H. Liu, C. Li, and L. Carin, "Triangle generative adversarial networks," in *Neural Information Processing Systems*, pp. 5247–5256, 2017.

[51] L. Chongxuan, T. Xu, J. Zhu, and B. Zhang, "Triple generative adversarial nets," in *Neural Information Processing Systems*, pp. 4088–4098, 2017.

[52] H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, and M. Marchand, "Domain-adversarial neural networks," *arXiv preprint arXiv:1412.4446*, 2014.

[53] 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*, pp. 2223–2232, 2017.

[54] T. Kim, M. Cha, H. Kim, J. K. Lee, and J. Kim, "Learning to discover cross-domain relations with generative adversarial networks," in *International Conference on Machine Learning*, pp. 1857–1865, 2017.

[55] Z. Yi, H. Zhang, P. Tan, and M. Gong, "Dualgan: Unsupervised dual learning for image-to-image translation," in *International Conference on Computer Vision*, pp. 2849–2857, 2017.

[56] Y. Choi, M. Choi, M. Kim, J.-W. Ha, S. Kim, and J. Choo, "Stargan: Unified generative adversarial networks for multi-domain image-to-image translation," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 8789–8797, 2018.

[57] J. Hoffman, E. Tzeng, T. Park, J.-Y. Zhu, P. Isola, K. Saenko, A. A. Efros, and T. Darrell, "Cycada: Cycle-consistent adversarial domain adaptation," in *International Conference on Machine Learning*, 2018.

[58] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell, "Adversarial discriminative domain adaptation," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7167–7176, 2017.

[59] S. Wang and L. Zhang, "Catgan: Coupled adversarial transfer for domain generation," *arXiv preprint arXiv:1711.08904*, 2017.

[60] Y. Zhang, Z. Qiu, T. Yao, D. Liu, and T. Mei, "Fully convolutional adaptation networks for semantic segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6810–6818, 2018.

[61] 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*, pp. 3722–3731, 2017.

[62] J. Song, H. Ren, D. Sadigh, and S. Ermon, "Multi-agent generative adversarial imitation learning," in *Neural Information Processing Systems*, pp. 7461–7472, 2018.

[63] 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*, pp. 4681–4690, 2017.

[64] X. Wang, K. Yu, S. Wu, J. Gu, Y. Liu, C. Dong, Y. Qiao, and C. Change Loy, "Esrgan: Enhanced super-resolution generative adversarial networks," in *European Conference on Computer Vision*, pp. 63–79, 2018.

[65] Y. Yuan, S. Liu, J. Zhang, Y. Zhang, C. Dong, and L. Lin, "Unsupervised image super-resolution using cycle-in-cycle generative adversarial networks," in *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 701–710, 2018.

[66] J. Guan, C. Pan, S. Li, and D. Yu, "Srdgan: learning the noise prior for super resolution with dual generative adversarial networks," *arXiv preprint arXiv:1903.11821*, 2019.

[67] Z. Ding, X.-Y. Liu, M. Yin, W. Liu, and L. Kong, "Tgan: Deep tensor generative adversarial nets for large image generation," *arXiv preprint arXiv:1901.09953*, 2019.

[68] L. Q. Tran, X. Yin, and X. Liu, "Representation learning by rotating your faces," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.

[69] R. Huang, S. Zhang, T. Li, and R. He, "Beyond face rotation: Global and local perception gan for photorealistic and identity preserving frontal view synthesis," in *International Conference on Computer Vision*, pp. 2439–2448, 2017.

[70] L. Ma, X. Jia, Q. Sun, B. Schiele, T. Tuytelaars, and L. Van Gool, "Pose guided person image generation," in *Neural Information Processing Systems*, pp. 406–416, 2017.

[71] W. Jiang, S. Liu, C. Gao, J. Cao, R. He, J. Feng, and S. Yan, "Psgan: Pose-robust spatial-aware gan for customizable makeup transfer," *arXiv preprint arXiv:1909.06956*, 2019.

[72] R. Yi, Y.-J. Liu, Y.-K. Lai, and P. L. Rosin, "Apdrawinggan: Generating artistic portrait drawings from face photos with hier-

archical gans," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 10743–10752, 2019.
[73] 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*, pp. 597–613, 2016.
[74] A. Brock, T. Lim, J. M. Ritchie, and N. Weston, "Neural photo editing with introspective adversarial networks," *arXiv preprint arXiv:1609.07093*, 2016.
[75] T. Park, M.-Y. Liu, T.-C. Wang, and J.-Y. Zhu, "Semantic image synthesis with spatially-adaptive normalization," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2337–2346, 2019.
[76] C. Li and M. Wand, "Precomputed real-time texture synthesis with markovian generative adversarial networks," in *European Conference on Computer Vision*, pp. 702–716, 2016.
[77] N. Jetchev, U. Bergmann, and R. Vollgraf, "Texture synthesis with spatial generative adversarial networks," *arXiv preprint arXiv:1611.08207*, 2016.
[78] U. Bergmann, N. Jetchev, and R. Vollgraf, "Learning texture manifolds with the periodic spatial gan," in *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 469–477, JMLR. org, 2017.
[79] K. Ehsani, R. Mottaghi, and A. Farhadi, "Segan: Segmenting and generating the invisible," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6144–6153, 2018.
[80] J. Li, X. Liang, Y. Wei, T. Xu, J. Feng, and S. Yan, "Perceptual generative adversarial networks for small object detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1222–1230, 2017.
[81] Y. Bai, Y. Zhang, M. Ding, and B. Ghanem, "Sod-mtgan: Small object detection via multi-task generative adversarial network," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 206–221, 2018.
[82] C. Vondrick, H. Pirsiavash, and A. Torralba, "Generating videos with scene dynamics," in *Neural Information Processing Systems*, pp. 613–621, 2016.
[83] E. L. Denton *et al.*, "Unsupervised learning of disentangled representations from video," in *Neural Information Processing Systems*, pp. 4414–4423, 2017.
[84] J. Walker, K. Marino, A. Gupta, and M. Hebert, "The pose knows: Video forecasting by generating pose futures," in *IEEE International Conference on Computer Vision*, pp. 3332–3341, 2017.
[85] T.-C. Wang, M.-Y. Liu, J.-Y. Zhu, G. Liu, A. Tao, J. Kautz, and B. Catanzaro, "Video-to-video synthesis," in *Neural Information Processing Systems*, pp. 1152–1164, 2018.
[86] S. Tulyakov, M.-Y. Liu, X. Yang, and J. Kautz, "Mocogan: Decomposing motion and content for video generation," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1526–1535, 2018.
[87] K. Lin, D. Li, X. He, Z. Zhang, and M.-T. Sun, "Adversarial ranking for language generation," in *Neural Information Processing Systems*, pp. 3155–3165, 2017.
[88] J. Wang, L. Yu, W. Zhang, Y. Gong, Y. Xu, B. Wang, P. Zhang, and D. Zhang, "Irgan: A minimax game for unifying generative and discriminative information retrieval models," in *International ACM SIGIR conference on Research and Development in Information Retrieval*, pp. 515–524, ACM, 2017.
[89] S. Lu, Z. Dou, X. Jun, J.-Y. Nie, and J.-R. Wen, "Psgan: A minimax game for personalized search with limited and noisy click data," in *ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 555–564, 2019.
[90] T. Qiao, J. Zhang, D. Xu, and D. Tao, "Mirrorgan: Learning text-to-image generation by redescription," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1505–1514, 2019.
[91] O. Mogren, "C-rnn-gan: Continuous recurrent neural networks with adversarial training," *arXiv preprint arXiv:1611.09904*, 2016.
[92] G. L. Guimaraes, B. Sanchez-Lengeling, C. Outeiral, P. L. C. Farias, and A. Aspuru-Guzik, "Objective-reinforced generative adversarial networks (organ) for sequence generation models," *arXiv preprint arXiv:1705.10843*, 2017.
[93] S.-g. Lee, U. Hwang, S. Min, and S. Yoon, "A seqgan for polyphonic music generation," *arXiv preprint arXiv:1710.11418*, 2017.
[94] L. Yu, W. Zhang, J. Wang, and Y. Yu, "Seqgan: Sequence generative adversarial nets with policy gradient," in *AAAI Conference on Artificial Intelligence*, pp. 2852–2858, 2017.
[95] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.
[96] D. J. Rezende, S. Mohamed, and D. Wierstra, "Stochastic backpropagation and approximate inference in deep generative models," *arXiv preprint arXiv:1401.4082*, 2014.
[97] G. E. Hinton, T. J. Sejnowski, and D. H. Ackley, *Boltzmann machines: Constraint satisfaction networks that learn*. Carnegie-Mellon University, Department of Computer Science Pittsburgh, 1984.
[98] D. H. Ackley, G. E. Hinton, and T. J. Sejnowski, "A learning algorithm for boltzmann machines," *Cognitive science*, vol. 9, no. 1, pp. 147–169, 1985.
[99] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural computation*, vol. 18, no. 7, pp. 1527–1554, 2006.
[100] A. Nguyen, A. Dosovitskiy, J. Yosinski, T. Brox, and J. Clune, "Synthesizing the preferred inputs for neurons in neural networks via deep generator networks," in *Neural Information Processing Systems*, pp. 3387–3395, 2016.
[101] Y. Bengio, E. Laufer, G. Alain, and J. Yosinski, "Deep generative stochastic networks trainable by backprop," in *International Conference on Machine Learning*, pp. 226–234, 2014.
[102] Y. Bengio, L. Yao, G. Alain, and P. Vincent, "Generalized denoising auto-encoders as generative models," in *Neural Information Processing Systems*, pp. 899–907, 2013.
[103] I. Goodfellow, "Nips 2016 tutorial: Generative adversarial networks," *arXiv preprint arXiv:1701.00160*, 2016.
[104] T. Salimans, A. Karpathy, X. Chen, and D. P. Kingma, "Pixelcnn++: Improving the pixelcnn with discretized logistic mixture likelihood and other modifications," *arXiv preprint arXiv:1701.05517*, 2017.
[105] B. J. Frey, G. E. Hinton, and P. Dayan, "Does the wake-sleep algorithm produce good density estimators?," in *Advances in neural information processing systems*, pp. 661–667, 1996.
[106] B. J. Frey, J. F. Brendan, and B. J. Frey, *Graphical models for machine learning and digital communication*. MIT press, 1998.
[107] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, *et al.*, "Mastering the game of go with deep neural networks and tree search," *Nature*, vol. 529, no. 7587, pp. 484–489, 2016.
[108] K. Eykholt, I. Evtimov, E. Fernandes, B. Li, A. Rahmati, C. Xiao, A. Prakash, T. Kohno, and D. Song, "Robust physical-world attacks on deep learning visual classification," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1625–1634, 2018.
[109] A. Kurakin, I. Goodfellow, and S. Bengio, "Adversarial examples in the physical world," *arXiv preprint arXiv:1607.02533*, 2016.
[110] G. Elsayed, S. Shankar, B. Cheung, N. Papernot, A. Kurakin, I. Goodfellow, and J. Sohl-Dickstein, "Adversarial examples that fool both computer vision and time-limited humans," in *Neural Information Processing Systems*, pp. 3910–3920, 2018.
[111] X. Jia, X. Wei, X. Cao, and H. Foroosh, "Comdefend: An efficient image compression model to defend adversarial examples," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6084–6092, 2019.
[112] A. Athalye, N. Carlini, and D. Wagner, "Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples," in *International Conference on Machine Learning*, 2018.
[113] D. Zügner, A. Akbarnejad, and S. Günnemann, "Adversarial attacks on neural networks for graph data," in *SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 2847–2856, 2018.
[114] Y. Dong, F. Liao, T. Pang, H. Su, J. Zhu, X. Hu, and J. Li, "Boosting adversarial attacks with momentum," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 9185–9193, 2018.
[115] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," *arXiv preprint arXiv:1312.6199*, 2013.
[116] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," *arXiv preprint arXiv:1412.6572*, 2014.
[117] J. Kos, I. Fischer, and D. Song, "Adversarial examples for generative models," in *IEEE Security and Privacy Workshops*, pp. 36–42, 2018.
[118] P. Samangouei, M. Kabkab, and R. Chellappa, "Defense-gan: Protecting classifiers against adversarial attacks using generative models," *arXiv preprint arXiv:1805.06605*, 2018.
[119] N. Akhtar and A. Mian, "Threat of adversarial attacks on deep learning in computer vision: A survey," *IEEE Access*, vol. 6, pp. 14410–14430, 2018.

[120] S. Shen, G. Jin, K. Gao, and Y. Zhang, "Ape-gan: Adversarial perturbation elimination with gan," *arXiv preprint arXiv:1707.05474*, 2017.

[121] H. Lee, S. Han, and J. Lee, "Generative adversarial trainer: Defense to adversarial perturbations with gan," *arXiv preprint arXiv:1705.03387*, 2017.

[122] L. Huang, A. D. Joseph, B. Nelson, B. I. Rubinstein, and J. Tygar, "Adversarial machine learning," in *ACM Workshop on Security and Artificial Intelligence*, pp. 43–58, 2011.

[123] I. J. Goodfellow, "On distinguishability criteria for estimating generative models," *arXiv preprint arXiv:1412.6515*, 2014.

[124] M. Kahng, N. Thorat, D. H. P. Chau, F. B. Viégas, and M. Wattenberg, "Gan lab: Understanding complex deep generative models using interactive visual experimentation," *IEEE Transactions on Visualization and Computer Graphics*, vol. 25, no. 1, pp. 1–11, 2018.

[125] D. Bau, J.-Y. Zhu, H. Strobelt, B. Zhou, J. B. Tenenbaum, W. T. Freeman, and A. Torralba, "Gan dissection: Visualizing and understanding generative adversarial networks," in *International Conference on Learning Representations*, 2019.

[126] M. Kocaoglu, C. Snyder, A. G. Dimakis, and S. Vishwanath, "Causalgan: Learning causal implicit generative models with adversarial training," *arXiv preprint arXiv:1709.02023*, 2017.

[127] S. Feizi, F. Farnia, T. Ginart, and D. Tse, "Understanding gans: the lqg setting," *arXiv preprint arXiv:1710.10793*, 2017.

[128] F. Farnia and D. Tse, "A convex duality framework for gans," in *Neural Information Processing Systems*, pp. 5248–5258, 2018.

[129] J. Zhao, J. Li, Y. Cheng, T. Sim, S. Yan, and J. Feng, "Understanding humans in crowded scenes: Deep nested adversarial learning and a new benchmark for multi-human parsing," in *ACM Multimedia Conference on Multimedia Conference*, pp. 792–800, ACM, 2018.

[130] A. Jahanian, L. Chai, and P. Isola, "On the"steerability" of generative adversarial networks," *arXiv preprint arXiv:1907.07171*, 2019.

[131] B. Zhu, J. Jiao, and D. Tse, "Deconstructing generative adversarial networks," *arXiv preprint arXiv:1901.09465*, 2019.

[132] K. B. Kancharagunta and S. R. Dubey, "Csgan: cyclic-synthesized generative adversarial networks for image-to-image transformation," *arXiv preprint arXiv:1901.03554*, 2019.

[133] Y. Wu, J. Donahue, D. Balduzzi, K. Simonyan, and T. Lillicrap, "Logan: Latent optimisation for generative adversarial networks," *arXiv preprint arXiv:1912.00953*, 2019.

[134] T. Kurutach, A. Tamar, G. Yang, S. J. Russell, and P. Abbeel, "Learning plannable representations with causal infogan," in *Neural Information Processing Systems*, pp. 8733–8744, 2018.

[135] A. Spurr, E. Aksan, and O. Hilliges, "Guiding infogan with semi-supervision," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 119–134, Springer, 2017.

[136] A. Nguyen, J. Clune, Y. Bengio, A. Dosovitskiy, and J. Yosinski, "Plug & play generative networks: Conditional iterative generation of images in latent space," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4467–4477, 2017.

[137] 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*, pp. 1–10, 2016.

[138] S. Hong, D. Yang, J. Choi, and H. Lee, "Inferring semantic layout for hierarchical text-to-image synthesis," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7986–7994, 2018.

[139] S. E. Reed, Z. Akata, S. Mohan, S. Tenka, B. Schiele, and H. Lee, "Learning what and where to draw," in *Neural Information Processing Systems*, pp. 217–225, 2016.

[140] H. Zhang, T. Xu, H. Li, S. Zhang, X. Wang, X. Huang, and D. Metaxas, "Stackgan++: Realistic image synthesis with stacked generative adversarial networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 8, pp. 1947–1962, 2019.

[141] X. Huang, Y. Li, O. Poursaeed, J. Hopcroft, and S. Belongie, "Stacked generative adversarial networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5077–5086, 2017.

[142] J. Gauthier, "Conditional generative adversarial nets for convolutional face generation," *Class Project for Stanford CS231N: Convolutional Neural Networks for Visual Recognition, Winter semester*, vol. 2014, no. 5, p. 2, 2014.

[143] G. Antipov, M. Baccouche, and J.-L. Dugelay, "Face aging with conditional generative adversarial networks," in *2017 IEEE International Conference on Image Processing (ICIP)*, pp. 2089–2093, IEEE, 2017.

[144] H. Tang, D. Xu, N. Sebe, Y. Wang, J. J. Corso, and Y. Yan, "Multi-channel attention selection gan with cascaded semantic guidance for cross-view image translation," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2417–2426, 2019.

[145] L. Karacan, Z. Akata, A. Erdem, and E. Erdem, "Learning to generate images of outdoor scenes from attributes and semantic layouts," *arXiv preprint arXiv:1612.00215*, 2016.

[146] B. Dai, S. Fidler, R. Urtasun, and D. Lin, "Towards diverse and natural image descriptions via a conditional gan," in *IEEE International Conference on Computer Vision*, pp. 2970–2979, 2017.

[147] S. Yao, T. M. Hsu, J.-Y. Zhu, J. Wu, A. Torralba, B. Freeman, and J. Tenenbaum, "3d-aware scene manipulation via inverse graphics," in *Neural Information Processing Systems*, pp. 1887–1898, 2018.

[148] G. G. Chrysos, J. Kossaifi, and S. Zafeiriou, "Robust conditional generative adversarial networks," *arXiv preprint arXiv:1805.08657*, 2018.

[149] K. K. Thekumparampil, A. Khetan, Z. Lin, and S. Oh, "Robustness of conditional gans to noisy labels," in *Neural Information Processing Systems*, pp. 10271–10282, 2018.

[150] Q. Mao, H.-Y. Lee, H.-Y. Tseng, S. Ma, and M.-H. Yang, "Mode seeking generative adversarial networks for diverse image synthesis," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1429–1437, 2019.

[151] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, 2009.

[152] G. Perarnau, J. Van De Weijer, B. Raducanu, and J. M. Álvarez, "Invertible conditional gans for image editing," *arXiv preprint arXiv:1611.06355*, 2016.

[153] M. Saito, E. Matsumoto, and S. Saito, "Temporal generative adversarial nets with singular value clipping," in *IEEE International Conference on Computer Vision*, pp. 2830–2839, 2017.

[154] K. Sricharan, R. Bala, M. Shreve, H. Ding, K. Saketh, and J. Sun, "Semi-supervised conditional gans," *arXiv preprint arXiv:1708.05789*, 2017.

[155] T. Miyato and M. Koyama, "cgans with projection discriminator," *arXiv preprint arXiv:1802.05637*, 2018.

[156] 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*, pp. 1125–1134, 2017.

[157] T.-C. Wang, M.-Y. Liu, J.-Y. Zhu, A. Tao, J. Kautz, and B. Catanzaro, "High-resolution image synthesis and semantic manipulation with conditional gans," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 8798–8807, 2018.

[158] C. Li, H. Liu, C. Chen, Y. Pu, L. Chen, R. Henao, and L. Carin, "Alice: Towards understanding adversarial learning for joint distribution matching," in *Neural Information Processing Systems*, pp. 5495–5503, 2017.

[159] L. C. Tiao, E. V. Bonilla, and F. Ramos, "Cycle-consistent adversarial learning as approximate bayesian inference," *arXiv preprint arXiv:1806.01771*, 2018.

[160] I. Csiszár, P. C. Shields, *et al.*, "Information theory and statistics: A tutorial," *Foundations and Trends® in Communications and Information Theory*, vol. 1, no. 4, pp. 417–528, 2004.

[161] D. J. Im, H. Ma, G. Taylor, and K. Branson, "Quantitatively evaluating gans with divergences proposed for training," in *International Conference on Learning Representation*, 2018.

[162] M. Uehara, I. Sato, M. Suzuki, K. Nakayama, and Y. Matsuo, "Generative adversarial nets from a density ratio estimation perspective," *arXiv preprint arXiv:1610.02920*, 2016.

[163] B. K. Sriperumbudur, A. Gretton, K. Fukumizu, B. Schölkopf, and G. R. Lanckriet, "Hilbert space embeddings and metrics on probability measures," *Journal of Machine Learning Research*, vol. 11, no. Apr, pp. 1517–1561, 2010.

[164] A. Uppal, S. Singh, and B. Poczos, "Nonparametric density estimation & convergence of gans under besov ipm losses," in *Neural Information Processing Systems*, pp. 9086–9097, 2019.

[165] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. Smola, "A kernel two-sample test," *Journal of Machine Learning Research*, vol. 13, no. Mar, pp. 723–773, 2012.

[166] G. K. Dziugaite, D. M. Roy, and Z. Ghahramani, "Training generative neural networks via maximum mean discrepancy optimization," *arXiv preprint arXiv:1505.03906*, 2015.

[167] C.-L. Li, W.-C. Chang, Y. Cheng, Y. Yang, and B. Póczos, "Mmd gan: Towards deeper understanding of moment matching network," in *Neural Information Processing Systems*, pp. 2203–2213, 2017.

[168] Y. Li, K. Swersky, and R. Zemel, "Generative moment matching networks," in *International Conference on Machine Learning*, pp. 1718–1727, 2015.

[169] D. J. Sutherland, H.-Y. Tung, H. Strathmann, S. De, A. Ramdas, A. Smola, and A. Gretton, "Generative models and model criticism via optimized maximum mean discrepancy," *arXiv preprint arXiv:1611.04488*, 2016.

[170] N. Kodali, J. Abernethy, J. Hays, and Z. Kira, "On convergence and stability of gans," *arXiv preprint arXiv:1705.07215*, 2017.

[171] J. Wu, Z. Huang, J. Thoma, D. Acharya, and L. Van Gool, "Wasserstein divergence for gans," in *European Conference on Computer Vision*, pp. 653–668, 2018.

[172] M. G. Bellemare, I. Danihelka, W. Dabney, S. Mohamed, B. Lakshminarayanan, S. Hoyer, and R. Munos, "The cramer distance as a solution to biased wasserstein gradients," *arXiv preprint arXiv:1705.10743*, 2017.

[173] H. Petzka, A. Fischer, and D. Lukovnicov, "On the regularization of wasserstein gans," in *International Conference on Learning Representations*, pp. 1–24, 2018.

[174] F. Juefei-Xu, V. N. Boddeti, and M. Savvides, "Gang of gans: Generative adversarial networks with maximum margin ranking," *arXiv preprint arXiv:1704.04865*, 2017.

[175] C.-C. Hsu, H.-T. Hwang, Y.-C. Wu, Y. Tsao, and H.-M. Wang, "Voice conversion from unaligned corpora using variational autoencoding wasserstein generative adversarial networks," in *Interspeech*, pp. 3364–3368, 2017.

[176] J. Adler and S. Lunz, "Banach wasserstein gan," in *Neural Information Processing Systems*, pp. 6754–6763, 2018.

[177] Y.-S. Chen, Y.-C. Wang, M.-H. Kao, and Y.-Y. Chuang, "Deep photo enhancer: Unpaired learning for image enhancement from photographs with gans," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6306–6314, 2018.

[178] S. Athey, G. Imbens, J. Metzger, and E. Munro, "Using wasserstein generative adversial networks for the design of monte carlo simulations," *arXiv preprint arXiv:1909.02210*, 2019.

[179] M. Arjovsky and L. Bottou:, "Towards principled methods for training generative adversarial networks," in *International Conference on Learning Representations*, 2017.

[180] A. Yadav, S. Shah, Z. Xu, D. Jacobs, and T. Goldstein, "Stabilizing adversarial nets with prediction methods," *arXiv preprint arXiv:1705.07364*, 2017.

[181] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "Gans trained by a two time-scale update rule converge to a local nash equilibrium," in *Neural Information Processing Systems*, pp. 6626–6637, 2017.

[182] K. J. Liang, C. Li, G. Wang, and L. Carin, "Generative adversarial network training is a continual learning problem," *arXiv preprint arXiv:1811.11083*, 2018.

[183] H. Shin, J. K. Lee, J. Kim, and J. Kim, "Continual learning with deep generative replay," in *Advances in Neural Information Processing Systems*, pp. 2990–2999, 2017.

[184] M. Lin, "Softmax gan," *arXiv preprint arXiv:1704.06191*, 2017.

[185] Y. LeCun, S. Chopra, R. Hadsell, M. Ranzato, and F. Huang, "A tutorial on energy-based learning," *Predicting structured data*, vol. 1, no. 0, 2006.

[186] J. Zhao, L. Xiong, P. K. Jayashree, J. Li, F. Zhao, Z. Wang, P. S. Pranata, P. S. Shen, S. Yan, and J. Feng, "Dual-agent gans for photorealistic and identity preserving profile face synthesis," in *Advances in Neural Information Processing Systems*, pp. 66–76, 2017.

[187] J. Zhao, L. Xiong, J. Li, J. Xing, S. Yan, and J. Feng, "3d-aided dual-agent gans for unconstrained face recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 10, pp. 2380–2394, 2018.

[188] R. Wang, A. Cully, H. J. Chang, and Y. Demiris, "Magan: Margin adaptation for generative adversarial networks," *arXiv preprint arXiv:1704.03817*, 2017.

[189] A. Krizhevsky, G. Hinton, *et al.*, "Learning multiple layers of features from tiny images," tech. rep., Citeseer, 2009.

[190] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, *et al.*, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[191] J. Susskind, A. Anderson, and G. E. Hinton, "The toronto face dataset," *U. Toronto, Tech. Rep. UTML TR*, vol. 1, p. 2010, 2010.

[192] P. Burt and E. Adelson, "The laplacian pyramid as a compact image code," *IEEE Transactions on Communications*, vol. 31, no. 4, pp. 532–540, 1983.

[193] T. R. Shaham, T. Dekel, and T. Michaeli, "Singan: Learning a generative model from a single natural image," in *IEEE International Conference on Computer Vision*, pp. 4570–4580, 2019.

[194] A. Shocher, S. Bagon, P. Isola, and M. Irani, "Ingan: Capturing and retargeting the "dna" of a natural image," in *IEEE International Conference on Computer Vision*, pp. 4492–4501, 2019.

[195] J. T. Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller, "Striving for simplicity: The all convolutional net," *arXiv preprint arXiv:1412.6806*, 2014.

[196] A. A. Rusu, N. C. Rabinowitz, G. Desjardins, H. Soyer, J. Kirkpatrick, K. Kavukcuoglu, R. Pascanu, and R. Hadsell, "Progressive neural networks," *arXiv preprint arXiv:1606.04671*, 2016.

[197] T. Xu, P. Zhang, Q. Huang, H. Zhang, Z. Gan, X. Huang, and X. He, "Attngan: Fine-grained text to image generation with attentional generative adversarial networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1316–1324, 2018.

[198] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila, "Analyzing and improving the image quality of stylegan," *arXiv preprint arXiv:1912.04958*, 2019.

[199] M. Lučić, M. Tschannen, M. Ritter, X. Zhai, O. Bachem, and S. Gelly, "High-fidelity image generation with fewer labels," in *International Conference on Machine Learning*, pp. 4183–4192, 2019.

[200] J. Donahue and K. Simonyan, "Large scale adversarial representation learning," *arXiv preprint arXiv:1907.02544*, 2019.

[201] G. Yildirim, N. Jetchev, R. Vollgraf, and U. Bergmann, "Generating high-resolution fashion model images wearing custom outfits," *arXiv preprint arXiv:1908.08847*, 2019.

[202] A. Makhzani, J. Shlens, N. Jaitly, I. Goodfellow, and B. Frey, "Adversarial autoencoders," *arXiv preprint arXiv:1511.05644*, 2015.

[203] L. Mescheder, S. Nowozin, and A. Geiger, "Adversarial variational bayes: Unifying variational autoencoders and generative adversarial networks," in *International Conference on Machine Learning*, pp. 2391–2400, 2017.

[204] X. Yu, X. Zhang, Y. Cao, and M. Xia, "Vaegan: A collaborative filtering framework based on adversarial variational autoencoders," in *International Joint Conference on Artificial Intelligence*, pp. 4206–4212, 2019.

[205] M.-Y. Liu, T. Breuel, and J. Kautz, "Unsupervised image-to-image translation networks," in *Neural Information Processing Systems*, pp. 700–708, 2017.

[206] Z. Hu, Z. Yang, R. Salakhutdinov, and E. P. Xing, "On unifying deep generative models," *arXiv preprint arXiv:1706.00550*, 2017.

[207] A. B. L. Larsen, S. K. Sønderby, H. Larochelle, and O. Winther, "Autoencoding beyond pixels using a learned similarity metric," in *International Conference on Machine Learning*, pp. 1558–1566, 2016.

[208] M. Rosca, B. Lakshminarayanan, D. Warde-Farley, and S. Mohamed, "Variational approaches for auto-encoding generative adversarial networks," *arXiv preprint arXiv:1706.04987*, 2017.

[209] S. Gurumurthy, R. Kiran Sarvadevabhatla, and R. Venkatesh Babu, "Deligan: Generative adversarial networks for diverse and limited data," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 166–174, 2017.

[210] X. B. Peng, A. Kanazawa, S. Toyer, P. Abbeel, and S. Levine, "Variational discriminator bottleneck: Improving imitation learning, inverse rl, and gans by constraining information flow," in *International Conference on Learning Representations*, 2019.

[211] I. Albuquerque, J. Monteiro, T. Doan, B. Considine, T. Falk, and I. Mitliagkas, "Multi-objective training of generative adversarial networks with multiple discriminators," *arXiv preprint arXiv:1901.08680*, 2019.

[212] K. Wang and X. Wan, "Sentigan: Generating sentimental texts via mixture adversarial networks.," in *International Joint Conference on Artificial Intelligence*, pp. 4446–4452, 2018.

[213] X. Wang and A. Gupta, "Generative image modeling using style and structure adversarial networks," in *European Conference on Computer Vision*, pp. 318–335, 2016.

[214] D. Xu, Y. Wu, S. Yuan, L. Zhang, and X. Wu, "Achieving causal fairness through generative adversarial networks," in *International Joint Conference on Artificial Intelligence*, pp. 1452–1458, 2019.

[215] Z. Wang, G. Healy, A. F. Smeaton, and T. E. Ward, "Use of neural signals to evaluate the quality of generative adversarial network performance in facial image generation," *arXiv preprint arXiv:1811.04172*, 2018.

[216] Z. Wang, Q. She, A. F. Smeaton, T. E. Ward, and G. Healy, "Neuroscore: A brain-inspired evaluation metric for generative adversarial networks," *arXiv preprint arXiv:1905.04243*, 2019.

[217] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2818–2826, 2016.

[218] I. Danihelka, B. Lakshminarayanan, B. Uria, D. Wierstra, and P. Dayan, "Comparison of maximum likelihood and gan-based training of real nvps," *arXiv preprint arXiv:1705.05263*, 2017.

[219] Q. Xu, G. Huang, Y. Yuan, C. Guo, Y. Sun, F. Wu, and K. Weinberger, "An empirical study on evaluation metrics of generative adversarial networks," *arXiv preprint arXiv:1806.07755*, 2018.

[220] M. Bińkowski, D. J. Sutherland, M. Arbel, and A. Gretton, "Demystifying mmd gans," *arXiv preprint arXiv:1801.01401*, 2018.

[221] Z. Wang, A. C. Bovik, H. R. Sheikh, E. P. Simoncelli, *et al.*, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.

[222] Z. Wang, E. P. Simoncelli, and A. C. Bovik, "Multiscale structural similarity for image quality assessment," in *Asilomar Conference on Signals, Systems & Computers*, vol. 2, pp. 1398–1402, 2003.

[223] W. Fedus, M. Rosca, B. Lakshminarayanan, A. M. Dai, S. Mohamed, and I. Goodfellow, "Many paths to equilibrium: Gans do not need to decrease a divergence at every step," 2018.

[224] K. Kurach, M. Lučić, X. Zhai, M. Michalski, and S. Gelly, "A large-scale study on regularization and normalization in gans," in *International Conference on Machine Learning*, pp. 3581–3590, 2019.

[225] L. Theis, A. v. d. Oord, and M. Bethge, "A note on the evaluation of generative models," in *International Conference on Learning Representations*, 2015.

[226] M. Lucic, K. Kurach, M. Michalski, S. Gelly, and O. Bousquet, "Are gans created equal? a large-scale study," in *Neural Information Processing Systems*, pp. 700–709, 2018.

[227] A. Borji, "Pros and cons of gan evaluation measures," *Computer Vision and Image Understanding*, vol. 179, pp. 41–65, 2019.

[228] E. Denton, S. Gross, and R. Fergus, "Semi-supervised learning with context-conditional generative adversarial networks," *arXiv preprint arXiv:1611.06430*, 2016.

[229] M. Ding, J. Tang, and J. Zhang, "Semi-supervised learning on graphs with generative adversarial nets," in *CM International Conference on Information and Knowledge Management*, pp. 913–922, ACM, 2018.

[230] A. Odena, "Semi-supervised learning with generative adversarial networks," *arXiv preprint arXiv:1606.01583*, 2016.

[231] Z. Dai, Z. Yang, F. Yang, W. W. Cohen, and R. R. Salakhutdinov, "Good semi-supervised learning that requires a bad gan," in *Neural Information Processing Systems*, pp. 6510–6520, 2017.

[232] A. Madani, M. Moradi, A. Karargyris, and T. Syeda-Mahmood, "Semi-supervised learning with generative adversarial networks for chest x-ray classification with ability of data domain adaptation," in *International Symposium on Biomedical Imaging*, pp. 1038–1042, 2018.

[233] T. Chen, X. Zhai, M. Ritter, M. Lucic, and N. Houlsby, "Self-supervised generative adversarial networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[234] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky, "Domain-adversarial training of neural networks," *The Journal of Machine Learning Research*, vol. 17, no. 1, pp. 2096–2030, 2016.

[235] A. M. Lamb, A. G. A. P. Goyal, Y. Zhang, S. Zhang, A. C. Courville, and Y. Bengio, "Professor forcing: A new algorithm for training recurrent networks," in *Neural Information Processing Systems*, pp. 4601–4609, 2016.

[236] 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*, pp. 2107–2116, 2017.

[237] K. Bousmalis, A. Irpan, P. Wohlhart, Y. Bai, M. Kelcey, M. Kalakrishnan, L. Downs, J. Ibarz, P. Pastor, K. Konolige, *et al.*, "Using simulation and domain adaptation to improve efficiency of deep robotic grasping," in *International Conference on Robotics and Automation*, pp. 4243–4250, 2018.

[238] S. James, P. Wohlhart, M. Kalakrishnan, D. Kalashnikov, A. Irpan, J. Ibarz, S. Levine, R. Hadsell, and K. Bousmalis, "Sim-to-real via sim-to-sim: Data-efficient robotic grasping via randomized-to-canonical adaptation networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 12627–12637, 2019.

[239] L. Pinto, J. Davidson, and A. Gupta, "Supervision via competition: Robot adversaries for learning tasks," in *International Conference on Robotics and Automation*, pp. 1601–1608, 2017.

[240] A. Anoosheh, E. Agustsson, R. Timofte, and L. Van Gool, "Combogan: Unrestrained scalability for image domain translation," in *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 783–790, 2018.

[241] X. Chen, C. Xu, X. Yang, and D. Tao, "Attention-gan for object transfiguration in wild images," in *European Conference on Computer Vision*, pp. 164–180, 2018.

[242] C. Wang, C. Xu, C. Wang, and D. Tao, "Perceptual adversarial networks for image-to-image transformation," *IEEE Transactions on Image Processing*, vol. 27, no. 8, pp. 4066–4079, 2018.

[243] J. Cao, H. Huang, Y. Li, J. Liu, R. He, and Z. Sun, "Biphasic learning of gans for high-resolution image-to-image translation," *arXiv preprint arXiv:1904.06624*, 2019.

[244] M. Amodio and S. Krishnaswamy, "Travelgan: Image-to-image translation by transformation vector learning," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 8983–8992, 2019.

[245] M.-Y. Liu, X. Huang, A. Mallya, T. Karras, T. Aila, J. Lehtinen, and J. Kautz, "Few-shot unsupervised image-to-image translation," *arXiv preprint arXiv:1905.01723*, 2019.

[246] Z. He, W. Zuo, M. Kan, S. Shan, and X. Chen, "Attgan: Facial attribute editing by only changing what you want," *IEEE Transactions on Image Processing*, 2019.

[247] M. Liu, Y. Ding, M. Xia, X. Liu, E. Ding, W. Zuo, and S. Wen, "Stgan: A unified selective transfer network for arbitrary image attribute editing," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3673–3682, 2019.

[248] H. Edwards and A. Storkey, "Censoring representations with an adversary," *arXiv preprint arXiv:1511.05897*, 2015.

[249] A. Beutel, J. Chen, Z. Zhao, and E. H. Chi, "Data decisions and theoretical implications when adversarially learning fair representations," *arXiv preprint arXiv:1707.00075*, 2017.

[250] B. H. Zhang, B. Lemoine, and M. Mitchell, "Mitigating unwanted biases with adversarial learning," in *AAAI/ACM Conference on AI, Ethics, and Society*, pp. 335–340, 2018.

[251] D. Madras, E. Creager, T. Pitassi, and R. Zemel, "Learning adversarially fair and transferable representations," in *International Conference on Machine Learning*, 2018.

[252] L. Hu, M. Kan, S. Shan, and X. Chen, "Duplex generative adversarial network for unsupervised domain adaptation," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1498–1507, 2018.

[253] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2009.

[254] S. Sankaranarayanan, Y. Balaji, A. Jain, S. Nam Lim, and R. Chellappa, "Learning from synthetic data: Addressing domain shift for semantic segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3752–3761, 2018.

[255] Y.-H. Tsai, W.-C. Hung, S. Schulter, K. Sohn, M.-H. Yang, and M. Chandraker, "Learning to adapt structured output space for semantic segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7472–7481, 2018.

[256] J. Shen, Y. Qu, W. Zhang, and Y. Yu, "Wasserstein distance guided representation learning for domain adaptation," *arXiv preprint arXiv:1707.01217*, 2017.

[257] S. Benaim and L. Wolf, "One-sided unsupervised domain mapping," in *Neural Information Processing Systems*, pp. 752–762, 2017.

[258] H. Zhao, S. Zhang, G. Wu, J. M. Moura, J. P. Costeira, and G. J. Gordon, "Adversarial multiple source domain adaptation," in *Neural Information Processing Systems*, pp. 8559–8570, 2018.

[259] M. Long, Z. Cao, J. Wang, and M. I. Jordan, "Conditional adversarial domain adaptation," in *Neural Information Processing Systems*, pp. 1640–1650, 2018.

[260] W. Hong, Z. Wang, M. Yang, and J. Yuan, "Conditional generative adversarial network for structured domain adaptation," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1335–1344, 2018.

[261] K. Saito, K. Watanabe, Y. Ushiku, and T. Harada, "Maximum classifier discrepancy for unsupervised domain adaptation," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3723–3732, 2018.

[262] R. Volpi, P. Morerio, S. Savarese, and V. Murino, "Adversarial feature augmentation for unsupervised domain adaptation," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5495–5504, 2018.

[263] X. Chen, S. Li, H. Li, S. Jiang, Y. Qi, and L. Song, "Generative adversarial user model for reinforcement learning based recommendation system," in *International Conference on Machine Learning*, pp. 1052–1061, 2019.

[264] D. Pfau and O. Vinyals, "Connecting generative adversarial networks and actor-critic methods," *arXiv preprint arXiv:1610.01945*, 2016.

[265] C. Finn, P. Christiano, P. Abbeel, and S. Levine, "A connection between generative adversarial networks, inverse reinforcement learning, and energy-based models," *arXiv preprint arXiv:1611.03852*, 2016.

[266] Y. Ganin, T. Kulkarni, I. Babuschkin, S. Eslami, and O. Vinyals, "Synthesizing programs for images using reinforced adversarial learning," in *International Conference on Machine Learning*, pp. 1666–1675, 2018.

[267] T. Bansal, J. Pachocki, S. Sidor, I. Sutskever, and I. Mordatch, "Emergent complexity via multi-agent competition," *arXiv preprint arXiv:1710.03748*, 2017.

[268] J. Yoo, H. Ha, J. Yi, J. Ryu, C. Kim, J.-W. Ha, Y.-H. Kim, and S. Yoon, "Energy-based sequence gans for recommendation and their connection to imitation learning," *arXiv preprint arXiv:1706.09200*, 2017.

[269] J. Ho and S. Ermon, "Generative adversarial imitation learning," in *Neural Information Processing Systems*, pp. 4565–4573, 2016.

[270] Y. Guo, J. Oh, S. Singh, and H. Lee, "Generative adversarial self-imitation learning," *arXiv preprint arXiv:1812.00950*, 2018.

[271] W. Shang, Y. Yu, Q. Li, Z. Qin, Y. Meng, and J. Ye, "Environment reconstruction with hidden confounders for reinforcement learning based recommendation," in *ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 566–576, 2019.

[272] J.-Y. Zhu, R. Zhang, D. Pathak, T. Darrell, A. A. Efros, O. Wang, and E. Shechtman, "Toward multimodal image-to-image translation," in *Neural Information Processing Systems*, pp. 465–476, 2017.

[273] A. Almahairi, S. Rajeswar, A. Sordoni, P. Bachman, and A. Courville, "Augmented cyclegan: Learning many-to-many mappings from unpaired data," *arXiv preprint arXiv:1802.10151*, 2018.

[274] X. Huang, M.-Y. Liu, S. Belongie, and J. Kautz, "Multimodal unsupervised image-to-image translation," in *European Conference on Computer Vision (ECCV)*, pp. 172–189, 2018.

[275] H.-Y. Lee, H.-Y. Tseng, J.-B. Huang, M. Singh, and M.-H. Yang, "Diverse image-to-image translation via disentangled representations," in *European Conference on Computer Vision*, pp. 35–51, 2018.

[276] W. Lotter, G. Kreiman, and D. Cox, "Unsupervised learning of visual structure using predictive generative networks," *arXiv preprint arXiv:1511.06380*, 2015.

[277] J. Jordon, J. Yoon, and M. van der Schaar, "Knockoffgan: Generating knockoffs for feature selection using generative adversarial networks," in *International Conference on Learning Representations*, 2019.

[278] J. Song, T. He, L. Gao, X. Xu, A. Hanjalic, and H. T. Shen, "Binary generative adversarial networks for image retrieval," in *AAAI Conference on Artificial Intelligence*, pp. 394–401, 2018.

[279] J. Zhang, Y. Peng, and M. Yuan, "Unsupervised generative adversarial cross-modal hashing," in *AAAI Conference on Artificial Intelligence*, pp. 539–546, 2018.

[280] J. Zhang, Y. Peng, and M. Yuan, "Sch-gan: Semi-supervised cross-modal hashing by generative adversarial network," *IEEE Transactions on Cybernetics*, vol. 50, no. 2, pp. 489–502, 2020.

[281] K. Ghasedi Dizaji, F. Zheng, N. Sadoughi, Y. Yang, C. Deng, and H. Huang, "Unsupervised deep generative adversarial hashing network," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3664–3673, 2018.

[282] G. Wang, Q. Hu, J. Cheng, and Z. Hou, "Semi-supervised generative adversarial hashing for image retrieval," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 469–485, 2018.

[283] Z. Qiu, Y. Pan, T. Yao, and T. Mei, "Deep semantic hashing with generative adversarial networks," in *ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 225–234, ACM, 2017.

[284] J. Zhang, Z. Wei, I. C. Duta, F. Shen, L. Liu, F. Zhu, X. Xu, L. Shao, and H. T. Shen, "Generative reconstructive hashing for incomplete video analysis," in *ACM International Conference on Multimedia*, pp. 845–854, ACM, 2019.

[285] Y. Wang, L. Zhang, F. Nie, X. Li, Z. Chen, and F. Wang, "Wegan: Deep image hashing with weighted generative adversarial networks," *IEEE Transactions on Multimedia*, 2019.

[286] G. Dai, J. Xie, and Y. Fang, "Metric-based generative adversarial network," in *ACM International Conference on Multimedia*, pp. 672–680, 2017.

[287] S. C.-X. Li, B. Jiang, and B. Marlin, "Misgan: Learning from incomplete data with generative adversarial networks," in *International Conference on Learning Representations*, 2019.

[288] C. Wang, C. Xu, X. Yao, and D. Tao, "Evolutionary generative adversarial networks," *IEEE Transactions on Evolutionary Computation*, 2019.

[289] C. R. Ponce, W. Xiao, P. F. Schade, T. S. Hartmann, G. Kreiman, and M. S. Livingstone, "Evolving images for visual neurons using a deep generative network reveals coding principles and neuronal preferences," *Cell*, vol. 177, no. 4, pp. 999–1009, 2019.

[290] Y. Wang, Y. Xia, T. He, F. Tian, T. Qin, C. Zhai, and T.-Y. Liu, "Multi-agent dual learning," in *International Conference on Learning Representations*, 2019.

[291] J.-J. Zhu and J. Bento, "Generative adversarial active learning," *arXiv preprint arXiv:1702.07956*, 2017.

[292] M.-K. Xie and S.-J. Huang, "Learning class-conditional gans with active sampling," in *ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 998–1006, ACM, 2019.

[293] P. Grnarova, K. Y. Levy, A. Lucchi, T. Hofmann, and A. Krause, "An online learning approach to generative adversarial networks," *arXiv preprint arXiv:1706.03269*, 2017.

[294] I. O. Tolstikhin, S. Gelly, O. Bousquet, C.-J. Simon-Gabriel, and B. Schölkopf, "Adagan: Boosting generative models," in *Neural Information Processing Systems*, pp. 5424–5433, 2017.

[295] Y. Zhu, M. Elhoseiny, B. Liu, X. Peng, and A. Elgammal, "A generative adversarial approach for zero-shot learning from noisy texts," in *IEEE conference on computer vision and pattern recognition*, pp. 1004–1013, 2018.

[296] P. Qin, X. Wang, W. Chen, C. Zhang, W. Xu, and W. Y. Wang, "Generative adversarial zero-shot relational learning for knowledge graphs," in *AAAI Conference on Artificial Intelligence*, 2020.

[297] P. Yang, Q. Tan, H. Tong, and J. He, "Task-adversarial cogenerative nets," in *ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 1596–1604, ACM, 2019.

[298] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT press, 2016.

[299] A. Srivastava, L. Valkov, C. Russell, M. U. Gutmann, and C. Sutton, "Veegan: Reducing mode collapse in gans using implicit variational learning," in *Neural Information Processing Systems*, pp. 3308–3318, 2017.

[300] D. Bau, J.-Y. Zhu, J. Wulff, W. Peebles, H. Strobelt, B. Zhou, and A. Torralba, "Seeing what a gan cannot generate," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 4502–4511, 2019.

[301] S. Arora, R. Ge, Y. Liang, T. Ma, and Y. Zhang, "Generalization and equilibrium in generative adversarial nets (gans)," in *International Conference on Machine Learning*, pp. 224–232, 2017.

[302] K. Roth, A. Lucchi, S. Nowozin, and T. Hofmann, "Stabilizing training of generative adversarial networks through regularization," in *Neural Information Processing Systems*, pp. 2018–2028, 2017.

[303] L. Mescheder, S. Nowozin, and A. Geiger, "The numerics of gans," in *Neural Information Processing Systems*, pp. 1825–1835, 2017.

[304] L. Mescheder, A. Geiger, and S. Nowozin, "Which training methods for gans do actually converge?," in *International Conference on Machine Learning*, pp. 3481–3490, 2018.

[305] Z. Lin, A. Khetan, G. Fanti, and S. Oh, "Pacgan: The power of two samples in generative adversarial networks," in *Advances in Neural Information Processing Systems*, pp. 1498–1507, 2018.

[306] S. Arora, A. Risteski, and Y. Zhang, "Do gans learn the distribution? some theory and empirics," in *International Conference on Learning Representations*, 2018.

[307] Y. Bai, T. Ma, and A. Risteski, "Approximability of discriminators implies diversity in gans," *arXiv preprint arXiv:1806.10586*, 2018.

[308] T. Liang, "On how well generative adversarial networks learn densities: Nonparametric and parametric results," *arXiv preprint arXiv:1811.03179*, 2018.

[309] S. Singh, A. Uppal, B. Li, C.-L. Li, M. Zaheer, and B. Póczos, "Nonparametric density estimation with adversarial losses," in *Neural Information Processing Systems*, pp. 10246–10257, Curran Associates Inc., 2018.

[310] S. Arora, A. Risteski, and Y. Zhang, "Theoretical limitations of encoder-decoder gan architectures," *arXiv preprint arXiv:1711.02651*, 2017.

[311] A. Creswell and A. A. Bharath, "Inverting the generator of a generative adversarial network," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 7, pp. 1967–1974, 2019.

[312] S. Mohamed and B. Lakshminarayanan, "Learning in implicit generative models," *arXiv preprint arXiv:1610.03483*, 2016.

[313] G. Gidel, H. Berard, G. Vignoud, P. Vincent, and S. Lacoste-Julien, "A variational inequality perspective on generative adversarial networks," in *International Conference on Learning Representations*, 2019.

[314] M. Sanjabi, J. Ba, M. Razaviyayn, and J. D. Lee, "On the convergence and robustness of training gans with regularized optimal transport," in *Neural Information Processing Systems*, pp. 7091–7101, 2018.

[315] V. Nagarajan, C. Raffel, and I. J. Goodfellow, "Theoretical insights into memorization in gans," in *Neural Information Processing Systems Workshop*, 2018.

[316] Y. Blau, R. Mechrez, R. Timofte, T. Michaeli, and L. Zelnik-Manor, "The 2018 pirm challenge on perceptual image super-resolution," in *European Conference on Computer Vision Workshops*, pp. 334–355, 2018.

[317] X. Yu and F. Porikli, "Ultra-resolving face images by discriminative generative networks," in *European conference on computer vision*, pp. 318–333, Springer, 2016.

[318] H. Zhu, A. Zheng, H. Huang, and R. He, "High-resolution talking face generation via mutual information approximation," *arXiv preprint arXiv:1812.06589*, 2018.

[319] H. Huang, R. He, Z. Sun, and T. Tan, "Wavelet domain generative adversarial network for multi-scale face hallucination," *International Journal of Computer Vision*, vol. 127, no. 6-7, pp. 763–784, 2019.

[320] C. K. Sønderby, J. Caballero, L. Theis, W. Shi, and F. Huszár, "Amortised map inference for image super-resolution," in *International Conference on Learning Representations*, 2017.

[321] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *European Conference on Computer Vision*, pp. 694–711, Springer, 2016.

[322] X. Wang, K. Yu, C. Dong, and C. Change Loy, "Recovering realistic texture in image super-resolution by deep spatial feature transform," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 606–615, 2018.

[323] W. Zhang, Y. Liu, C. Dong, and Y. Qiao, "Ranksrgan: Generative adversarial networks with ranker for image super-resolution," in *International Conference on Computer Vision*, 2019.

[324] L. Tran, X. Yin, and X. Liu, "Disentangled representation learning gan for pose-invariant face recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1415–1424, 2017.

[325] J. Cao, Y. Hu, H. Zhang, R. He, and Z. Sun, "Learning a high fidelity pose invariant model for high-resolution face frontalization," in *Advances in Neural Information Processing Systems*, pp. 2867–2877, 2018.

[326] A. Siarohin, E. Sangineto, S. Lathuilière, and N. Sebe, "Deformable gans for pose-based human image generation," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3408–3416, 2018.

[327] C. Wang, C. Wang, C. Xu, and D. Tao, "Tag disentangled generative adversarial networks for object image re-rendering," in *International Joint Conference on Aritifical Intelligence*, 2017.

[328] Z. Shu, E. Yumer, S. Hadap, K. Sunkavalli, E. Shechtman, and D. Samaras, "Neural face editing with intrinsic image disentangling," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5541–5550, 2017.

[329] H. Chang, J. Lu, F. Yu, and A. Finkelstein, "Pairedcyclegan: Asymmetric style transfer for applying and removing makeup," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 40–48, 2018.

[330] B. Dolhansky and C. Canton Ferrer, "Eye in-painting with exemplar generative adversarial networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7902–7911, 2018.

[331] A. Pumarola, A. Agudo, A. M. Martinez, A. Sanfeliu, and F. Moreno-Noguer, "Ganimation: Anatomically-aware facial animation from a single image," in *European Conference on Computer Vision*, pp. 818–833, 2018.

[332] W. Yin, Y. Fu, L. Sigal, and X. Xue, "Semi-latent gan: Learning to generate and modify facial images from attributes," *arXiv preprint arXiv:1704.02166*, 2017.

[333] C. Donahue, Z. C. Lipton, A. Balsubramani, and J. McAuley, "Semantically decomposing the latent spaces of generative adversarial networks," in *International Conference on Learning Representations*, 2018.

[334] A. Duarte, F. Roldan, M. Tubau, J. Escur, S. Pascual, A. Salvador, E. Mohedano, K. McGuinness, J. Torres, and X. Giro-i Nieto, "Wav2pix: speech-conditioned face generation using generative adversarial networks," in *International Conference on Acoustics, Speech and Signal Processing*, vol. 3, 2019.

[335] B. Gecer, S. Ploumpis, I. Kotsia, and S. Zafeiriou, "Ganfit: Generative adversarial network fitting for high fidelity 3d face reconstruction," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1155–1164, 2019.

[336] Z. Shu, M. Sahasrabudhe, R. Alp Guler, D. Samaras, N. Paragios, and I. Kokkinos, "Deforming autoencoders: Unsupervised disentangling of shape and appearance," in *European Conference on Computer Vision*, pp. 650–665, 2018.

[337] C. Fu, X. Wu, Y. Hu, H. Huang, and R. He, "Dual variational generation for low-shot heterogeneous face recognition," in *Neural Information Processing Systems*, 2019.

[338] Y. Lu, Y.-W. Tai, and C.-K. Tang, "Attribute-guided face generation using conditional cyclegan," in *European Conference on Computer Vision*, pp. 282–297, 2018.

[339] J. Cao, Y. Hu, B. Yu, R. He, and Z. Sun, "3d aided duet gans for multi-view face image synthesis," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 8, pp. 2028–2042, 2019.

[340] Y. Liu, Q. Li, and Z. Sun, "Attribute-aware face aging with wavelet-based generative adversarial networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 11877–11886, 2019.

[341] J. Bao, D. Chen, F. Wen, H. Li, and G. Hua, "Cvae-gan: fine-grained image generation through asymmetric training," in *IEEE International Conference on Computer Vision*, pp. 2745–2754, 2017.

[342] H. Dong, S. Yu, C. Wu, and Y. Guo, "Semantic image synthesis via adversarial learning," in *IEEE International Conference on Computer Vision*, pp. 5706–5714, 2017.

[343] J. Wu, C. Zhang, T. Xue, B. Freeman, and J. Tenenbaum, "Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling," in *Neural Information Processing Systems*, pp. 82–90, 2016.

[344] D. J. Im, C. D. Kim, H. Jiang, and R. Memisevic, "Generating images with recurrent adversarial networks," *arXiv preprint arXiv:1602.05110*, 2016.

[345] J. Yang, A. Kannan, D. Batra, and D. Parikh, "Lr-gan: Layered recursive generative adversarial networks for image generation," *arXiv preprint arXiv:1703.01560*, 2017.

[346] X. Wang, A. Shrivastava, and A. Gupta, "A-fast-rcnn: Hard positive generation via adversary for object detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2606–2615, 2017.

[347] R. Villegas, J. Yang, S. Hong, X. Lin, and H. Lee, "Decomposing motion and content for natural video sequence prediction," *arXiv preprint arXiv:1706.08033*, 2017.

[348] E. Santana and G. Hotz, "Learning a driving simulator," *arXiv preprint arXiv:1608.01230*, 2016.

[349] C. Chan, S. Ginosar, T. Zhou, and A. A. Efros, "Everybody dance now," *arXiv preprint arXiv:1808.07371*, 2018.

[350] A. Clark, J. Donahue, and K. Simonyan, "Efficient video generation on complex datasets," *arXiv preprint arXiv:1907.06571*, 2019.

[351] M. Mathieu, C. Couprie, and Y. LeCun, "Deep multi-scale video prediction beyond mean square error," *arXiv preprint arXiv:1511.05440*, 2015.

[352] X. Liang, L. Lee, W. Dai, and E. P. Xing, "Dual motion gan for future-flow embedded video prediction," in *IEEE International Conference on Computer Vision*, pp. 1744–1752, 2017.

[353] A. Bansal, S. Ma, D. Ramanan, and Y. Sheikh, "Recycle-gan: Unsupervised video retargeting," in *European Conference on Computer Vision*, pp. 119–135, 2018.

[354] X. Liang, H. Zhang, L. Lin, and E. Xing, "Generative semantic manipulation with mask-contrasting gan," in *European Conference on Computer Vision*, pp. 558–573, 2018.

[355] T. Kim, B. Kim, M. Cha, and J. Kim, "Unsupervised visual attribute transfer with reconfigurable generative adversarial networks," *arXiv preprint arXiv:1707.09798*, 2017.

[356] Y. Chen, Y.-K. Lai, and Y.-J. Liu, "Cartoongan: Generative adversarial networks for photo cartoonization," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 9465–9474, 2018.

[357] R. Villegas, J. Yang, D. Ceylan, and H. Lee, "Neural kinematic networks for unsupervised motion retargetting," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 8639–8648, 2018.

[358] S. Zhou, T. Xiao, Y. Yang, D. Feng, Q. He, and W. He, "Genegan: Learning object transfiguration and attribute subspace from unpaired data," *arXiv preprint arXiv:1705.04932*, 2017.

[359] H. Wu, S. Zheng, J. Zhang, and K. Huang, "Gp-gan: Towards realistic high-resolution image blending," 2019.

[360] N. Souly, C. Spampinato, and M. Shah, "Semi supervised semantic segmentation using generative adversarial network," in *IEEE International Conference on Computer Vision*, pp. 5688–5696, 2017.

[361] J. Pan, C. C. Ferrer, K. McGuinness, N. E. O'Connor, J. Torres, E. Sayrol, and X. Giro-i Nieto, "Salgan: Visual saliency prediction with generative adversarial networks," *arXiv preprint arXiv:1701.01081*, 2017.

[362] Y. Song, C. Ma, X. Wu, L. Gong, L. Bao, W. Zuo, C. Shen, R. W. Lau, and M.-H. Yang, "Vital: Visual tracking via adversarial learning," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 8990–8999, 2018.

[363] Y. Han, P. Zhang, W. Huang, Y. Zha, G. D. Cooper, and Y. Zhang, "Robust visual tracking using unlabeled adversarial instance generation and regularized label smoothing," *Pattern Recognition*, pp. 1–15, 2019.

[364] D. Engin, A. Genç, and H. Kemal Ekenel, "Cycle-dehaze: Enhanced cyclegan for single image dehazing," in *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 825–833, 2018.

[365] X. Yang, Z. Xu, and J. Luo, "Towards perceptual image dehazing by physics-based disentanglement and adversarial training," in *AAAI conference on artificial intelligence*, pp. 7485–7492, 2018.

[366] W. Liu, X. Hou, J. Duan, and G. Qiu, "End-to-end single image fog removal using enhanced cycle consistent adversarial networks," *arXiv preprint arXiv:1902.01374*, 2019.

[367] S. Lutz, K. Amplianitis, and A. Smolic, "Alphagan: Generative adversarial networks for natural image matting," *arXiv preprint arXiv:1807.10088*, 2018.

[368] 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*, pp. 5485–5493, 2017.

[369] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang, "Generative image inpainting with contextual attention," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5505–5514, 2018.

[370] X. Liu, Y. Wang, and Q. Liu, "Psgan: a generative adversarial network for remote sensing image pan-sharpening," in *IEEE International Conference on Image Processing*, pp. 873–877, IEEE, 2018.

[371] S. Iizuka, E. Simo-Serra, and H. Ishikawa, "Globally and locally consistent image completion," *ACM Transactions on Graphics*, vol. 36, no. 4, p. 107, 2017.

[372] F. Liu, L. Jiao, and X. Tang, "Task-oriented gan for polsar image classification and clustering," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 9, pp. 2707–2719, 2019.

[373] A. Creswell and A. A. Bharath, "Adversarial training for sketch retrieval," in *European Conference on Computer Vision*, pp. 798–809, 2016.

[374] M. Zhang, K. Teck Ma, J. Hwee Lim, Q. Zhao, and J. Feng, "Deep future gaze: Gaze anticipation on egocentric videos using adversarial networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4372–4381, 2017.

[375] M. Zhang, K. T. Ma, J. Lim, Q. Zhao, and J. Feng, "Anticipating where people will look using adversarial networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 8, pp. 1783–1796, 2019.

[376] F. Fang, J. Yamagishi, I. Echizen, and J. Lorenzo-Trueba, "High-quality nonparallel voice conversion based on cycle-consistent adversarial network," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 5279–5283, 2018.

[377] T. Kaneko and H. Kameoka, "Parallel-data-free voice conversion using cycle-consistent adversarial networks," *arXiv preprint arXiv:1711.11293*, 2017.

[378] C. Esteban, S. L. Hyland, and G. Rätsch, "Real-valued (medical) time series generation with recurrent conditional gans," *arXiv preprint arXiv:1706.02633*, 2017.

[379] K. G. Hartmann, R. T. Schirrmeister, and T. Ball, "Eeg-gan: Generative adversarial networks for electroencephalograhic (eeg) brain signals," *arXiv preprint arXiv:1806.01875*, 2018.

[380] C. Donahue, J. McAuley, and M. Puckette, "Synthesizing audio with generative adversarial networks," *arXiv preprint arXiv:1802.04208*, vol. 1, 2018.

[381] D. Li, D. Chen, B. Jin, L. Shi, J. Goh, and S.-K. Ng, "Madgan: Multivariate anomaly detection for time series data with generative adversarial networks," in *International Conference on Artificial Neural Networks*, pp. 703–716, Springer, 2019.

[382] J. Li, W. Monroe, T. Shi, S. Jean, A. Ritter, and D. Jurafsky, "Adversarial learning for neural dialogue generation," *arXiv preprint arXiv:1701.06547*, 2017.

[383] Y. Zhang, Z. Gan, and L. Carin, "Generating text via adversarial training," in *NIPS workshop on Adversarial Training*, vol. 21, 2016.

[384] W. Fedus, I. Goodfellow, and A. M. Dai, "Maskgan: better text generation via filling in the _," *arXiv preprint arXiv:1801.07736*, 2018.

[385] S. Yang, J. Liu, W. Wang, and Z. Guo, "Tet-gan: Text effects transfer via stylization and destylization," in *AAAI Conference on Artificial Intelligence*, pp. 1238–1245, 2019.

[386] L. Cai and W. Y. Wang, "Kbgan: Adversarial learning for knowledge graph embeddings," *arXiv preprint arXiv:1711.04071*, 2017.

[387] X. Wang, W. Chen, Y.-F. Wang, and W. Y. Wang, "No metrics are perfect: Adversarial reward learning for visual storytelling," *arXiv preprint arXiv:1804.09160*, 2018.

[388] P. Qin, W. Xu, and W. Y. Wang, "Dsgan: generative adversarial training for distant supervision relation extraction," *arXiv preprint arXiv:1805.09929*, 2018.

[389] C. d. M. d'Autume, M. Rosca, J. Rae, and S. Mohamed, "Training language gans from scratch," *arXiv preprint arXiv:1905.09922*, 2019.

[390] A. Dash, J. C. B. Gamboa, S. Ahmed, M. Liwicki, and M. Z. Afzal, "Tac-gan-text conditioned auxiliary classifier generative adversarial network," *arXiv preprint arXiv:1703.06412*, 2017.

[391] T.-H. Chen, Y.-H. Liao, C.-Y. Chuang, W.-T. Hsu, J. Fu, and M. Sun, "Show, adapt and tell: Adversarial training of cross-domain image captioner," in *IEEE International Conference on Computer Vision*, pp. 521–530, 2017.

[392] R. Shetty, M. Rohrbach, L. Anne Hendricks, M. Fritz, and B. Schiele, "Speaking the same language: Matching machine to human captions by adversarial training," in *IEEE International Conference on Computer Vision*, pp. 4135–4144, 2017.

[393] S. Rao and H. Daumé III, "Answer-based adversarial training for generating clarification questions," *arXiv preprint arXiv:1904.02281*, 2019.

[394] X. Yang, M. Khabsa, M. Wang, W. Wang, A. Awadallah, D. Kifer, and C. L. Giles, "Adversarial training for community question answer selection based on multi-scale matching," 2019.

[395] B. Liu, J. Fu, M. P. Kato, and M. Yoshikawa, "Beyond narrative description: Generating poetry from images by multi-adversarial training," in *ACM Multimedia Conference on Multimedia Conference*, pp. 783–791, ACM, 2018.

[396] Y. Luo, H. Zhang, Y. Wen, and X. Zhang, "Resumegan: An optimized deep representation learning framework for talent-job fit via adversarial learning," in *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pp. 1101–1110, ACM, 2019.

[397] C. Garbacea, S. Carton, S. Yan, and Q. Mei, "Judge the judges: A large-scale evaluation study of neural language models for online review generation," in *Conference on Empirical Methods in Natural Language Processing & International Joint Conference on Natural Language Processing*, 2019.

[398] H. Aghakhani, A. Machiry, S. Nilizadeh, C. Kruegel, and G. Vigna, "Detecting deceptive reviews using generative adversarial networks," in *IEEE Security and Privacy Workshops*, pp. 89–95, 2018.

[399] K. Takuhiro, K. Hirokazu, H. Nobukatsu, I. Yusuke, H. Kaoru, and K. Kunio, "Generative adversarial network-based postfiltering for statistical parametric speech synthesis," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 4910–4914, 2017.

[400] Y. Saito, S. Takamichi, and H. Saruwatari, "Statistical parametric speech synthesis incorporating generative adversarial networks," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 1, pp. 84–96, 2018.

[401] C. Donahue, J. McAuley, and M. Puckette, "Adversarial audio synthesis," *arXiv preprint arXiv:1802.04208*, 2018.

[402] S. Pascual, A. Bonafonte, and J. Serra, "Segan: Speech enhancement generative adversarial network," in *Interspeech*, pp. 3642–3646, 2017.

[403] C. Donahue, B. Li, and R. Prabhavalkar, "Exploring speech enhancement with generative adversarial networks for robust speech recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 5024–5028, 2018.

[404] N. Killoran, L. J. Lee, A. Delong, D. Duvenaud, and B. J. Frey, "Generating and designing dna with deep generative models," *arXiv preprint arXiv:1712.06148*, 2017.

[405] A. Gupta and J. Zou, "Feedback gan (fbgan) for dna: A novel feedback-loop architecture for optimizing protein functions," *arXiv preprint arXiv:1804.01694*, 2018.

[406] M. Benhenda, "Chemgan challenge for drug discovery: can ai reproduce natural chemical diversity?," *arXiv preprint arXiv:1708.08227*, 2017.

[407] E. Choi, S. Biswal, B. Malin, J. Duke, W. F. Stewart, and J. Sun, "Generating multi-label discrete patient records using generative adversarial networks," *arXiv preprint arXiv:1703.06490*, 2017.

[408] W. Dai, J. Doyle, X. Liang, H. Zhang, N. Dong, Y. Li, and E. P. Xing, "Scan: Structure correcting adversarial network for chest x-rays organ segmentation," *arXiv preprint arXiv:1703.08770*, vol. 1, 2017.

[409] T. Schlegl, P. Seeböck, S. M. Waldstein, U. Schmidt-Erfurth, and G. Langs, "Unsupervised anomaly detection with generative adversarial networks to guide marker discovery," in *International Conference on Information Processing in Medical Imaging*, pp. 146–157, Springer, 2017.

[410] J. M. Wolterink, A. M. Dinkla, M. H. Savenije, P. R. Seevinck, C. A. van den Berg, and I. Išgum, "Deep mr to ct synthesis using unpaired data," in *International Workshop on Simulation and Synthesis in Medical Imaging*, pp. 14–23, 2017.

[411] T. M. Quan, T. Nguyen-Duc, and W.-K. Jeong, "Compressed sensing mri reconstruction using a generative adversarial network with a cyclic loss," *IEEE Transactions on Medical Imaging*, vol. 37, no. 6, pp. 1488–1497, 2018.

[412] M. Mardani, E. Gong, J. Y. Cheng, S. S. Vasanawala, G. Zaharchuk, L. Xing, and J. M. Pauly, "Deep generative adversarial neural networks for compressive sensing mri," *IEEE Transactions on Medical Imaging*, vol. 38, no. 1, pp. 167–179, 2018.

[413] Y. Xue, T. Xu, H. Zhang, L. R. Long, and X. Huang, "Segan: Adversarial network with multi-scale l_1 loss for medical image segmentation," *Neuroinformatics*, vol. 16, no. 3-4, pp. 383–392, 2018.

[414] Q. Yang, P. Yan, Y. Zhang, H. Yu, Y. Shi, X. Mou, M. K. Kalra, Y. Zhang, L. Sun, and G. Wang, "Low-dose ct image denoising using a generative adversarial network with wasserstein distance and perceptual loss," *IEEE Transactions on Medical Imaging*, vol. 37, no. 6, pp. 1348–1357, 2018.

[415] G. St-Yves and T. Naselaris, "Generative adversarial networks conditioned on brain activity reconstruct seen images," in *IEEE International Conference on Systems, Man, and Cybernetics*, pp. 1054–1061, 2018.

[416] J.-J. Hwang, S. Azernikov, A. A. Efros, and S. X. Yu, "Learning beyond human expertise with generative models for dental restorations," *arXiv preprint arXiv:1804.00064*, 2018.

[417] B. Tian, Y. Zhang, X. Chen, C. Xing, and C. Li, "Drgan: A gan-based framework for doctor recommendation in chinese on-line qa communities," in *International Conference on Database Systems for Advanced Applications*, pp. 444–447, 2019.

[418] Z. Zheng, L. Zheng, and Y. Yang, "Unlabeled samples generated by gan improve the person re-identification baseline in vitro," in *IEEE International Conference on Computer Vision*, pp. 3754–3762, 2017.

[419] M. Gorijala and A. Dukkipati, "Image generation and editing with variational info generative adversarial networks," *arXiv preprint arXiv:1701.04568*, 2017.

[420] X. Wang, Z. Man, M. You, and C. Shen, "Adversarial generation of training examples: applications to moving vehicle license plate recognition," *arXiv preprint arXiv:1707.03124*, 2017.

[421] B. Chang, Q. Zhang, S. Pan, and L. Meng, "Generating handwritten chinese characters using cyclegan," in *IEEE Winter Conference on Applications of Computer Vision*, pp. 199–207, 2018.

[422] L. Sixt, B. Wild, and T. Landgraf, "Rendergan: Generating realistic labeled data," *Frontiers in Robotics and AI*, vol. 5, p. 66, 2018.

[423] D. Xu, S. Yuan, L. Zhang, and X. Wu, "Fairgan: Fairness-aware generative adversarial networks," in *IEEE International Conference on Big Data*, pp. 570–575, 2018.

[424] M.-C. Lee, B. Gao, and R. Zhang, "Rare query expansion through generative adversarial networks in search advertising," in *ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 500–508, ACM, 2018.

[425] M. O. Turkoglu, W. Thong, L. Spreeuwers, and B. Kicanaoglu, "A layer-based sequential framework for scene generation with gans," *arXiv preprint arXiv:1902.00671*, 2019.

[426] A. El-Nouby, S. Sharma, H. Schulz, D. Hjelm, L. El Asri, S. E. Kahou, Y. Bengio, and G. W. Taylor, "Tell, draw, and repeat: Generating and modifying images based on continual linguistic instruction," in *International Conference on Computer Vision*, 2019.

[427] N. Ratzlaff and L. Fuxin, "Hypergan: A generative model for diverse, performant neural networks," in *International Conference on Machine Learning*, pp. 5361–5369, 2019.

[428] M. Frid-Adar, I. Diamant, E. Klang, M. Amitai, J. Goldberger, and H. Greenspan, "Gan-based synthetic medical image augmentation for increased cnn performance in liver lesion classification," *Neurocomputing*, vol. 321, pp. 321–331, 2018.

[429] Q. Wang, H. Yin, H. Wang, Q. V. H. Nguyen, Z. Huang, and L. Cui, "Enhancing collaborative filtering with generative augmentation," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 548–556, 2019.

[430] Y. Zhang, Y. Fu, P. Wang, X. Li, and Y. Zheng, "Unifying inter-region autocorrelation and intra-region structures for spatial embedding via collective adversarial learning," in *ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 1700–1708, 2019.

[431] H. Gao, J. Pei, and H. Huang, "Progan: Network embedding via proximity generative adversarial network," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 1308–1316, 2019.

[432] B. Hu, Y. Fang, and C. Shi, "Adversarial learning on heterogeneous information networks," in *ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 120–129, 2019.

[433] P. Wang, Y. Fu, H. Xiong, and X. Li, "Adversarial substructured representation learning for mobile user profiling," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 130–138, 2019.

[434] W. Hu and Y. Tan, "Generating adversarial malware examples for black-box attacks based on gan," *arXiv preprint arXiv:1702.05983*, 2017.

[435] M. Chidambaram and Y. Qi, "Style transfer generative adversarial networks: Learning to play chess differently," *arXiv preprint arXiv:1702.06762*, 2017.

[436] C. Chu, A. Zhmoginov, and M. Sandler, "Cyclegan, a master of steganography," *arXiv preprint arXiv:1712.02950*, 2017.

[437] D. Volkhonskiy, I. Nazarov, B. Borisenko, and E. Burnaev, "Steganographic generative adversarial networks," *arXiv preprint arXiv:1703.05502*, 2017.

[438] H. Shi, J. Dong, W. Wang, Y. Qian, and X. Zhang, "Ssgan: secure steganography based on generative adversarial networks," in *Pacific Rim Conference on Multimedia*, pp. 534–544, Springer, 2017.

[439] J. Hayes and G. Danezis, "Generating steganographic images via adversarial training," in *Neural Information Processing Systems*, pp. 1954–1963, 2017.

[440] M. Abadi and D. G. Andersen, "Learning to protect communications with adversarial neural cryptography," *arXiv preprint arXiv:1610.06918*, 2016.

[441] A. N. Gomez, S. Huang, I. Zhang, B. M. Li, M. Osama, and L. Kaiser, "Unsupervised cipher cracking using discrete gans," *arXiv preprint arXiv:1801.04883*, 2018.

[442] B. K. Beaulieu-Jones, Z. S. Wu, C. Williams, R. Lee, S. P. Bhavnani, J. B. Byrd, and C. S. Greene, "Privacy-preserving generative deep neural networks support clinical data sharing," *BioRxiv*, p. 159756, 2018.
[443] A. Gupta, J. Johnson, L. Fei-Fei, S. Savarese, and A. Alahi, "Social gan: Socially acceptable trajectories with generative adversarial networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2255–2264, 2018.
[444] H. Shu, Y. Wang, X. Jia, K. Han, H. Chen, C. Xu, Q. Tian, and C. Xu, "Co-evolutionary compression for unpaired image translation," in *International Conference on Computer Vision*, 2019.
[445] S. Lin, R. Ji, C. Yan, B. Zhang, L. Cao, Q. Ye, F. Huang, and D. Doermann, "Towards optimal structured cnn pruning via generative adversarial learning," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2790–2799, 2019.
[446] C. Daskalakis, A. Ilyas, V. Syrgkanis, and H. Zeng, "Training gans with optimism," 2018.
[447] A. Bora, E. Price, and A. G. Dimakis, "Ambientgan: Generative models from lossy measurements," in *International Conference on Learning Representations*, 2018.
[448] M. J. Kusner and J. M. Hernández-Lobato, "Gans for sequences of discrete elements with the gumbel-softmax distribution," *arXiv preprint arXiv:1611.04051*, 2016.
[449] E. Jang, S. Gu, and B. Poole, "Categorical reparameterization with gumbel-softmax," *arXiv preprint arXiv:1611.01144*, 2016.
[450] C. J. Maddison, A. Mnih, and Y. W. Teh, "The concrete distribution: A continuous relaxation of discrete random variables," *arXiv preprint arXiv:1611.00712*, 2016.
[451] R. J. Williams, "Simple statistical gradient-following algorithms for connectionist reinforcement learning," *Machine learning*, vol. 8, no. 3-4, pp. 229–256, 1992.
[452] R. D. Hjelm, A. P. Jacob, T. Che, A. Trischler, K. Cho, and Y. Bengio, "Boundary-seeking generative adversarial networks," in *International Conference on Learning Representations*, 2018.
[453] T. Che, Y. Li, R. Zhang, R. D. Hjelm, W. Li, Y. Song, and Y. Bengio, "Maximum-likelihood augmented discrete generative adversarial networks," *arXiv preprint arXiv:1702.07983*, 2017.
[454] Z. Junbo, Y. Kim, K. Zhang, A. Rush, and Y. LeCun, "Adversarially regularized autoencoders for generating discrete structures," *arXiv preprint arXiv:1706.04223*, 2017.
[455] Y. Mroueh and T. Sercu, "Fisher gan," in *Neural Information Processing Systems*, pp. 2513–2523, 2017.
[456] J. Yoo, Y. Hong, Y. Noh, and S. Yoon, "Domain adaptation using adversarial learning for autonomous navigation," *arXiv preprint arXiv:1712.03742*, 2017.
[457] Y. Mroueh, T. Sercu, and V. Goel, "Mcgan: Mean and covariance feature matching gan," *arXiv preprint arXiv:1702.08398*, 2017.
[458] Y. Mroueh, C.-L. Li, T. Sercu, A. Raj, and Y. Cheng, "Sobolev gan," *arXiv preprint arXiv:1711.04894*, 2017.
[459] Y. Saatci and A. G. Wilson, "Bayesian gan," in *Neural Information Processing Systems*, pp. 3622–3631, 2017.
[460] P. Zhang, Q. Liu, D. Zhou, T. Xu, and X. He, "On the discrimination-generalization tradeoff in gans," *arXiv preprint arXiv:1711.02771*, 2017.