Generative Adversarial Networks: A Survey

Anlan Zhang

Beihang University, zal1506@buaa.edu.cn

Abstract. Generative Adversarial Networks (GANs) offer an attractive alternative method to maximum likelihood techniques and this framework has lots of applications. In some cases when the explicit density is intractable, GANs sidestep the challenging of characterizing it but can still generate high-quality images. Yet, GANs have some issues including unstable training process as well as mode collaspe and difficulities in convergence in certain situations. Many research works have been conducted on GANs and some pratical tricks for training stalization have been put forward. This paper aims to present a comprehensive survey about the recent advances towards GANs which are related to computer vision. Specifically, it will present an overview of GANs, provide various architectures of GANs, and highlight wide range of applications of GANs. It will also summarize theorical investigations made to understand the training dynamics of GANs as well as effective techniques for stabilization of training process.

1 Introduction

Generative Adversarial Networks (GANs), which correspond to a minimax two-player game, are an emerging model for both semi-supervised and unsupervised learning [7]. Since the original Generative Adversarial Nets [13] was proposed by Ian J. Goodfellow et al with its conditional version [25] by Mehdi Mirza et al in 2014, it has become a promising alternative to maximum likelihood. It sidesteps the characterization of an explicit density yet can generate high-quality images at the same time. However, the training of GANs has been acknowledged to be delicate and unstable, which often leads to the meaningless outputs of the generators or mode collaspe. The algorithms of GANs can fail to converge in some cases. Theoretical investigations have been done to understand the training dynamics of GANs. Some practical tricks have been put forward to stabilize the training process of GANs while a variety of new architectures are coming out one after another. And the various architectures of GANs have been applied in various kinds of applications, including image synthesis [8,15,32,35,41], style transfer [16,18,22,39,43], image super-resolution [21], etc.

This article aims to provide an overview of GANs related to computer vision, the various architectures that have been developed, the theoretical analysis on GANs as well as the techniques which have been explored to efficiently stabilize and optimize the training process and besides, several applications of GANs.

The rest of this article is organized as follows:

– **Section 2** offers background on the motivation of proposing Generative Adversarial Networks and its applications.
– **Section 3** provides an overview of the basic components of GANs along with its conditional version and several popular GAN models currently in use.
– **Section 4** discusses the issues of GANs and the theorical researches on it.
– **Section 5** offers useful tricks that have been advanced to stabilize and optimize the training of GANs.

2 Background on Generative Adversarial Networks

In this section, I discribe the motivations that result in the appearance of Generative Adversarial Networks and its conditional version. I will also present some current domains to which GANs are being applied.

2.1 Motivations of GANs

Deep generative models have achieved less striking sucesses than discriminative models do, because of not only the difficulty of approximating many intractable probabilistic computations that arise in maximum likelihood estimation and related strategies, but also difficulty of leveraging the benefits of piecewise linear units in the generative context [13]. And the basic Generative Adversarial Nets were put forward to avoid these difficulties. Unlike other techniques which also do not need an explicit probability distribution, such as the generative stochastic network (GSN) [3], GANs do not involve any Markov chains or unrolled approximate inference networks for training or sampling.

2.2 Applications of GANs

GANs have achieved great success in many applications related to computer vision ranging from image synthesis to style transfer. In this section, I will provide examples of computer vision areas where GANs are currently making an impact and highlight other emerging areas where GANs hope to make an impact in the future.

– **Image Synthesis** Enhancing the quality and utility of the image generation capabilities has always been a focus of GANs research. The LAPGAN introduces a cascade of convolutional networks within a Laplacian pyramid framwork to generate images in a coarse-to-fine fashion [8]. The key idea of LAPGAN is to divide the generation into successive refinements. Experimental results show that this model do work and can generate higher quality images compared to alternate approaches.
  Ashish Shrivastava et al [35] develop a method for Simulated+Unsupervised learning to learn a model to improve the realism of a simulator's output using unlabeled real data. This model makes several key modifications to the

standard GAN algorithm, including the combination of a self-regularization loss and a local adversarial loss as well as updating the discriminator using a history of refined images, to preserve annotations, avoid artifacts and stabilize training. Their method has bee demonstrated state-of-the-art results without any labeled real data.

The capability to control the content of generated images is one of the desirable properties when employing GANs to image synthesis. Specifically, automatic synthesis of realistic images from text remains a useful and challenging goal. While there have been conditional GANs that condition on class labels, Scott Reed et al who proposed GAN-INT-CLS [32] took the first step in using GANs to design end-to-end differentiable architecture from the character level to pixel level. GAN-INT-CLS [32] combines DC-GAN [30] with a hybrid character-level convolutional-recurrent neural network proposed by Scott Reed et al [31] as a text features encoder. The DC-GAN [30] conditions on the encoded text features. They also proposed a manifold interpolation regularizer to improve the text-to-image synthesis. Their method has been proved to have the capability to generate plausible images of birds and flowers from detailed text descriptions [32].

Instead of just using one generator for text-to-image synthesis, Han Zhang et al put forward StackGAN [41] which provides two different generators. The Stage-I GAN is applied for generating low-resolution images from the given text description while the Stage-II GAN is used to generate high-resolution images with photo-realistic details by taking Stage-I results and text descriptions as inputs. And this model, producing sharper images, outperform the GAN-INT-CLS [32].

– **Style transfer** Conditional Generative Adversarial Networks conditioned on an input image and generate a corresponding output image are suitable for image-to-image translation tasks. These networks not only learn the mapping from input image to ouput image, but also learn a loss function to train this mapping. The pix2pix model [16] is advanced as a general-purpose solution to image-to-image translation problems and has been demonstrated to be applicable in a wide variety of scenarios including semantic segmentation, background removal, palette generation and pose transfer.

  Yet, paired training data will not be available for many image-to-image translation tasks. An early attempt to learn a joint distribution without any tuple of corresponding images is CoGAN [22] which is made of a pair of GANs, each for one image domain. Via enforcing a simple weight-sharing constraint to the layers that responsible for decoding abstract semantics, the CoGAN [22] can learn the joint distribution of images by just using samples drawn separately from the marginal distribution.

  CycleGAN [43] offers another algorithm that can learn to translate between domains without a training set of aligned image pairs. CycleGAN [43] introduces a cycle consistency loss to further reduce the space of possible mapping functions, ensuring that the model will arrive where it started when translating from one domain to the other and back again. This method can achieve striking results in many cases, for example, collection style transfer, objective

transfiguration and photo generation from paintings. Interestingly, there exist other two models called DualGAN [39] and DiscoGAN [18] which have almost the same idea and the similar architecture with CycleGAN [43].
- **Image Super-Resolution** The highly challenging task of estimating a high-resolution image from its low-resolution counterpart is referred to as super-resolution. The first framework capable of inferring photo-realistic natural images for $4 \times upscaling factors$ is SRGAN [21] proposed by Christian Ledig et al, for which they apply a deep residual network (ResNet) with skip-connection and diverge from the mean squared error (MSE) as the sole optimization target. In addition, they define a perceptual loss function which consists of an adversarial loss and a content loss motivated by perceptual similarity in pixel space, using high-level feature maps of the VGG network [4,17,36]. And the extensive mean-opinion-score (MOS) test shows great significant gains in perceptual quality using SRGAN [21].

In addition to its widely use in various computer vision applications today, the framework of GANs can also be employed on a myriad of other areas, including natural language processing (NLP) [10,12,20,40], novelty detection [19,34], speech enhancement [28] and etc. The myriad application domains pose more challenges to the efficient processing, especially training, of GANs; the solutions then have to be adaptive and scalable in order to handle the new and varied forms of GANs that these applications may adopt.

3 Overview of GANs

In this section, I gives an overview of the basic componenets of GANs and conditional GANs. In addition, I will present some popular GAN models currently in use. Fig. 1 shows the architectures of both GAN [13] and cGAN [25].

3.1 Generative Adversarial Nets

The first Generative Adversarial Nets proposed by Ian J. Goodfellow et al [13] consists of a generative model G that captures the data distribution and a discriminative model D that estimates the probability that a sample came from the training data rather than G. Training of this framework involves simultaneously training the generator G to maximize the probability of discriminator D making a mistake and finding the parameters of D to maximize the probability of assigning the correct label to both training examples and samples from G. In other words, this framework corresponds to a two-player minimax game with value function $V(G, D)$:

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

where $p_{data}(x)$ is the generator's distribution over data x while $p_z(z)$ is a prior noise distribution over input noise z. $G(z;\theta_g)$ is a differentiable mapping function from $p_z(z)$ to $p_{data}(x)$ represented by a multilayer perceptron with parameters

θ_g. $D(x;\theta_d)$, another multilayer perceptron, outputs single scalar that represents the probability that x came from the data rather than a learned distribution. In the case where G and D are difined by multilayer perceptrons, the entire system can be trained with backpropagation [13].

3.2 Conditional Generative Adversarial Nets

The first conditional version of Generative Adversarial Nets proposed by Mehdi Mirza et al [25] can be constructed by simplely feeding some extra information y into the both the discriminator and generator as additional input layer. y can be any kind of auxiliary information including class labels, data from other modalities and etc.

In the generator the prior noise $p_z(z)$ and y are combined in joint hidden representation while x and y are presented as inputs in the discriminator. Therefore, the objective function of the two-player minimax game would be as follow:

$$\min_G \max_D V(D,G) = \mathbb{E}_{x\sim p_{data}(x)}[\log D(x \mid y)] + \mathbb{E}_{z\sim p_z(z)}[\log(1 - D(G(z \mid y)))]$$

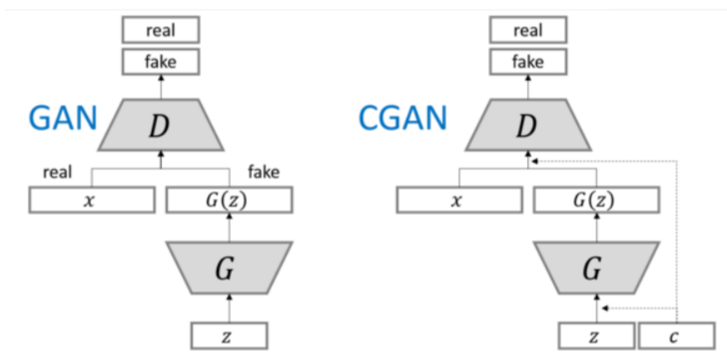

Fig. 1. The left shows the architecture of GAN [13]. The right shows the architecture of Conditional GAN [25]

3.3 Popular GAN models

Many GAN models have been designed over the past four years. In addtion to the aforementioned models including GAN [13], CGAN [25], LAPGAN [8], DiscoGAN [18], SRGAN [21], CoGAN [22], CycleGAN [43], StackGAN [41] and etc., there exist many other popular GAN models.

Deep convolutional generative adversarial networks (DC-GANs) [30], scaling up GANs using CNNs, are a strong candidate for unsupervised learning. The family of architectures is identified by architecture guidelines as follows [30]:

– Replace any pooling layers with strided convolutions (discriminator) and fractional-strided convolutions (generator).
– Use batchnorm in both the generator and the discriminator.
– Remove fully connected hidden layers for deeper architectures.
– Use ReLU activation in generator for all layers except for the output, which uses Tanh.
– Use LeakyReLU activation in the discriminator for all layers.

This model has been proved to learn good representations of images for supervised learning and generative modeling.

Auxiliary Classifier GANs (ACGANs) [27] make a modification to the standard GAN formulation. In the ACGAN architecture, besides the noise z, every generated sample has a corresponding class label, $c \sim p_c$. The generator G use both to generate images $X_{fake} = G(c, z)$ while the discriminator D calculates a probability distribution over sources as well as a probability distribution over the class labels, $P(S \mid X), P(C \mid X) = D(X)$. Therefore, the objective function would consist of two parts: the log-likelihood of the correct source L_S and the log-likelihood of the correct class L_C:

$$L_S = E[\log P(S = real \mid X_{real})] + E[\log P(S = fake \mid X_{fake})]$$

$$L_C = E[\log P(C = c \mid X_{real})] + E[\log P(C = c \mid X_{fake})]$$

D is trained to maximize $L_S + L_C$ and G is trained to maximize $L_C - L_S$. This modification appears to stabilize training. ACGAN model has been demonstrated to generate global coherent ImageNet samples. In addition, it shows that reducing the variability introduced by all 1000 classes of ImageNet significantly improves the quality of training.

InfoGAN [5], which is able to learn disentangled representations in a completely unsupervise manner, is an information-theoretic extension to a standard GAN model. The key idea of InfoGAN is to maximize the mutual information between a fixed small subset of the GAN's noise variables and the observations. In information theory, mutual information between X and Y, $I(X;Y)$, measures the "amount of information" learned from knowledge of random variable Y about the other random variable X. The input noise vector is decomposed into two parts: the source of incompressible noise z and the latent code c which will target the salient structured semantic features of the data distribution. The information-regularized minimax game to solve is put forward as follows:

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

where λ is a hyperparameter while $I(c; G(z, c))$ represents the mutual information between latent code c and the generated sample from the input vector (z, c). Considering the difficulty of maximizing the mutual informatino term $I(c; G(z, c))$ directly in practice, InfoGAN is finally defined with a variational regularization of mutual information and a hyperparameter λ:

$$\min_{G,Q} \max_D V_{InfoGAN}(D, G, Q) = V(D, G) - \lambda L_I(G, Q)$$

where $Q(c \mid x)$ is an auxiliary distribution to approximate the posterior $P(c \mid x)$.

Wasserstein GAN (WGAN) [2] makes significant progress toward stable training of GANs. Arjovsky et al [2] argue that the Jensen-Shannon divergence along with other common distances and divergences are potentially not continuous and thus do not provide a usable gradient for the generator. The Earth Mover (EM) distance, which is intuitively defined as the optimal cost of transporting mass in order to transform the distribution q into the distribution p (where the cost is mass times transport distance), has been theorically demonstrated to have the desirable property that under mild assumptions it is continuous everywhere and differentiable almost everywhere. After applying the Kantorovich-Rubinstein duality, the value function is defined as follow:

$$\max_{w \in \mathcal{W}} V(D,G) = \mathbb{E}_{x \sim \mathbb{P}_r}[f_w(x)] - \mathbb{E}_{z \sim p_z(z)}[f_w(g_\theta(z))]$$

where $\{f_w\}_{w \in \mathcal{W}}$ is a parameterized family of functions that are all K-Lipschitz for some K and specifically in WGAN, $K = 1$. WGAN uses weight clipping to enforce the Lipschitz constraint on the critic (another name for discriminator). In the case that is under an optimal critic, minimizing the value function with respect to the generator parameters minimizes the Earth-Mover distance between $\mathbb{P}_{g_\theta(z)}$ and $\mathbb{P}_r$. The empirical results show that WGAN can improve the stability of learning, get rid of problems like mode collapse and provide meaningful learning curves which correlates with the quality of samples and which is useful for debugging and hyperparameter searches.

Yet, WGAN can still generate low-quality samples or fail to converge in some settings because of the use of weight clipping in WGAN to enforce a Lipschitz constraint on the critic which will cause capacity underuse, gradients exploding and gradients vanishing. How to effectively enforce the Lipschitz constraint on the critic remains an open question. Therefore, WGAN-GP [14] put forward an alternative method that instead of clipping weights, penalize the norm of the gradient of the critic with respect to its input. The new objective for the critic is:

$$L = \underbrace{\mathbb{E}_{\tilde{x} \sim \mathbb{P}_g}[D(\tilde{x})] - \mathbb{E}_{x \sim \mathbb{P}_r}[D(x)]}_{Original\ critic\ loss} + \underbrace{\lambda \mathbb{E}_{\hat{x} \sim \mathbb{P}_{\hat{x}}}[(\parallel \nabla_{\hat{x}} D(\hat{x}) \parallel_2 - 1)^2]}_{gradient\ penalty}$$

This is a soft version since exactly enforcing the Lipschitz constraint is intractable. At certain points sampled from a distribution over the input space $\hat{x} \sim \mathbb{P}_{\hat{x}}$, evaluate the gradient of the critic $\nabla_{\hat{x}} D(\hat{x})$ and penalize its squared distance from 1 in the critic loss function. WGAN-GP [14] converges faster, enables very stable GAN training and generates higher-quality samples than WGAN [2] with weight clipping.

Another model that modifies the loss function for the discriminator D is Least Squares GAN [24]. This model adopts the least squares loss function for the discriminator and it has been proved that minimizing the objective function of LSGAN [24] yields minimizing the Pearson χ^2 divergence. Specifically, the objective function for this model can be defined as follows:

$$\min_D V_{LSGAN}(D) = \frac{1}{2}\mathbb{E}_{x \sim \mathbb{P}_{data}(x)}[(D(x) - b)^2] + \frac{1}{2}\mathbb{E}_{z \sim p_z(z)}[(D(G(z)) - a)^2]$$

$$\min_G V_{LSGAN}(G) = \frac{1}{2}\mathbb{E}_{z\sim p_z(z)}[(D(G(z)) - c)^2]$$

where a and b are the labels for fake data and real data and c denotes the value that G wants D to believe for fake data. LSGANs [24] are able to generate higher quality images than regular GANs. Besides, this model performs more stable during learning process.

There still exist many novel and effective GAN models. Yet subject to the length of this article, I can not introduce all of them in detail. I refer readers to these impressive work, AAE [23], Loss-Sensitive GAN (LSGAN) [29], Energy-Based GAN (EBGAN) [42], Multi-Agent Diverse GAN [11], Denoising Feature Matching GAN [37], ALI model [9] and etc.

4 Issues of GANs & Theorical Analysis

Though GAN models have achieved striking success in various areas, the training process of GAN is known to be unstable. In this section, I will discuss the issues of training of GANs and theorical analysis on it.

4.1 Issues of GANs

The problems that may appear during training are as follows:

- Mode collapse, which means to generate very similar samples for different inputs;
- Hard to converge;
- Updates get worse as the discriminator gets better;
- The genetor and discriminator need to be well defined;
- etc.

These issues have made training of GANs a big challenge. Many works have been conducted to seek a general solution.

4.2 Theorical Analysis

An early analysis is f-GAN [26]. Sebastian Nowozin et al demonstrate that GAN is a special case of an existing more general variational divergence estimation mothod and any f-divergence can be employed when training GAN models.

What is a f-divergence? Given two distributions P and Q which have an entirely continuous density function p and q on a base measure $\mathrm{d}x$ defined on the domain $\mathcal{X}$ respectively, the $f - divergence$ can be defined as follows:

$$D_f(P \parallel Q) = \int_{\mathcal{X}} q(x) f\left(\frac{p(x)}{q(x)}\right) dx$$

where the generator function $f : \mathbb{R}_+ \to \mathbb{R}$ is a convex, low-semicontinuous function satisfying $f(1) = 0$. Table 1 shows the list of f-divergences $D_f(P \parallel Q)$ together with generator functions and the optimal variational functions

Table 1. List of f-divergences $D_f(P \parallel Q)$ together with generator functions and the optimal variational functions

Name	$D_f(P \parallel Q)$	Generator $f(u)$	$T^*(x)$
Total variation	$\frac{1}{2}\int \lvert p(x) - q(x) \rvert dx$	$\frac{1}{2}\lvert u-1 \rvert$	$\frac{1}{2}\text{sign}(\frac{p(x)}{q(x)} - 1)$
Kullback-Leibler	$\int p(x) \log \frac{p(x)}{q(x)} dx$	$u \log u$	$1 + \log \frac{p(x)}{q(x)}$
Reverse Kullback-Leibler	$\int q(x) \log \frac{q(x)}{p(x)} dx$	$-u \log u$	$-\frac{q(x)}{p(x)}$
Pearson χ^2	$\int \frac{(q(x)-p(x))^2}{p(x)} dx$	$(u-1)^2$	$2(\frac{p(x)}{q(x)} - 1)$
Neyman χ^2	$\int \frac{(p(x)-q(x))^2}{q(x)} dx$	$\frac{(1-u)^2}{u}$	$1 - \left[\frac{q(x)}{p(x)}\right]^2$
Squared Hellinger	$\int \left(\sqrt{p(x)} - \sqrt{q(x)}\right)^2 dx$	$(\sqrt{u} - 1)^2$	$(\sqrt{\frac{p(x)}{q(x)}} - 1) \cdot \sqrt{\frac{q(x)}{p(x)}}$
Jeffrey	$\int (p(x) - q(x)) \log \left(\frac{p(x)}{q(x)}\right) dx$	$(u-1) \log u$	$1 + \log \frac{p(x)}{q(x)} - \frac{q(x)}{p(x)}$
Jensen-Shannon	$\frac{1}{2}\int p(x) \log \frac{2p(x)}{p(x)+q(x)} + q(x) \log \frac{2q(x)}{p(x)+q(x)} dx$	$-(u+1)\log \frac{1+u}{2} + u \log u$	$\log \frac{2p(x)}{p(x)+q(x)}$
Jensen-Shannon-weighted	$\int p(x)\pi \log \frac{p(x)}{\pi p(x)+(1-\pi)q(x)} + (1-\pi)q(x) \log \frac{q(x)}{\pi p(x)+(1-\pi)q(x)} dx$	$\pi u \log u - (1 - \pi + \pi u)\log(1 - \pi + \pi u)$	$\pi \log \frac{p(x)}{\pi p(x)+(1-\pi)q(x)}$
GAN	$\frac{1}{2}\int p(x) \log \frac{2p(x)}{p(x)+q(x)} + q(x) \log \frac{2q(x)}{p(x)+q(x)} dx - \log(4)$	$u \log u - (u+1)\log(u+1)$	$\log \frac{p(x)}{p(x)+q(x)}$
α-divergence ($\alpha \notin \{0,1\}$)	$\frac{1}{\alpha(\alpha-1)} \int \left(p(x)\left[\left(\frac{q(x)}{p(x)}\right)^{\alpha} - 1\right] - \alpha(q(x) - p(x))\right) dx$	$\frac{1}{\alpha(\alpha-1)}(u^{\alpha} - 1 - \alpha(u-1))$	$\frac{1}{\alpha-1}\left[\left[\frac{p(x)}{q(x)}\right]^{\alpha-1} - 1\right]$

In addition, they provide experimental insight into comparing different divergence functions on training complexity and the quality of the obtained generative models.

Another theorical analysis is conducted by Martin Arjovsky et al [1]. According to them, if the supports of $\mathbb{P}_r$ and $\mathbb{P}_g$ are disjoint or lie in low dimensional manifolds, there is always a perfect discriminator between them and this will lead to an unreliable training of the generator. Besides, they prove that as the approximation to the optimal discriminator gets better, either vanishing gradients or the massively unstable behaviour occurs in practice, depending on which cost function is employed. For more details of the proofs, readers can refer this work [1].

5 Techniques for Training GANs

Training GANs aims to find a Nash equilibrium to a two-player non-cooperative game. It's very difficult because GAN models have non-convex cost functions and continuous parameters along with exceedingly high-demonsional parameter space. Tim Salimans et al [33] summarize several heuristical techniques that are able to optimize convergence when training GANs as follows:

- **Feature matching** that specifies a new value function for G to prevent it from overattaining on the current D. This can reduce the instability of GANs.
- **Minibatch discrimination** which allows the discriminator to look at multiple data examples in combination rather than in isolation. This can avoid mode collapse to some extend.
- **Historical averaging** makes it able to find equilibria of low-dimensional, continuous non-convex games.
- **One-sided label smoothing** use smoothed values for a classifier instead of the 0 and 1. This can make neural networks to adversarial examples less fragile. [38].
- **Virtual batch normalization** in which each example x normalized based on the statistic collected on a reference batch of examples that are chosen

once and fixed at the start of training and on x itself. This can avoid the highly dependence of the output of a neural network for an input example x to the other inputs x' in the same minibatch when using batch normalization. Yet, it is computationally expensive.

In addtion, Soumith Chintala et al [6] have summarized 16 tricks for training GANs. Readers with some interest can refer to their work.

6 Conclusion

In this paper, I present a survey on Generative Adversarial Networks. Specifically, I discuss why GAN models are proposed, provide an overview of the basic components of GANs as well as its conditional version, give several popular GAN models that are currently in use, discuss the issues of training of GANs and theorical analysis and finally offer some effective tricks that can stabilize and optimize the training of GANs. Within the subtleties of GAN training, there are many opportunities for developments in theory and algorithms, and with the power of deep networks, there are vast opportunities for new applications [7].

Acknowledgments

The author would like to thank all the people who help me with this article.

References

1. Arjovsky, M., Bottou, L.: Towards principled methods for training generative adversarial networks. arXiv preprint arXiv:1701.04862 (2017)
2. Arjovsky, M., Chintala, S., Bottou, L.: Wasserstein gan. arXiv preprint arXiv:1701.07875 (2017)
3. Bengio, Y., Alain, G., Yosinski, J., et al.: Deep generative stochastic networks trainable by backprop. In: Proceedings of the 31st International Conference on International Conference on Machine Learning-Volume 32. pp. II–226. JMLR. org (2014)
4. Borman, S., Stevenson, R.L.: Super-resolution from image sequences-a review. In: Circuits and Systems, 1998. Proceedings. 1998 Midwest Symposium on. pp. 374–378. IEEE (1998)
5. Chen, X., Duan, Y., Houthooft, R., Schulman, J., Sutskever, I., Abbeel, P.: Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In: Advances in Neural Information Processing Systems. pp. 2172–2180 (2016)
6. Chintala, S., Denton, E., Arjovsky, M., Mathieu, M.: How to train a gan. In: NIPS, workshop on generative adversarial networks (2016)
7. Creswell, A., White, T., Dumoulin, V., Arulkumaran, K., Sengupta, B., Bharath, A.A.: Generative adversarial networks: An overview. IEEE Signal Processing Magazine **35**(1), 53–65 (2018)

8. Denton, E.L., Chintala, S., Fergus, R., et al.: Deep generative image models using a laplacian pyramid of adversarial networks. In: Advances in neural information processing systems. pp. 1486–1494 (2015)
9. Dumoulin, V., Belghazi, I., Poole, B., Mastropietro, O., Lamb, A., Arjovsky, M., Courville, A.: Adversarially learned inference. arXiv preprint arXiv:1606.00704 (2016)
10. Fedus, W., Goodfellow, I., Dai, A.M.: Maskgan: Better text generation via filling in the _. arXiv preprint arXiv:1801.07736 (2018)
11. Ghosh, A., Kulharia, V., Namboodiri, V., Torr, P.H., Dokania, P.K.: Multi-agent diverse generative adversarial networks. arXiv preprint arXiv:1704.02906 (2017)
12. Glover, J.: Modeling documents with generative adversarial networks. arXiv preprint arXiv:1612.09122 (2016)
13. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: Advances in neural information processing systems. pp. 2672–2680 (2014)
14. Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., Courville, A.C.: Improved training of wasserstein gans. In: Advances in Neural Information Processing Systems. pp. 5769–5779 (2017)
15. Huang, H., Yu, P.S., Wang, C.: An introduction to image synthesis with generative adversarial nets. arXiv preprint arXiv:1803.04469 (2018)
16. Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1125–1134 (2017)
17. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: European Conference on Computer Vision. pp. 694–711. Springer (2016)
18. Kim, T., Cha, M., Kim, H., Lee, J., Kim, J.: Learning to discover cross-domain relations with generative adversarial networks. arXiv preprint arXiv:1703.05192 (2017)
19. Kliger, M., Fleishman, S.: Novelty detection with gan. arXiv preprint arXiv:1802.10560 (2018)
20. Lamb, A.M., GOYAL, A.G.A.P., Zhang, Y., Zhang, S., Courville, A.C., Bengio, Y.: Professor forcing: A new algorithm for training recurrent networks. In: Advances In Neural Information Processing Systems. pp. 4601–4609 (2016)
21. Ledig, C., Theis, L., Huszar, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J., Wang, Z., et al.: Photo-realistic single image super-resolution using a generative adversarial network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4681–4690 (2017)
22. Liu, M.Y., Tuzel, O.: Coupled generative adversarial networks. In: Advances in neural information processing systems. pp. 469–477 (2016)
23. Makhzani, A., Shlens, J., Jaitly, N., Goodfellow, I., Frey, B.: Adversarial autoencoders. arXiv preprint arXiv:1511.05644 (2015)
24. Mao, X., Li, Q., Xie, H., Lau, R.Y., Wang, Z., Smolley, S.P.: Least squares generative adversarial networks. In: 2017 IEEE International Conference on Computer Vision (ICCV). pp. 2813–2821. IEEE (2017)
25. Mirza, M., Osindero, S.: Conditional generative adversarial nets. arXiv preprint arXiv:1411.1784 (2014)
26. Nowozin, S., Cseke, B., Tomioka, R.: f-gan: Training generative neural samplers using variational divergence minimization. In: Advances in Neural Information Processing Systems. pp. 271–279 (2016)

27. Odena, A., Olah, C., Shlens, J.: Conditional image synthesis with auxiliary classifier gans. arXiv preprint arXiv:1610.09585 (2016)
28. Pascual, S., Bonafonte, A., Serra, J.: Segan: Speech enhancement generative adversarial network. arXiv preprint arXiv:1703.09452 (2017)
29. Qi, G.J.: Loss-sensitive generative adversarial networks on lipschitz densities. arXiv preprint arXiv:1701.06264 (2017)
30. Radford, A., Metz, L., Chintala, S.: Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv preprint arXiv:1511.06434 (2015)
31. Reed, S., Akata, Z., Lee, H., Schiele, B.: Learning deep representations of fine-grained visual descriptions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 49–58 (2016)
32. Reed, S., Akata, Z., Yan, X., Logeswaran, L., Schiele, B., Lee, H.: Generative adversarial text to image synthesis. arXiv preprint arXiv:1605.05396 (2016)
33. Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., Chen, X.: Improved techniques for training gans. In: Advances in Neural Information Processing Systems. pp. 2234–2242 (2016)
34. Schlegl, T., Seeböck, P., Waldstein, S.M., Schmidt-Erfurth, U., Langs, G.: 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)
35. Shrivastava, A., Pfister, T., Tuzel, O., Susskind, J., Wang, W., Webb, R.: Learning from simulated and unsupervised images through adversarial training. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR). vol. 3, p. 6 (2017)
36. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
37. Warde-Farley, D., Bengio, Y.: Improving generative adversarial networks with denoising feature matching (2016)
38. Warde-Farley, D., Goodfellow, I.: 11 adversarial perturbations of deep neural networks. Perturbations, Optimization, and Statistics p. 311 (2016)
39. Yi, Z., Zhang, H., Tan, P., Gong, M.: Dualgan: Unsupervised dual learning for image-to-image translation. arXiv preprint (2017)
40. Yu, L., Zhang, W., Wang, J., Yu, Y.: Seqgan: Sequence generative adversarial nets with policy gradient. In: AAAI. pp. 2852–2858 (2017)
41. Zhang, H., Xu, T., Li, H., Zhang, S., Wang, X., Huang, X., Metaxas, D.N.: Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5907–5915 (2017)
42. Zhao, J., Mathieu, M., LeCun, Y.: Energy-based generative adversarial network. arXiv preprint arXiv:1609.03126 (2016)
43. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. arXiv preprint arXiv:1703.10593 (2017)