2020 IEEE Sixth International Conference on Multimedia Big Data (BigMM)

Recent Developments in Generative Adversarial Networks: A Review (Workshop Paper)

Ankit Yadav[1], Dinesh Kumar Vishwakarma[2]
Department of Information Technology
Delhi Technological University
Delhi, India
ankit4607@gmail.com[1], dinesh@dtu.ac.in[2]

***Abstract*—In recent times, Generative Adversarial Networks (GANs) have created a lot of buzz in the research community. GANs are formulated on the zero-sum game theory, where two neural nets compete against each other. The resultant deep model is capable of generating data similar to any data distribution provided. It utilizes the adversarial learning approach and is far more capable in learning features than the traditional machine learning models. This review focusses on the origin and evolution of GANs. Firstly, the traditional GAN is explored in terms of its structure and loss functions. Then come the common challenges of training GANs. Thirdly, the review dives into numerous GAN variants and explains their improvements. The review then lists the wide variety of applications and ends with the conclusion.**

Keywords—Generative Adversarial Networks (GANs), Deep Learning, Machine Learning, Generative Models, Semi-supervised.

I. Introduction

Data generation is one of the key challenges for machines to be more human-like. Whether one requires to enhance decade old gray-scale photographs to coloured versions, identify a permanently occluded person from a crowd automatically or compose creative lines of poetry from scratch, the ability to generate samples closely resembling a real distribution is of paramount importance. Generated samples also help to enrich the diversity of input samples used to train deep models for classification tasks [1] [2]. Traditional machine learning based generative models include the Boltzmann Machine (RBM) [3], Naïve Bayes Model (NBM) [4] and many more. While these models achieved some progress, their effectiveness decreased with the use of large scale datasets. As researchers have turned to deep approaches of data generation, two major models have prevailed, the Variational Autoencoeers (VAE) [5] and Generative Adversarial Networks (GAN) [6]. The VAE uses variational Bayesian inference to regulate the lower limit of log-likelihood instead of optimizing the log-likelihood itself. Such an approach leads to fuzzy samples lacking in the finer details. In contrast to VAE, GANs use an adversarial approach to model a data distribution. Two adversaries, the generator and discriminator compete against each other. As the two players improve, so does the quality of generated samples. While GANs have seen their fair share of challenges, the models are extremely popular due to the variety of structural possibilities and wide applications. This review is devoted to highlight this importance.

This review contributes the following: Firstly, the origin of traditional GAN is explored. The structural arrangements of two neural networks is specified along with the loss functions. Since the purpose of GAN is to imitate a data distribution, standard measures of evaluating the similarity of distributions is discussed. Secondly, the review analyses the most prevalent challenges of training GANs. Thirdly, the review dives into a wide variety of GANs highlighting their improvements, both structural as well as theoretical. Finally, the review features numerous applications of GAN.

Table 1. Abbreviations and Citations

Short Form	Full Name	Reference
GAN	Generative Adversarial Network	Goodfellow et al. [6]
WGAN	Wasserstein GAN	Arjovsky et al. [7]
WGAN-GP	WGAN with Gradient Penalty	Gulrajani et al. [8]
EM	Earth Mover distance	Arjovsky et al. [7]
DCGAN	Deep Convolutional GAN	Radford et al. [9]
CNN	Convolutional Neural Network	Radford et al. [9]
CGAN	Conditional GAN	Mirza et al. [10]
ACGAN	Auxiliary Classifier GAN	Odena et al. [11]
InfoGAN	Information Maximizing GAN	Chen et al. [12]
EBGAN	Energy Based GAN	Zhao et al. [13]
BEGAN	Boundary Equilibrium GAN	Berthelot et al. [14]
LSGAN	Least Square GAN	Mao et al. [15]
LAPGAN	Laplacian Pyramid of GAN	Denton et al. [16]
CycleGAN	Cycle-Consistent GAN	Zhu et al. [17]
StackGAN	Stacked GAN	Zhang et al. [18]
JR-GAN	Jacobian Regularization GAN	Nie et al. [19]
BWGAN	Banach Wasserstein GAN	Adler et al. [20]
TripleGAN	Triple GAN	Li et al. [21]
SNGAN	Spectral Normalization for GAN	Miyato et al. [22]
DRAGAN	Degenerate Avoided GAN	Kodali et al. [23]
CapsGAN	Capsule GAN	Saqur et al. [24]
SRGAN	Super Resolution GAN	Ledig et al. [25]
DiscoGAN	Discovery GAN	Kim et al. [26]
SGAN	Supervising GAN	Chavdarova et al. [27]
BiGAN	Bidirectional GAN	Donahue et al. [28]
SemGAN	Semantically Consistent GAN	Cherian et al. [29]
SAGAN	Self Attention GAN	Zhang et al. [30]
DEGAN	Decoder Encoder GAN	Zhong et al. [31]
f-GAN	f-divergence GAN	Nowozin et al. [32]
Δ-GAN	Triangle GAN	Gan et al. [33]
MGAN	Mixture GAN	Hoang et al. [34]
GMAN	Generative Multi-Adversarial Network	Durugkar et al. [35]
SeqGAN	Sequence GAN	Yu et al. [36]
GRAN	Generative Recurrent Adversarial Networks	Im et al. [37]
SEGAN	Speech Enhancement GAN	Pascual et al. [38]
STGAN	Selective Transfer GAN	Liu et al. [39]
D2GAN	Dual Discriminator GAN	Nguyen et al. [40]

978-1-7281-9325-0/20/$31.00 ©2020 IEEE
DOI 10.1109/BigMM50055.2020.00068

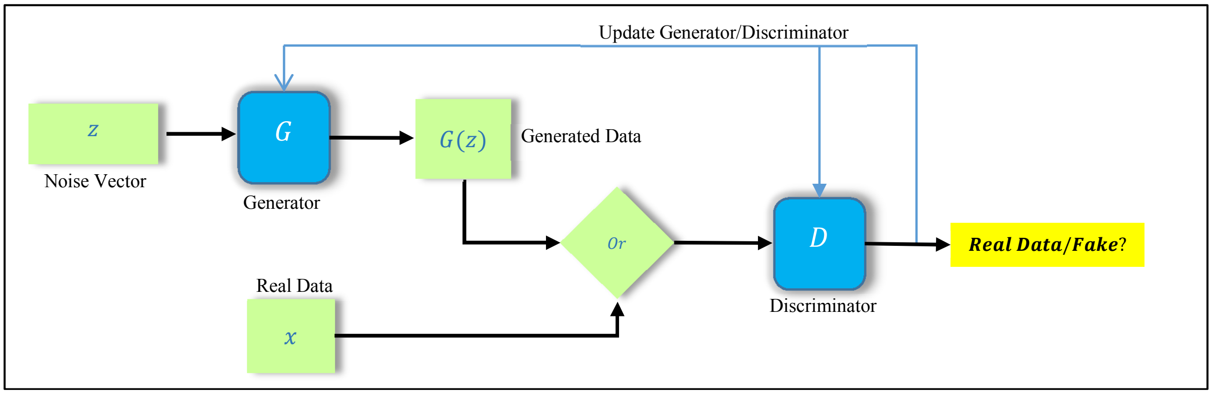

Fig. 1. The Fundamental Generative Adversarial Network

The organization of this review is as follows. Section II outlines the fundamental idea behind GANs. Section III highlights the common, recurring challenges of training GANs. Section IV lists the most commons variations of the traditional GAN [6] based on structural or theoretical modifications. Section V is used to specify the wide area of applications of GANs and conclusion is discussed in section VI.

II. The Fundamentals of GAN

A. Key Idea

GAN is motivated from the min max zero-sum game theory between two opponents. This game theory ensures that the progress of one equates to the loss of the other. In GAN, two neural networks namely, discriminator and generator, compete against each other. The generator aims to generate data samples that closely resemble those from a real dataset. Given that the generator performs this task well, it becomes a challenging task to distinguish generated samples from samples of real dataset, which is the job of the discriminator. The discriminator is fed with inputs repeatedly, some from the real dataset and others as generated fake samples and it must be able to differentiate between them. The discriminator output is a probability value indicating that the input sample is from the real data set. Hence, a strong discriminator's output will be close to 1 if the input is from the real dataset and close to 0 for fake generated input. If the generator produces samples that are easily caught by the discriminator, the generator is made to improve through training. If the discriminator cannot tell the difference between fake/generated and real samples, it is improved to get better in detecting the difference. This adversarial game continues with the objective of finding a Nash equilibrium [41] such that the generated samples resemble real data samples perfectly. Now the discriminator always outputs the probability ½ for each input, signifying that it cannot determine which input is real and which one is fake. At this point, the generator is capable of generating real-like data.

B. The Generator and Discriminator

Figure 1 features the structure of a basic GAN. The generator can be any differentiable function G, thereby allowing the use of neural networks. It takes a randomly selected noise vector z as input and gives the output $G(z)$ which represents the generated sample. The generator G aims to produce samples that perfectly match those from a real dataset.

The discriminator D, another differential function takes sample inputs either from the real dataset as x or generated samples from the generator as $G(z)$. It produces a scalar output giving a real sample input probability value. Hence, its job is to maximize $P(x)$ to 1 and minimize $P(G(z))$ to 0. Conversely, the generator G wants to maximize $P(G(z))$. The performance of generator and discriminator can be iteratively improved and optimized in this adversarial manner [42].

C. Loss Function

Let p_{data} represent the real distribution, p_g be the distribution of generated data and x is the sample from real dataset. The generator G is a differentiable function which inputs noise z. The discriminator D produces a scalar output which gives the probability that source of x is the real data distribution. The loss function $V(D,G)$ [6] is defined as:

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

For any given generator G, the optimal discriminator [6] is defined by the following equation:

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

Also, the training criterion defined by $C(G) = \max_D(G,D)$ has a global minimum value of $-\log 4$ at a point where $p_g = p_{data}$ meaning that the generating model perfectly follows the given data distribution [6].

D. Divergence of Probability Distributions

The central idea of GAN is to minimize the difference of p_{data} and p_g distributions. Here, the term 'distance' does not mean the difference between two points in any spatial plane. It represents the difference between the two distributions and minimizing this difference brings p_g closer to p_{data}. Two metrics to quantify this difference is discussed below:

1) Kullback-Leibler or KL divergence

The KL divergence measures the difference between two probability distributions, say p and q.

$$D_{KL}(p \parallel \mathrm{q}) = \int_x p(x) \log\frac{p(x)}{q(x)} dx \quad (3)$$

D_{KL} is zero for perfectly identical probability distributions. This means that $p(x) = q(x)$ for all values of x. From (3) it is clear that KL divergence is asymmetric i.e.,

$$\int_x p(x) \log \frac{p(x)}{q(x)} dx \neq \int_x q(x) \log \frac{q(x)}{p(x)} dx.$$

2) *Jensen-Shannon (JS) Divergence*

The JS divergence provides another method to quantify dissimilarity between two distributions.

$$D_{JS}(p \parallel \mathrm{q}) = \frac{1}{2} D_{KL}\left(p \parallel \frac{p+q}{2}\right) + \frac{1}{2} D_{KL}\left(q \parallel \frac{p+q}{2}\right) \quad (4)$$

JS divergence is bounded by [0,1] and unlike its counterpart, it is both symmetric and smoother. Switching from KL to JS divergence is a major reason for the massive success of GAN as generative models [43].

III. Challenges of Training GAN

A. *Attaining Nash Equilibrium is challenging*

The discriminator and generator fight in an adversarial manner to get the better of each other. At any given step, either the generator is improved (gradient descent) keeping the discriminator constant or the discriminator learns while the generator stays the same. This concurrent learning of the two new neural networks does not guarantee a convergence to Nash Equilibrium. Saliman et al. [44] propose *feature matching* that prevents the generator from overtraining for any given discriminator using statistics of real data distribution where the discriminator specifies the statistics worth matching. Usually the statistic under consideration is the expected value (mean) of the features.

B. *Frequent Mode Collapse*

Mode collapse is one of the most common pitfalls of GAN. The generator learns a parameter setting that allows it to produce a certain sample appearing highly realistic to the discriminator. Since this generated sample easily confuses the discriminator, the generator starts generating this same fake sample again and again. As a result, the generator fails to learn of the entire data distribution and focuses only on producing the high scoring sample. Since the discriminator works on single inputs, it has no idea if it is receiving same input repeatedly. Saliman et al. [44] address this problem through *mini-batch discrimination*. The discriminator not only receives its singular input but also has an overall side information of input mini batches. This allows the discriminator to know if the generator is generating same sample repeatedly and make it generate different samples.

C. *The Problem of Vanishing Gradient*

If the discriminator works perfectly, it is always correct in tagging real samples as real and generated samples as fake. No matter how good the generated samples are, it is able to classify them as fake i.e.,

$$D(x) = 0, if\ x \epsilon p_g \quad (5)$$

Hence, the loss falls to zero and the generator gradient vanishes leading to no learning. Arjovsky et al. [45] propose to add noise € to real data p_{data} which removes the problem of vanishing gradient. However, the training is highly sensitive to different types of noise that can be used.

IV. Variants of GAN

A. *Wasserstein GAN*

Wasserstein GAN (WGAN) is a variant of the traditional GAN model. It solves the problem of mode collapse by utilizing the Earth Mover (EM) distance instead of the JS divergence. The EM (also known as the Wasserstein-1) can be defined by the following expression [7]:

$$W(p_{data}, p_g) = inf_{\gamma \epsilon \Pi(p_{data}, p_g)} [\mathbb{E}_{(x,y)\sim\gamma}[\parallel x - y \parallel]] \quad (6)$$

Here, $\Pi(p_{data}, p_g)$ represents the set of all joint distributions possible between p_{data} and p_g. $\gamma \epsilon \Pi(p_{data}, p_g)$ describes the dirt transport plan and the infimum function inf indicates the minimization of cost. $\gamma(a, b)$ specifies the dirt proportion to be moved from a to b so as to match b distribution.

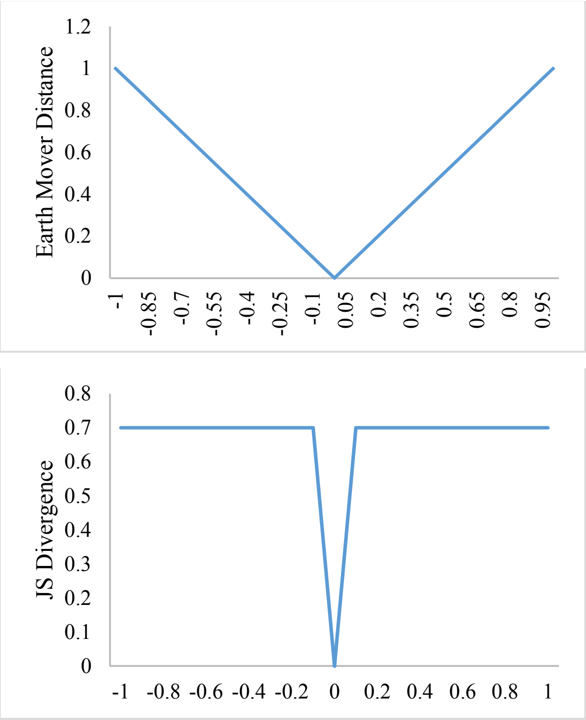

Fig. 2. Plot shows the Earth Mover distance (top) against the JS divergence (bottom). The Earth Mover plot is smoothly continuous and with gradient that can be used for backprop everywhere. The JS plot is discontinuous and gives hard to use gradient values [7].

WGAN outperforms the traditional model in the following ways: Firstly, EM distance works for disjoint data distributions that do not overlap. Secondly, WGAN prevents training instability and creates diverse samples. In WGAN, weight clipping is used to enforce the Lipschitz constraint. This can sometimes cause the vanishing gradient problem for a large number of layers or batch normalization is not used [7].

B. *WGAN with Gradient Penalty*

WGAN defined in the previous section showed signs of training instability in certain cases [7]. In the words of the authors, the weight clipping mechanism is “clearly a terrible way” of enforcing the Lipschitz constraint. Gulrajani et al. [8]

propose a modified WGAN termed as the WGAN with Gradient Penalty (WGAN-GP). The WGAN-GP replaces the weight clipping with a penalty mechanism to enforce the Lipschitz constraint. Any given function is said to 1-Lipschitz only when its gradient norm does not exceed the value of 1 anywhere. This constraint is enforced through a penalty to restrict the gradient norm associated with the critic's output with respect to its input. The critic loss function is [8]:

$$L = \mathbb{E}_{\tilde{x}\sim P_g}[D(\tilde{x})] - \mathbb{E}_{x\sim P_{data}}[D(x)] + \lambda\mathbb{E}_{\hat{x}\sim P_{\hat{x}}}[(\| \nabla_{\hat{x}} D(\hat{x}) \|_2 - 1)^2] \quad (7)$$

Equation (7) shows the constraint enforced with the gradient norm penalty on the random samples $\hat{x} \sim P_{\hat{x}}$. The first two terms represent the original critic loss while the third term containing λ represents the gradient penalty [8]. WGAN-GP skips using the batch normalization since the model performs better without it. WGAN-GP has shown better training stability and greater variety in generated samples than those of WGAN. However, its convergence rate is slower than WGAN.

C. Deep Convolutional GAN

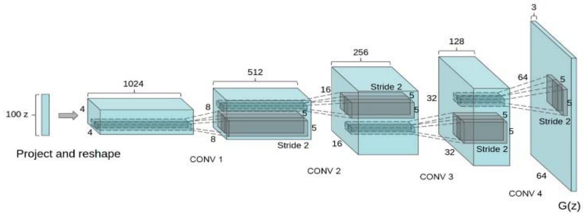

Fig. 3. Generator of DCGAN [9].

Convolutional Neural Networks (CNNs) have seen massive success in supervised learning. The proposal of deep convolutional GANs is an attempt to transfer this success into the domain of unsupervised learning [9]. The DCGAN imposes a set of structural constraints on the CNN architecture. Instead of pooling functions, the network uses *strided convolutions* to learn its down sampling. The strided convolutions cause no loss of accuracy when compared with pooling functions [46]. *Global average pooling* increases DCGAN's stability. The model uses *batch normalization* to handle poor initialization and is capable of preventing the mode collapse issue which is a common problem in GANs. However, batch normalization when applied to all layers led to model instability. Hence, the output layer in generator and the input layer in discriminator are exempted. All layers of DCGAN utilize the ReLU function [47] of activation barring the output layer having Tanh activation. LeakyReLU [48] is used for all layers of discriminator [9]. Due to its unsupervised nature, DCGAN performance is evaluated by using it in the form of a feature extractor on the CIFAR-10 dataset and then comparing linear model performances on the those features. This approach yields 82.8% accuracy which is better than all K-means based approaches. Similarly, DCGAN surpasses the state of the art on the SVHN (Street View House Number) dataset [49] with a low 22.8% test error.

D. Conditional GAN

The conditional GAN [10] adds a condition to the generation and discrimination process of GANs. The condition can be in the form of class labels, data from different modals etc. Figure 4 shows the structure of CGAN that adds new input to the generator and the discriminator.

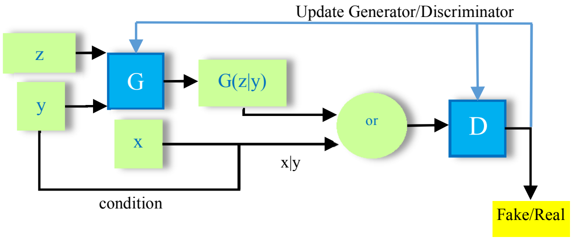

Fig. 4. Structure of CGAN.

The condition c acts as a target for the generator and helps to increase the convergence speed of the model. CGAN has been used for innovative generative applications such as generating key phrases from research papers using title and abstract [50].

E. Auxiliary Classifier GAN

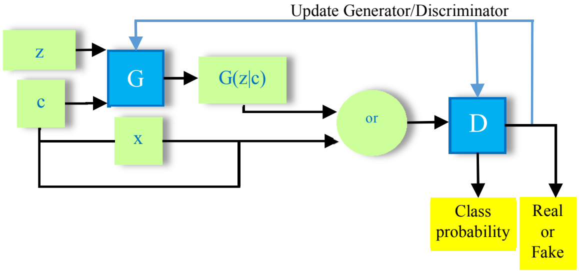

Fig. 5. Structure of ACGAN.

Odena et al. [11] propose Auxiliary Classifier GAN (AC-GAN), a GAN variant to improve training for image synthesis. The generator G takes two inputs, noise z and and a class label $c \sim P_c$ for each generated sample $X_{fake} = G(c,z)$. The discriminator outputs two probabilities, probability distribution over source of input (real/fake) and class labels probability. Hence, the objective function has two components, one for source L_S and the other for class L_C [11].

$$L_S = \mathbb{E}[\log P(S = real \mid X_{real})] + \mathbb{E}[\log P(S = fake \mid X_{fake})] \quad (8)$$

$$L_C = \mathbb{E}[\log P(C = c \mid X_{real})] + \mathbb{E}[\log P(C = fake \mid X_{fake})] \quad (9)$$

The discriminator D tries to maximize $L_S + L_C$. At the same time, generator is trained to maximize $L_C - L_S$. Such a structural modification allows for subdivision of massive datasets into various classes and training discriminator and generator accordingly for each subset. The ACGAN is capable of generating both diverse and high resolution discriminable images in a semi-supervised manner, which are not merely a naïve resizing of low resolution images.

F. Information Maximizing GAN

Chen et al. [12] proposed the InfoGAN which aims to understand interpretable disentangled representations in a fully unsupervised fashion. To achieve this, the mutual information between observed values and small fixed part of noise variable z is maximized. Two inputs, noise z and latent code c are fed into the generator $G(z,c)$. The goal is to have a high mutual information between the generator distribution $G(z,c)$ and latent code c. The min max game for the InfoGAN can be expressed as under [12].

$$\min_{G}\max_{D} V_I(D,G) = V(D,G) - \lambda I(c;G(z,c)) \qquad (10)$$

The proposed model performs well on challenging datasets like MNIST, SVHN and CelebA [12].

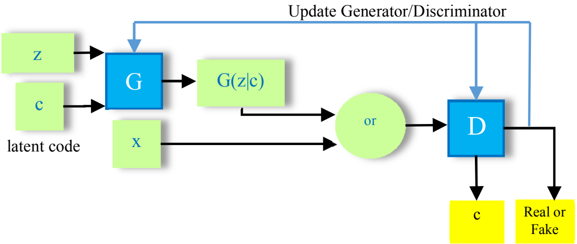

Fig. 6. Structure of InfoGAN.

G. Energy Based GAN

Zhao et al. [13] propose an energy based GAN variant. Any energy based model attempts to develop a function mapping individual points in the input space to scalar values. These scalar values are termed as "energy". Desired configurations are kept at low energies while undesired are restricted to high values. In this variant of GAN, both the generator and discriminator are treated as energy function. The generator tries to generate low energy samples while discriminator tries to push generated samples to high energy and keeps the real data samples to low energy. The discriminator output is fed into a margin loss to shape the energy function. The loss functions are [13]:

$$L_D(x,z) = D(x) + [m - D(G(z))]^+ \qquad (11)$$

$$L_G(x,z) = D(G(z)) \qquad (12)$$

Here, $[a]^+ = \max(0,a)$, m represents a positive value of margin, x is the real sample, $G(z)$ represents generated sample, L_G and L_D are the discriminator loss and generator loss respectively. Authors demonstrate the EBGAN framework by using an auto-encoder architecture in place of discriminator with energy as reconstruction error. EBGANs depict higher scalability and convergence rate in generating high resolution images compared to traditional GANs.

H. Boundary Equilibrium GAN

Berthelot et al. [14] utilize the auto-encoder based discriminator from EBGAN [13] and the Wasserstein loss from WGAN [7] to develop a new equilibrium enforcing method based on proportional control theory. This method defines a new approximation of convergence measure, ensures stable training and high quality sample generation. The generator and discriminator loss are in equilibrium if [14]:

$$\mathbb{E}[L(x)] = \mathbb{E}[L(G(z)] \qquad (13)$$

A new parameter $\gamma \epsilon$ [0,1] known as the "diversity ratio" [14] is defined as:

$$\gamma = \frac{\mathbb{E}[L(G(z))]}{\mathbb{E}[L(x)]} \qquad (14)$$

The diversity ratio γ allows the discriminator to meet its two fold objective, differentiating between real and generated samples and auto-encoding real images. The "convergence measure" $\mathcal{M}_{global}$ [14] has been defined as:

$$\mathcal{M}_{global} = L(x) + |\,\gamma L(x) - L(G(z_G))| \qquad (15)$$

The BEGAN is successful in converging to visually pleasant images having detailed diversity. Training is robust to parameter changes, stable and fast to converge even for high resolution image generation.

I. Least Square GAN

Mao et al. [15] solve the problem of gradient disappearance in traditional GAN. In traditional GAN, the discriminator uses sigmoid cross entropy loss function. The cause of this problem are generated samples which are far away from the real samples despite being on the correct side of decision boundary. The proposed Least Square Generative Adversarial Network (LSGAN) replaces the above loss with least square loss function. Minimizing this objective function is equivalent to minimizing Pearson $\mathcal{X}^2$ divergence. The LSGAN loss is [15]:

$$\min_{D} V_{LSGAN}(D) = \frac{1}{2}\mathbb{E}_{x\sim P_{data}(x)}[(D(x)-b)^2] + \frac{1}{2}\mathbb{E}_{z\sim P_z(z)}[(D(G(z))-a)^2] \qquad (16)$$

$$\min_{G} V_{LSGAN}(G) = \frac{1}{2}\mathbb{E}_{z\sim P_z(z)}[(D(G(z))-c)^2] \qquad (17)$$

Here, a represents fake data label, b represents real data label and value c is the value G wants D to see as fake.

Certain samples existing on the correct decision boundary side are not penalized by traditional GAN despite being far from real samples. LSGAN is able to penalize such data samples. Also, penalizing such samples allows to

TABLE 2. GAN VARIANTS, IMPROVEMENTS, ADVANTAGE, APPLICATION & DATASETS USED

GAN variants	Improvements	Datasets	Advantage	Application
GAN [6]	Adversarial training	MNIST [51], Toronto Face Database, CIFAR10 [52]	Infinite data modelling capacity theoritically	Generate high quality samples
WGAN [7]	Wasserstein distance	LSUN [53]	Increased stability of optimization	
WGAN-GP [8]	Gradient Penalty	Swiss Roll toy [54], LSUN [53], Google Billion Word [55]	Reliable training for different types of GAN architectures	

GAN variants	Improvements	Datasets	Advantage	Application
DCGAN [9]	CNN structures for generator & discriminator, strided convolution, batch normalization	LSUN [53], ImageNet [56], new Face dataset [57]	Stable training, immune to poor initialization	
CGAN [10]	Sample generation based on a condition (label, data modality)	MNIST [51], MIR Flickr	Convergence is faster due to target condition	
ACGAN [11]	Additional class label input to generator, discriminator gives two probability outputs	ImageNet [56], CIFAR10 [52]	Training and Data Generation for various subsets of large datasets	
InfoGAN [12]	Generated sample and given latent code *c* share high mutual information	MNIST [51], SVHN [49]	Learns interpretable disentangled data representations using mutual information based on information theory	
BWGAN [20]	WGAN-GP in Banach Spaces	CIFAR10 [52], CelebA [58]	Features can be specified to the generator	
SNGAN [22]	Spectral Normalization	CIFAR10 [52], STL-10, ImageNet [56]	Better quality image generation using "spectral normalization" weight optimization which is computationally light	
DRAGAN [23]	Improved gradient penalty by avoiding undesirable local equilibria	MNIST [51], CelebA [58],	Fewer mode collapse and faster convergence	
BEGAN [14]	EBGAN discriminator & Wasserstein distance	CelebA [58]	Generating visually pleasant images with great diversity	
LSGAN [15]	Least Square loss function	LSUN [53], HWDB1.0 [59]	Solves the vanishing gradient problem	
LAPGAN [16]	Laplacian pyramid of cGANs, Gaussian Parzen window estimate,	LSUN [53]	successive improvements in generated samples at each layer of pyramid	
f-GAN [32]	Variational divergence estimate (f-divergence), simplified saddle-point optimization	MNIST [51], LSUN [53]	Generalizes generative adversarial approach to variational divergence estimation approach	
JR-GAN [19]	Avoiding *Phase factor* and *Conditioning factor* of Jacobian which leads to non-convergence (Jacobian regularization-JARE)	CIFAR10 [52], CelebA [58], ImageNet [56]	Stable training and convergence	
EBGAN [13]	Generator & Discriminator as energy functions, assign low energy for realistic samples	ImageNet [56], LSUN [53]	High rate of convergence and scalability to create images of high resolution	
SGAN [27]	Global generator-discriminator pair trained against multiple local generator-discriminator pairs	mixtures of M Gaussians (M-GMM), Swiss Roll toy [54]	Global G & D pair continue to learn even when a local pair fails to converge	
BiGAN [28]	Learns inverse mapping from data to latent representation	MNIST [51], ImageNet [56]	Unsupervised BiGAN performs better visual feature learning than existing weak-supervised models	
SAGAN [30]	Self Attention	ImageNet [56]	Improved training due to spectral normalization in G training	
CycleGAN [17]	Cycle Consistency Loss	Cityscapes [60], CMP Facade Database [61], ImageNet [56], UT Zappos50K [62]	Unsupervised way of unpaired image to image translation	Style transfer, Image Translation
StackGAN [18]	Two stage GANs, first GAN to generate basic shape & second GAN to generate fine texture based details	CUB , Oxford-102 [63], COCO [64]	Generation of sharp images due to improvement in two phases	Text to Image Transformation
TripleGAN [21]	Three player game, contains a Classifier along with generator & discriminator	MNIST [51], SVHN [49], CIFAR10 [52]	Model can disentangle styles and classes thereby achieving state of the art classification results	Style transfer
CapsGAN [24]	Capsule Network, Binary Cross Entropy, Mean Squared Error	MNIST [51], SmallNORB	Model performs better than DCGAN to generate images with geometric transformations	Image reconstruction
SRGAN [25]	16 block deep ResNet, perceptual loss function	Set5 [65], Set14 [66], BSD100 [67]	High quality image super-resolution for large upscaling factors (upto 4 times)	Image super resolution
DiscoGAN [26]	Combination of GAN & GAN with reconstruction loss	Car [68], Face [57], CelebA [58], Chair [69], Handbags [70], Shoes [62]	Model discovers cross domain relation without labelling of data	Style transfer, Image Translation
SemGAN [29]	Semantic dropout, semantic consistency loss	Cityscapes [60], Mitsubishi Precision, VIPER [71]	Produces high quality translation	Image to Image Translation

GAN variants	Improvements	Datasets	Advantage	Application
DEGAN [31]	Variational Bayesian Inference, Decoder-Encoder architechture	MNIST [51], CelebA [58], CIFAR10 [52]	Input vector of G is closer to data distribution than Gaussian noise, Better quality image generation than DCGAN samples	Very high quality image generation
∆-GAN [33]	Two pairs of generator & discriminator, joint distribution matching	MNIST [51], edges2shoes [72], CelebA [58], COCO [64]	Model learns mapping between two domains in a bidirectional manner using few sample pairs.	Image translation, attribute conditional image generation
MGAN [34]	Multiple generators, Jensen-Shannon divergence in generator	CIFAR10 [52], STL-10, ImageNet [56]	Solves the mode collapse problem	Generate large diversity samples
GMAN [35]	Multiple discriminators, Generative Multi Adversarial Metric (GMAM)	MNIST [51], CIFAR10 [52], CelebA [58]	Faster convergence to a high quality state	High quality image generation in a fraction number of iterations
SeqGAN [36]	Stochastic policy for data generation using reinforcement learning	Nottingham dataset	Generates structured sequence of data.	Model generates creative sequences in poems, speech and music domain.
GRAN [37]	Recurrent feedback loop in generator, Generative Adversarial Metric (GAM)	MNIST [51], CIFAR10 [52], LSUN [53]	Produces high quality images	Image generation
SEGAN [38]	Speech enhancement with GAN	Dataset from Valentini et al. [73]	Quick enhancement process	Speech enhancement
STGAN [39]	Input is a difference attribute vector, reconstruction loss, attribute manipulation loss	CelebA [58]	Specific attribute editing using selective transfer units (STUs)	High quality image reconstruction
D2GAN [40]	Two discriminator with one generator, 3 player game, combines KL & reverse KL divergence	MNIST [51], CIFAR10 [52], STL-10, ImageNet [56]	Solves the mode collapse problem	Improved quality and diversity of generated samples.

generate more gradients, helping the generator to improve and thereby solving the vanishing gradient problem. This leads to increased stability in training. Two LSGAN architectures are proposed. First one generates 112×112 resolution images that are evaluated on various scene datasets. The second one is able to generate readable characters from a Chinese character dataset with 3470 classes (characters).

J. *Other Variants*

GAN is an ongoing area of research and hence, it is challenging to capture all variations in a single research. Besides the major GAN variants mentioned above, Table 2 provides numerous variations in GAN along with their improvements, advantages and applications.

V. Applications of GAN

The fundamental of GAN is to match a given data distribution. If successful, GAN structures then have infinite data modelling power from those data distribution, at least theoretically. GANs are widely used in numerous applications. Table 2 shows a detailed list of GAN variants along with their applications.

A. *Classification*

GANs are capable of performing image classification tasks. A fully trained GAN's discriminator can extract features of data from supervised datasets and then evaluating linear model performances for those features [9].

B. *Image Synthesis*

The fundamental purpose of GAN is data generation. By learning the data distribution of any dataset, GANs can model new samples that resemble those from original datasets. Multiple variants of GAN such as CGAN, WGAN, WGAN-GP, LAPGAN, ACGAN, InfoGAN are all capable of generating data samples.

C. *Style Transfer (Image Translation)*

Style Transfer, also known as Image Translation aims to propagate one image style to another. This means to modify images into different forms like from real photographs to facial sketches. Conditional GANs like CycleGAN, TripleGAN, DiscoGAN etc are well suited for such operations. GAN has been used to perform style transfer to user images in a manner that preserves the theaesthetic beauty desired by users [74].

D. *Super Resolution*

Super resolution is another challenge in the field of computer vision applications. Super resolution refers to generating a high resolution equivalent of a low resolution input image. The task is not merely about increasing the dimensions but also adding finer details missing from the low resolution input. GAN models like SRGAN have succeeded in upscaling images by a factor of four.

E. *Reconstruction*

Image reconstruction means to add missing details to an incomplete/lower domain image to enhance it. A major area of such application is face generation where a model tries complete a missing region of face by finding the closest vector. CapsGAN has shown good reconstruction capabilities.

F. *Text to Image Generation*

Generating images through text is another exciting application. Generating face image from fine-grained textual descriptions is just one example of numerous possibilities in this application area [75]. StackGAN utilizes two-level GANs to do the task. First pair of generator and discriminator is responsible to draw an outline image given some input text. The second GAN is responsible fine tuning details of the image being created.

G. Other Applications

GANs can be used in any scenario that requires data generation. GANs have been used to for visual speech recognition by generating visual samples of unseen classes to increase training capability of a visual speech recognition system [76]. GANs are used widely to enhance datasets where the number of samples is limited. It can also transform two dimensional images to three dimensions. GAN has also been used to enhance the performance of hash functions for venue discovery [77].

VI. Conclusion

GAN aims to learn from massive unlabeled data distributions. The ability to create endless samples from any data distribution has tremendous applications in numerous fields. GANs solve the problems of insufficient data, improving the quality of generated data. However, certain challenges still lie in formulating a perfect GAN. Firstly, measuring the difference between two data distributions is an ongoing topic of debate. The JS divergence fails for disjoint distributions. While the Wasserstein distance solves this issue, it brings a new set of constraints that may lead to challenge of model instability and vanishing gradient. Secondly, no universal metric has been realized which is capable of evaluating a GAN model in its entirety. Different metrics evaluate different aspects of GAN models but there is a sure need of introducing generalizability of metric. This review dives into the concept of GANs, explaining the underline idea and basic structure of traditional GAN. Next come the numerous variants of GAN, each defining a specific improvement on the previous version(s). Finally, the various applications are described and discussed.

VII. References

[1] A. K. Gautam, L. Misra, A. Kumar, K. Misra, S. Aggarwal and R. R. Shah, "Multimodal Analysis of Disaster Tweets," in *International Conference on Multimedia Big Data*, Singapore, 2019.

[2] M. Agarwal, M. Leekha, R. Sawhney and R. R. Shah, "Crisis-DIAS: Towards Multimodal Damage Analysis - Deployment, Challenges and Assessment," in *AAAI Conference on Artificial Intelligence*, New York, 2020.

[3] R. Salakhutdinov and G. Hinton, "Deep Boltzmann Machines," in *Twelfth International Conference on Artificial Intelligence and Statistics*, Florida, 2009.

[4] L. Jiang, H. Zhang and Z. Cai, "A Novel Bayes Model: Hidden Naive Bayes," *IEEE Transactions on Knowledge and Data Engineering,* vol. 21, no. 10, pp. 1361-1371, 2009.

[5] Y. Pu, . Z. Gan, . R. Henao, . X. Yuan, C. Li, A. Stevens and . L. Carin, "Variational Autoencoder for Deep Learning of Images, Labels and Captions," in *Advances in Neural Information Processing Systems*, Barcelona, 2016.

[6] 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*, Montreal, 2014.

[7] M. Arjovsky, S. Chintala and L. Bottou, "Wasserstein Generative Adversarial Networks," in *Proceedings of the 34th International Conference on Machine Learning*, Sydney, 2017.

[8] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin and A. C. Courville, "Improved Training of Wasserstein GANs," in *Neural Information Processing Systems*, California, 2017.

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

[10] M. Mirza and S. Osindero, "Conditional Generative Adversarial Nets," *arXiv preprint arXiv:1411.1784,* 2014.

[11] A. Odena, C. Olah and J. Shlens, "Conditional Image Synthesis with Auxiliary Classifier Gans," in *34th International Conference on Machine Learning*, Sydney, 2016.

[12] X. Chen, Y. Duan, R. Houthooft, . J. Schulman, I. Sutskever and P. Abbeel, "Infogan: Interpretable Representation Learning by Information Maximizing Generative Adversarial Nets," in *Advances in Neural Information Processing Systems 2016*, Barcelona, 2016.

[13] J. Zhao, M. Mathieu and Y. LeCun, "Energy-Based Generative Adversarial Network," *arXiv preprint arXiv:1609.03126,* 2016.

[14] D. Berthelot, T. Schumm and L. Metz, "BEGAN: Boundary Equilibrium Generative Adversarial Networks," *arXiv preprint arXiv:1703.10717,* 2017.

[15] X. Mao, Q. Li, H. Xie, R. Y. Lau, Z. Wang and S. Paul, "Least Squares Generative Adversarial Networks," in *IEEE International Conference on Computer Vision*, Venice, 2017.

[16] E. L. Denton, S. Chintala and R. Fergus, "Deep Generative Image Models using a Laplacian Pyramid of Adversarial Networks," in *Advances in neural information processing systems*, Montreal, 2015.

[17] J. Y. Zhu, T. Park, P. Isola and A. A. Efros, "Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks," in *2017 IEEE International Conference on Computer Vision (ICCV)*, Venice, 2017.

[18] H. Zhang, T. Xu, H. Li, S. Zhang, X. Wang, X. Huang and D. Metaxas, "StackGAN: Text to Photo-Realistic Image Synthesis with Stacked Generative Adversarial Networks," in *2017 IEEE International Conference on Computer Vision (ICCV)*, Venice, 2017.

[19] W. Nie and A. B. Patel, "Towards a Better Understanding and Regularization of GAN Training Dynamics," in *Conference on Uncertainty in Artificial Intelligence 2019*, Tel Aviv, Israel, 2019.

[20] J. Adler and S. Lunz, "Banach Wasserstein GAN," in *Advances in Neural Information Processing Systems*, Montreal, 2018.

[21] C. LI, T. Xu, J. Zhu and B. Zhang, "Triple Generative Adversarial Nets," in *Advances in Neural Information Processing Systems*, California, 2017.

[22] T. Miyato, T. Kataoka, M. Koyama and Y. Yoshida, "Spectral Normalization for Generative Adversarial Networks," in *International Conference on Learning Representations (ICLR)*, Canada, 2018.

[23] N. Kodali, J. Abernethy, J. Hays and Z. Kira, "On Convergence and Stability of GANs," *arXiv preprint arXiv:1705.07215v5,* 2017.

[24] R. Saqur and S. Vivona, "Capsgan: Using Dynamic Routing for Generative Adversarial Networks," in *Computer Vision Conference (CVC): Science and Information Conference*, Las Vegas, 2019.

[25] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang and W. Shi, "Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Hawaii, 2017.

[26] T. Kim, M. Cha, H. Kim, J. K. Lee and J. Kim, "Learning to Discover Cross-Domain Relations with Generative Adversarial Networks," in *Proceedings of the 34th International Conference on Machine Learning*, Sydney, 2017.

[27] T. Chavdarova and F. Fleuret, "SGAN: An Alternative Training of Generative Adversarial Networks," in *IEEE international conference on Computer Vision and Pattern Recognition (CVPR)*, Utah, 2018.

[28] J. Donahue, P. Krähenbühl and T. Darrell, "Adversarial Feature Learning," in *5th International Conference on Learning Representations*, Toulon, France, 2017.

[29] A. Cherian and A. Sullivan, "Sem-GAN: Semantically-Consistent Image-to-Image Translation," in *IEEE Winter Conference on Applications of Computer Vision*, Waikoloa Village, HI, USA, 2019.

[30] H. Zhang, I. Goodfellow, D. Metaxas and A. Odena, "Self-Attention Generative Adversarial Networks," in *International Conference on Machine Learning*, California, 2019.

[31] G. Zhong, W. Gao, Y. Liu and Y. Yang, "Generative Adversarial Networks with Decoder-Encoder Output Noise," *arXiv preprint arXiv:1807.03923,* 2018.

[32] S. Nowozin, B. Cseke and R. Tomioka, "f-GAN: Training Generative Neural Samplers using Variational Divergence Minimization," in *Advances in neural information processing systems*, Barcelona, 2016.

[33] Z. Gan, L. Chen, W. Wang, Y. Pu, Y. Zhang, H. Liu, C. Li and L. Carin, "Triangle Generative Adversarial Networks," in *Advances in Neural Information Processing Systems*, California, 2017.

[34] Q. Hoang, N. T. Dinh, L. Trung and P. Dinh, "MGAN: Training Generative Adversarial Nets with Multiple Generators," in *6th International Conference on Learning Representations*, Vancouver, 2018.

[35] I. Durugkar, I. Gemp and S. Mahadevan, "Generative Multi-Adversarial Networks," in *5th International Conference on Learning Representations*, Toulon, France, 2017.

[36] L. Yu, W. Zhang, J. Wang and Y. Yu, "SeqGAN: Sequence Generative Adversarial Nets with Policy Gradient," in *AAAI Conference on Artificial Intelligence*, California, 2017.

[37] D. J. Im, C. D. Kim, H. Jiang and R. Memisevic, "Generating Images with Recurrent Adversarial Networks," *arXiv preprint arXiv:1602.05110,* 2016.

[38] S. Pascual, A. Bonafonte and J. Serrà, "SEGAN: Speech Enhancement Generative Adversarial Network," in *18th Annual Conference of the International Speech Communication Association*, Stockholm, 2017.

[39] 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*, Long Beach, CA, USA, 2019.

[40] T. D. Nguyen, T. Le, . H. Vu and D. Q. Phung, "Dual Discriminator Generative Adversarial Nets," in *Advances in Neural Information Processing Systems*, Long Beach, CA, USA, 2017.

[41] J. Nash, "Two-Person Cooperative Games," *Econometricia,* vol. 21, no. 1, pp. 128-140, 1953.

[42] A. Krizhevsky, I. Sutskever and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *Advances in Neural Information Processing Systems*, Nevada, 2012.

[43] F. Huszár, "How (not) to train your generative model: Scheduled sampling, likelihood, adversary?," *arXiv preprint arXiv:1511.05101,* 2015.

[44] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford and X. Chen, "Improved Techniques for Training GANs," in *Neural Information Processing Systems*, Barcelona, 2016.

[45] M. Arjovsky and L. Bottou, "Towards Principled Methods for Training Generative Adversarial Networks," *arXiv:1701.04862,* 2017.

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

[47] V. Nair and G. E. Hinton , "Rectified Linear Units Improve Restricted Boltzmann Machines," in *International Conference on Machine Learning*, Washington, 2010.

[48] A. L. Maas, A. Y. Hannun and . A. Y. Ng, "Rectifier nonlinearities improve neural network acoustic models," in *ICML Workshop on Deep Learning for Audio, Speech and Language Processing*, Atlanta, 2013.

[49] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu and A. Y. Ng, "Reading digits in natural images with unsupervised feature learning," in *NIPS Workshop on Deep Learning and Unsupervised Feature Learning*, 2011.

[50] A. Swaminathan, R. K. Gupta, H. Zhang, D. Mahata, R. Gosangi and R. R. Shah, "Keyphrase Generation for Scientific Articles Using GANs," in *AAAI Conference on Artificial Intelligence*, New York, 2020.

[51] Y. LeCun, L. Bottou, Y. Bengio and P. Haffner, "Gradient-Based Learning Applied to Document Recognition," *Proceedings of the IEEE,* vol. 86, no. 11, pp. 2278-2324, 1998.

[52] A. Krizhevsky, "Learning Multiple Layers of Features from Tiny Images," *Technical report, University of Toronto,* vol. 1, no. 4, p. 7, 2009.

[53] F. Yu, A. Seff, Y. Zhang, S. Song, T. Funkhouser and J. Xiao, "LSUN: Construction of a Large-scale Image Dataset using Deep Learning with Humans in the Loop," *arXiv preprint arXiv:1506.03365,* 2015.

[54] S. Marsland, Machine Learning: An Algorithmic Perspective, Chapman and Hall/CRC, 2009.

[55] C. Chelba, T. Mikolov, M. Schuster, Q. Ge, T. Brants and P. Koehn, "One Billion Word Benchmark for Measuring Progress in Statistical Language Modeling," in *arXiv preprint arXiv:1312.3005*, 2013.

[56] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision,* vol. 115, no. 3, p. 211–252, 2015.

[57] . P. Paysan, R. Knothe, . B. Amberg, S. Romdhani and . T. Vetter, "A 3D Face Model for Pose and Illumination Invariant Face Recognition," in *Sixth IEEE International Conference on Advanced Video and Signal Based Surveillance*, Genova, 2009.

[58] Z. Liu, P. Luo, X. Wang and X. Tang, "Deep Learning Face Attributes in the Wild," in *2015 IEEE International Conference on Computer Vision*, Santiago, 2015.

[59] C.-L. Liu, F. Yin, Q.-F. Wang and D.-H. Wang, "ICDAR 2011 Chinese Handwriting Recognition Competition," in *2011 International Conference on Document Analysis and Recognition*, Beijing, China, 2011.

[60] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth and B. Schiele, "The Cityscapes Dataset for Semantic Urban Scene Understanding," in *2016 IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, 2016.

[61] R. Tylecek and R. Sára, "Spatial Pattern Templates for Recognition of Objects with Regular Structure," in *Pattern Recognition - 35th German Conference*, Saarbrücken, Germany, 2013.

[62] A. Yu and . K. Grauman, "Fine-Grained Visual Comparisons with Local Learning," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, Columbus, 2014.

[63] M.-E. Nilsback and A. Zisserman, "Automated Flower Classification over a Large Number of Classes," in *Sixth Indian Conference on Computer Vision, Graphics & Image Processing*, Bhubaneswar, 2008.

[64] T. Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár and C. L. Zitnick, "Microsoft COCO: Common Objects in Context," in *European Conference on Computer Vision*, Zurich, 2014.

[65] M. Bevilacqua, A. Roumy, C. Guillemot and M.-L. Alberi-Morel, "Low-Complexity Single-Image Super-Resolution based on Nonnegative Neighbor Embedding," in *British Machine Vision Conference*, Surrey, 2012.

[66] R. Zeyde, M. Elad and M. Protter, "On Single Image Scale-Up Using Sparse-Representations," in *International Conference on Curves and Surfaces*, Avignon, France, 2010.

[67] D. R. Martin, C. C. Fowlkes, D. Tal and J. Malik, "A Database of Human Segmented Natural Images and its Application to Evaluating Segmentation Algorithms and Measuring Ecological Statistics," in *Proceedings of the Eighth International Conference On Computer Vision*, Vancouver, 2001.

[68] S. Fidler, S. J. Dickinson and . R. Urtasun, "3D Object Detection and Viewpoint Estimation with a Deformable 3D Cuboid Model," in *Conference on Neural Information Processing Systems*, Lake Tahoe, Nevada, 2012.

[69] M. Aubry, . D. Maturana, . A. A. Efros, . B. C. Russell and J. Sivic, "Seeing 3D Chairs: Exemplar Part-Based 2D-3D Alignment Using a Large Dataset of CAD Models," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, Columbus, 2014.

[70] 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*, Amsterdam, 2016.

[71] S. R. Richter, Z. Hayder and V. Koltun, "Playing for Benchmarks," in *IEEE International Conference on Computer Vision*, Venice, 2017.

[72] P. Isola, J. Y. Zhu, T. Zhou and A. A. Efros, "Image-to-Image Translation with Conditional Adversarial Networks," in *2017 IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, HI, USA, 2017.

[73] C. V. Botinhao, X. Wang, S. Takaki and J. Yamagishi, "Investigating RNN-Based Speech Enhancement Methods for Noise-Robust Text-to-Speech," in *The 9th ISCA Speech Synthesis Workshop*, Sunnyvale, CA, USA, 2016.

[74] S. Pai, N. Sachdeva, R. Shah and R. Zimmermann, "User Input Based Style Transfer While Retaining Facial Attributes," in *International Conference on Multimedia Big Data*, Singapore, 2019.

[75] O. R. Nasir, S. K. Jha, M. S. Grover, Y. Yu, A. Kumar and R. R. Shah, "Text2FaceGAN: Face Generation from Fine Grained Textual Descriptions," in *International Conference on Multimedia Big Data*, Singapore, 2019.

[76] Y. Kumar, D. Sahrawat, S. Maheshwari, D. Mahata, A. Stent, Y. Yin, R. R. Shah and R. Zimmermann, "Harnessing GANs for Zero-Shot Learning of New Classes in Visual Speech Recognition," in *AAAI Conference on Artificial Intelligence*, New York, 2020.

[77] H. Aggarwal, R. R. Shah, S. Tang and F. Zhu, "Supervised Generative Adversarial Cross-Modal Hashing by Transferring Pairwise Similarities for Venue Discovery," in *International Conference on Multimedia Big Data*, Singapore, 2019.

[78] D. Berthelot, T. Schumm and L. Metz, "Began: Boundary Equilibrium Generative Adversarial Networks," *arXiv preprint arXiv:1703.10717,* 2017.