CHAPTER 5

A review of the techniques of images using GAN

Rituraj Soni[a] and Tanvi Arora[b]
[a]Department of CSE, Engineering College Bikaner, Bikaner, Rajasthan, India
[b]Department of CSE, CGC College of Engineering, Landran, Mohali, Punjab, India

5.1 Introduction to GANs

The generative adversarial networks (GANs) are the models that have been constructed for the image-to-image translations. They are considered a powerful class of neural networks implemented for the purpose of unsupervised learning. The concept of GAN was introduced by Ian J. Goodfellow [1] in 2014. It can be divided into three parts.

- *Generative*: It describes how the data is generated.
- *Adversarial*: The process of the training of the model is carried out in a competitive manner.
- *Networks*: It is use of the deep learning neural network for training process.

GAN basically consists of the two networks: generator network and a discriminator network as shown in Fig. 5.1. Both these networks try to compete with each other and in this process they also train each other through multiple cycles of generation and discrimination. The generator network aims at generating new images, text, audio, etc. These new items (text, audio, and images) are fake in nature. The discriminator checks these images with the help of a training model, whether these images are fake or real. It does this analysis with the help of the feedback and loss functions.

Figs. 5.2–5.4 display the output obtained from different types of GANs. Fig. 5.2 displays the transformation of one object into another, depending on the given inputs. Similarly, in Fig. 5.3, the GAN generates high-resolution images. Lastly, in Fig. 5.4, the GAN performs the image-to-image translation and thus contributes to an increase in the dataset, which is close to realistic images.

5.1.1 Need for GANs

The GANs have gained popularity over just 2–3 years. They have the capability to generate very realistic images and videos that can assist in implementing the image editor or processor in our tablets or smartphones. The GANs have the capability of modeling and data distribution, and can produce clearer and sharper images. The GANs can train any

Generative Adversarial Networks for Image-to-Image Translation
https://doi.org/10.1016/B978-0-12-823519-5.00006-3

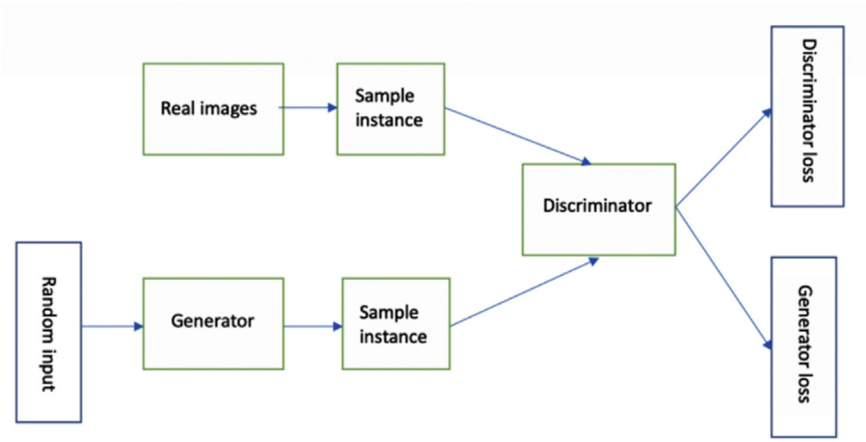

Fig. 5.1 Basic structure of GAN [2].

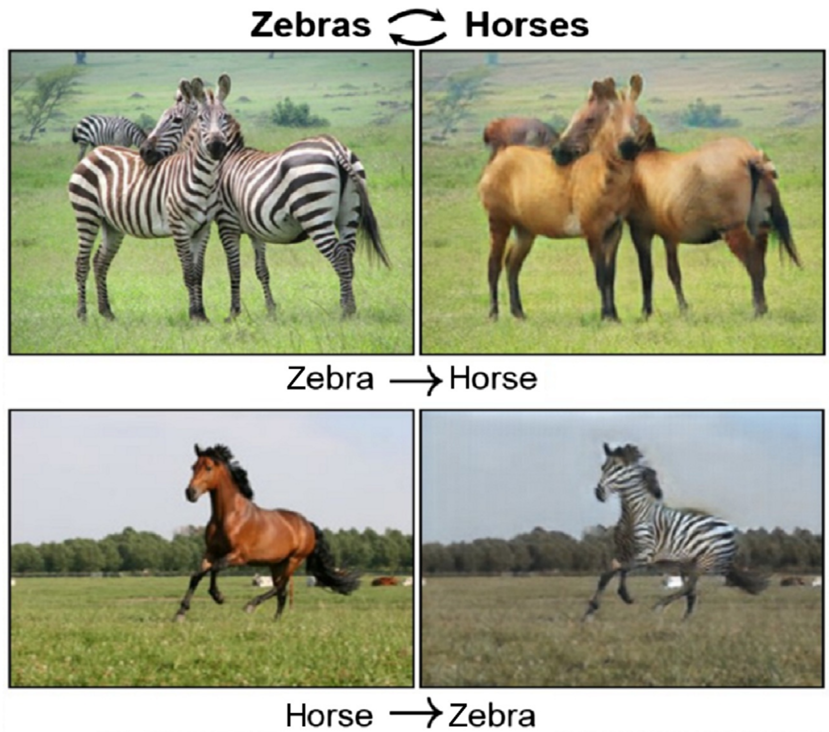

Fig. 5.2 Example of GANs transforming zebra to horse [3].

Fig. 5.3 Example of GANs generating high-resolution images [3].

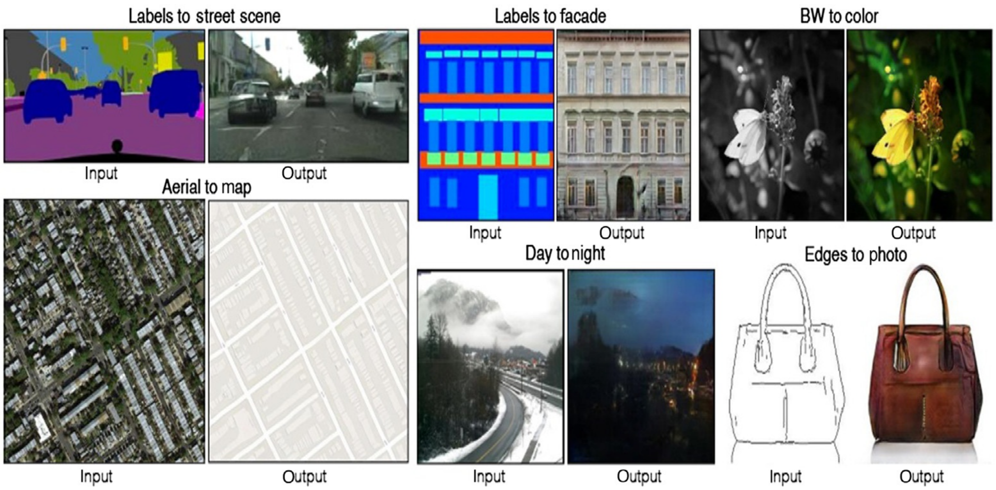

Fig. 5.4 Example of image-to-image translation [3].

type of generator network, with no limitation, whereas other techniques have limitations on the generator networks and can only be used in specific cases. Moreover, the GAN models do not depend on the Markov chain, which is used to generate the samples.

The earlier advantages associated with the GAN models make them promising solutions for the generation of the image dataset, which are required for training the deep learning models that require a large number of items to be trained. The cost of physically collecting and labeling the items is quite high, whereas the GAN can help to generate the items of the dataset with minimum effort and quite low cost. The GANs can also help to generate the face photos, cartoon characters, photos of emojis, automatically generate the models for advertisements, and all these activities can be just done by feeding in the base photo. The different variants can be automatically generated with the help of GAN.

The GAN models are also needed for the purpose of photo editing; they can aid in making the photos clearer and improve the resolution of the images as well. That can be used to derive meaningful information from otherwise unclear images. They can help the researchers generate a large number of images that appear to be real with input given in the form of the sketch or semantic images. Apart from that, the GAN can also be used to generate the images from the text descriptions. The image-to-image conversion can also be carried out with the help of the GANs. The GAN models can be used for photo editing to such an extent that one can produce different kinds of images related to the variation in facial expressions, gestures, lip movements, gender, hair colors, etc.

Therefore, it can be comprehended that the GAN models are needed for generating the synthetic datasets, for an image-to-image conversion, for text-to-image conversion, for editing blurred or low-resolution images, to forecast the looks of an individual after a certain age and to generate the 3D models.

The ultimate need of the GAN is for generating the data that can be used to train the neural network-based models, as the accuracy of the neural network models depends upon the effectiveness of the training data. On the contrary, the success of the GAN application depends on the extent of the training of the GAN architecture; if not carried out perfectly, the results may not be good enough to carry out research on real-time applications.

In Section 5.2, the various architectures related to the GANs are discussed with their underlying models and working.

5.2 GAN architectures

This section provides an essential insight into the working and modeling of the different architecture of the GANs. Each architecture has its working style thus contributing to the generation of the images to create datasets in various research problems.

5.2.1 Fully connected GANs

The basic concept in the research scenario related to the GANs field is the utilization of the deep convolutional neural networks (CNNs) for the process of the image synthesis tasks. Therefore, in this traditional approach, the pooling layers and the fully connected layers 4,5 are removed or minimized from the GANs. Barua et al. [6] proposed the use of the fully connected convolution net architecture for the GANs (FCC-GANs), by stating that the implementation of these multiple fully connected layers along with the convolution layers gives better performance than the conventional architecture.

In case of the conventional GANs, the single process of deep convolution generates the images. However, the work proposed by Barua et al. [6] states the two-step process method for image generation using FCC-GANs. The first step states ways to obtain the high-dimensional image features with the help of the low-dimensional input noise. The second step involves the generation of the image features using these high-dimensional features. These fully connected layers help to understand the relation between the input noise features and, thus, the generation of the final image features, which are closer to the natural images. The convolution layers cannot achieve this global mapping operation due to emphasis on the local connectivity. The methodology given by Barua et al. [6] accomplishes the following aims:

- The use of the fully connected and the convolution layers is proposed that generates higher-level images on different benchmark datasets as compared to the existing GAN methods.
- The learning rate of the FCC-GANs is higher than the conventional GANs. The former also produces very high-quality realistic images within a few rounds (epoch) of training.
- The FCC-GANs give better results on the parameters such as Fretchet inception distance and inception score compared to existing CNN architecture on the benchmark datasets.
- The architecture proposed as fully GANs is robust and stable as compared to existing CNN architecture.

A simple example of the FCC-GANs proposed by Barua et al. [6] is shown in Fig. 5.5 and that of the conventional GANs is shown in Fig. 5.6. These models create $32 \times 32 \times 3$ RGB images from the random noise vector z. In Fig. 5.5, the number of the nodes is denoted by the number in the boxes. Whereas, in the case of conventional architecture, as shown in Fig. 5.6, the number in the boxes indicates the shape of the output layers. The FCC-GANs shown in Fig. 5.5 can be utilized for the images of different resolutions by changing the depth and shape of the convolution stack.

The experiments in Barua et al. [6] are carried out on the four datasets MNIST [7], CIFAR-10 [8], SVHN [9], and Celeb A [10]. It has been proved by experiments on these datasets that FCC-GANs produces higher quality images and converges faster than the

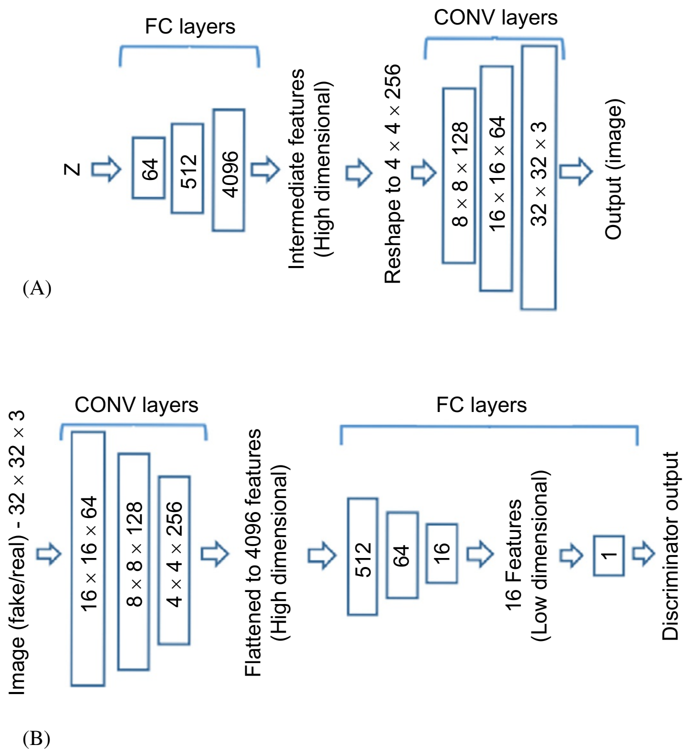

Fig. 5.5 (*Top*) FCC-GAN generator; (*Bottom*) FCC-GAN discriminator [6].

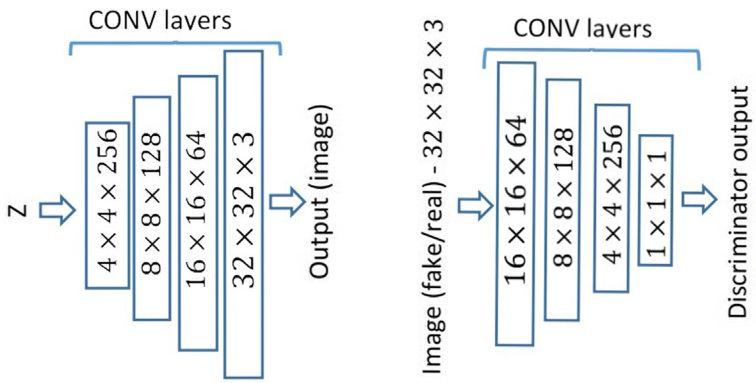

Fig. 5.6 (*Left*) Conventional generator; (*Right*) conventional discriminator [6].

traditional GANs approach. The stability of the FCC-GANs has been proved using different parameters to indicate the importance of the FCC-GANs in Pix2Pix image generation. The most important advantage of the FCC-GANs is that it can be associated with any GAN method, and it can also be used in complex networks such as ResNet [11].

5.2.2 Conditional GANs

The concept of the conditional generative adversarial network (CGANs) by Mirza and Osindero [12] was first introduced to the world by Mehdi Mirza and Simon Osindero. This idea is an augmentation on the GAN. It is implemented in the machine learning domain for the training of the image-to-image generative models.

In the traditional GANs model, there are no conditions applied to the generator and discriminator and there is no control on the types pf data generated by such GANs. Thus, if the given framework does not require such data, it is just a waste of effort. Whereas, in the case of the CGANs, a condition can be applied to both the generator and discriminator. These conditions can be based on the same class labels of the image or some other property [13]. Therefore, the available GANs model can be converted into the CGANs by applying other additional conditions to the generator and discriminator y. This extra conditional information can be applied to both generator and discriminator. It can be seen in Fig. 5.7 that along with input z, a condition is also applied to GANs to convert it into the CGANs.

Another example of the CGANs is shown in Fig. 5.8; here the condition y is added to the generator as well as the discriminator for the desired output.

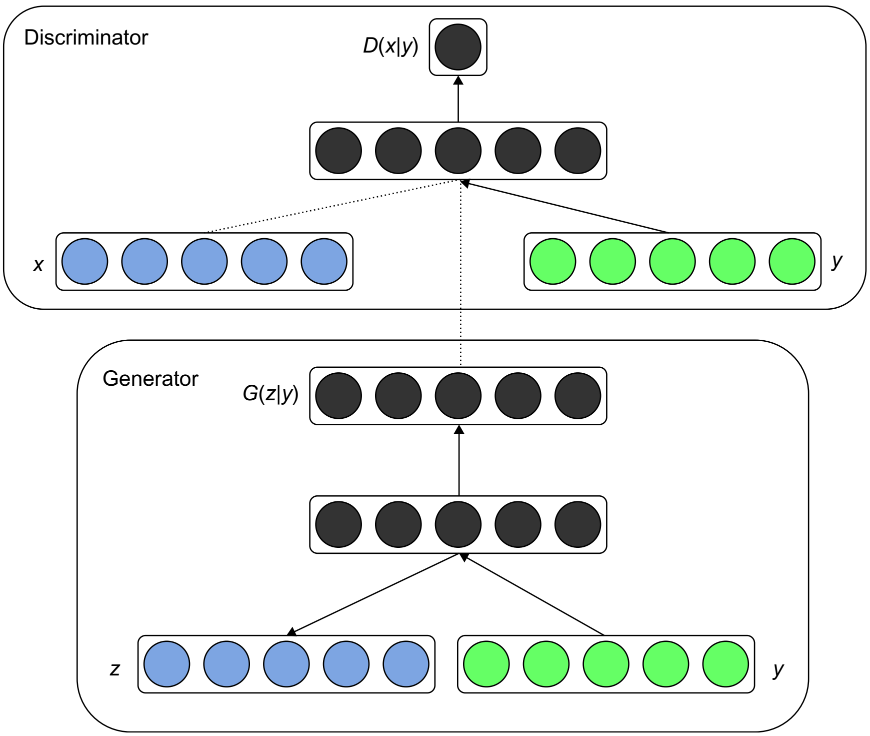

Fig. 5.7 An example of the conditional adversarial net [12].

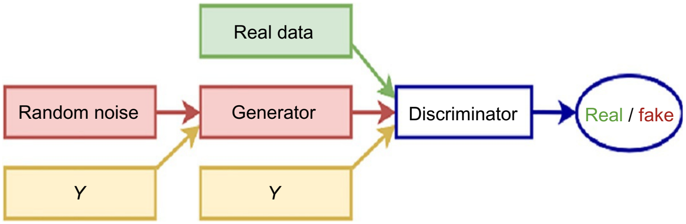

Fig. 5.8 An example of the conditional adversarial net [14].

The factors for the construction of the CGANs can be as follows:

- The first and foremost is to add features or conditions to control the output and direct the generator to produce images as per the given conditions.
- These features or sets of features should be available from the images that classify them into specific classes such as images of human beings if the aim is to create the face of the imaginary actors, etc. It can contain features like the complexion of the hairs and the type of the eyes, etc.
- The information, as well as the data that will learn, can also be incorporated in the images and into the inputs.
- The evaluation of the discriminator is performed on the similarity of the fake and the real data. It also takes into account the mapping of the input features with the fake data image.
- The condition can be imposed on both the input of the generator and discriminator. It can in the form of digits forming a vector (condition) and is linked as a real or fake image to the given generator or discriminator.

Fig. 5.9 depicts the output of the generation of the digits [12] using the MNIST datasets with the help of the CGANs. The CGANs suffer from one disadvantage: they always need labels to perform the work, as they are completely unsupervised.

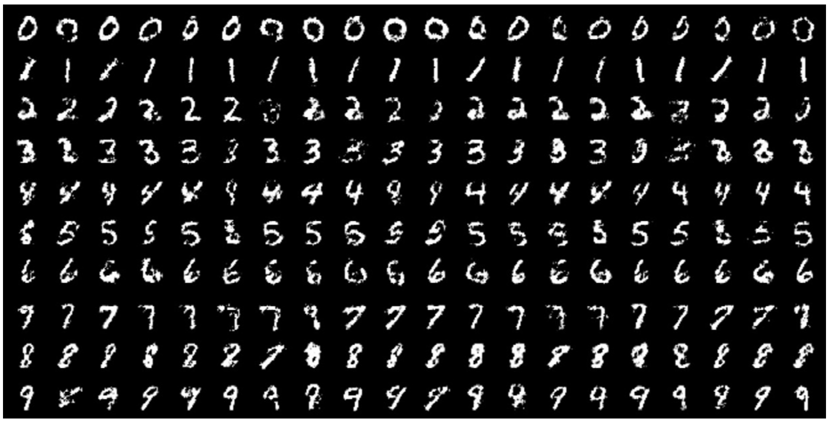

Fig. 5.9 MNIST digit generation using CGANs [12].

5.2.3 Adversarial autoencoders

Adversarial autoencoder [15] is a probabilistic autoencoder that uses GAN to perform variational inference by matching the aggregated posterior of the hidden code vector of the autoencoder with an arbitrary prior distribution. Autoencoders [16] works on the similar approach of the feed-forward neural networks and uses the concepts of unsupervised learning. The autoencoder's main task is to encode information related to the input in between the architecture and deconstruct that information by best means to the output.

As shown in Fig. 5.10, the first layer is used for encoding the information (up to the middle layer), and therefore it is known as encoder as it is used for encoding information. The middle layer in the given architecture is termed an encoded vector. The end layer from the output of the middle layer is termed as the decoder. The end layer assists in the reconstruction of the information available through code. So the input layer of the autoencoder after receiving the data is sent to the autoencoder's middle layer and that middle layer is essential as this layer has data which has reduced dimension.

Makhzani et al. [15] propose another variation of the GANs called adversarial autoencoders (AAE) that converts an autoencoder into the generative model. The job of the autoencoders is to generate new random data with the help of given input data. The only

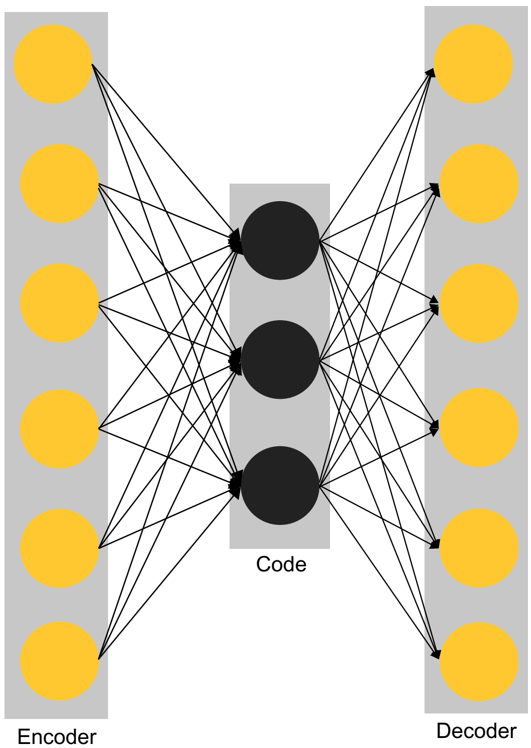

Fig. 5.10 A simple encoder [16].

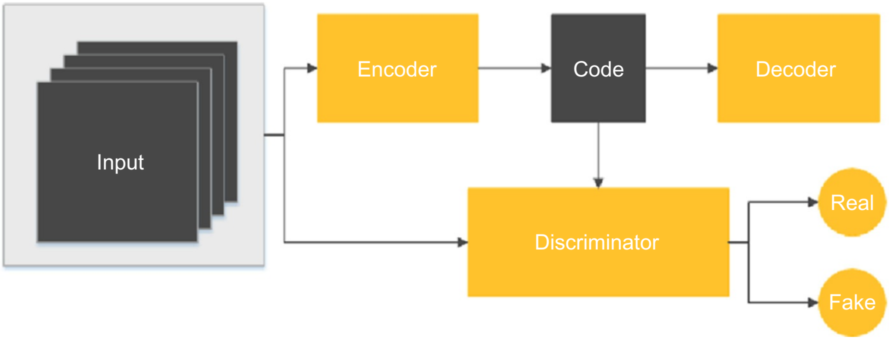

Fig. 5.11 A simple adversarial autoencoder [16].

variation of the GANs with AAEs is that the latter controls the encoder output with the assistance of the prior distribution. This encoded vector is comprised of the mean value and standard deviation, and now along with this, it also has a prior distribution function. On the other hand, the decoder can map the prior (imposed) distribution to the data distribution with the help of the deep generative model. The type of distribution for the prior distribution can be any distribution, for example, normal distribution, gamma distribution, Gaussian distribution, etc. The prominent concept is the settlement of the distribution of the encoded values in the direction of the prior distribution. Therefore, the detector performs mapping of the prior distribution with the data distribution.

Fig. 5.11 demonstrates the simple AAE, in which the standard autoencoder is placed at the top row, and it is generating the image x as per information given by the latent code z. The second network is set at the bottom row to discriminate the fact, that whether the sample is coming from the sampled distribution specified by the user or hidden code of the autoencoder.

Makhzani et al. [15] have proposed that the AAE, attains the competitive test probabilities on Toronto Face Dataset [17] and real-valued MNIST datasets. The proposed method can be applied to the semisupervised scenarios. It obtains excellent semisupervised classification performance on SVHN and MNIST datasets. The AAEs find applications in dimensionality reduction and data visualization and disentangle the content and style of images, and unsupervised clustering.

5.2.4 Deep convolution GANs

The GANs, as discussed in the earlier section, consist of the primary two networks, generator and discriminator, to carry out different works. To make the GANs more powerful, to accomplish the more complex applications, both the generator and discriminator will be augmented with the convolutional neural network layers. This structure is known as deep convolution GANs. The concept of the deep convolution GANs (DCGANs) is floated by Radford et al. [18] in 2015, and they succeed in utilizing the ConvNEt idea into the GANs. This idea of incorporating ConvNEts into GANs make this DCGANs as the most eligible candidate for implementing unsupervised learning.

Many attempts were made to integrate the CNN with GANs to improve the performance. The approach used by Radford et al. [18] uses a family of architecture to train the model for a large number of the datasets and allow the training for higher resolution and deeper networks. The DCGANs [18] were implemented by the following three approaches:

- The concept given by Springenberg et al. [19] that replaces the idea of the max-pooling by the strided convolutions so that the network learnt from its downsampling is used as the first step in implementing the DCGANs.
- The second step is to eliminate all connected layers on top of convolutional features. It is applied by Mordvintsev et al. [20], where the concept of global average pooling is applied for the image classification models. This idea of the global average pooling gives ample stability to the system model.
- The third and last step is to apply the concept of the batch normalization 21. It helps to stabilize the learning process by normalizing the assigning each unit zero mean and unit variance. The process of stabilization solves the training issues in those problems due to poor initialization and helps gradient flow in deeper models. In this way, this process helps the generators begin learning and prevents them from collapsing at a single point.

Fig. 5.12 demonstrates the implementation of the DCGANs in detail. A 100-dimensional uniform distribution Z is projected to a small spatial extent convolution representation with many feature maps. Then this high-level representation is converted to a 64×64-pixel image with the help of the series of four fractionally strided convolutions. There is no use of the fully connected layers in this figure. In another work, Durall et al. [22] discussed a method to handle the problem of the stabilization that occurs in the training

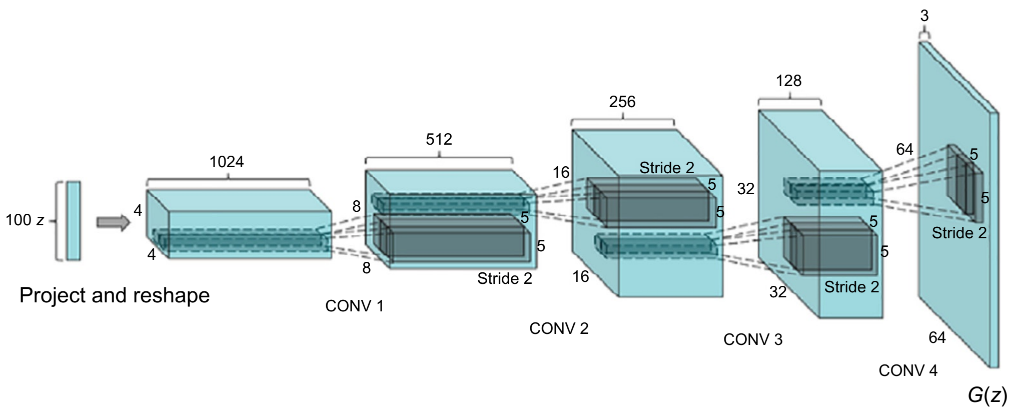

Fig. 5.12 DCGAN generator used for LSUN scene modeling [23].

phase of the GANs. A new framework called OC-GAN (Octave-GAN) that uses octave convolution is proposed in this work. It reduces the problem of modal collapse in the existing GANs and generates images of higher quality. The method is tested on the Celeb-A dataset.

5.2.5 StackGANs

A StackGAN consists of the two stacks which are considered as stage-1 and stage-2. The function of the stage-1 GAN is to produce the low-resolution images based on the description given by the user. Such images have very rough sketches and basic colors to give a preview of the low-resolution images. After generating these images from stage-1, these images are passed into the stage-2, in which high-resolution images are generated by these images which appears more realistic.

The process of the image generation is achieved by describing the form of the text or text embedding in the instructions. The stage-2 network adds all kinds of the relevant details as per the text instructions and thus produces images that are very close to the realistic images with proper resolutions. The working of the StackGAN can be compared with that of a painter. In the case of the complex painting, a painter always first draws some edges, rough sketches, and lines, etc. to prepare the overview of the image. In the next stage, the painter fills all relevant colors, adds more specific details, and shapes this artwork. Thus, it is in the second stage the painter gives a realistic view to his pictures.

Similarly, stage-1 produces the low-resolution images with the help of the given text description, and stage-2 that works on stage-1 tries to capture the details which are erased by stage-1. Stage-2 adds more information to the images generated by stage-1. The support of model distribution generated from a roughly aligned low-resolution image has a better probability of intersecting with the support of image distribution [24].

Fig. 5.13 depicts the architecture of the StackGAN. As discussed earlier, it is composed of the two stages, and for each step, there are two generators and two discriminators.

The StackGAN at each level consists of the text encoder, conditioning augmentation network, generator network, discriminator network, and embedding compressor network. As is very clear from Fig. 5.13 that stage-1 GAN is generating the images of low resolution with the size of 64×64 and then stage-2 GAN takes these images as inputs, applies some conditional augmentation on them to generate the high-resolution images of the size 256×256.

Fig. 5.14 displays the example of the images generated by StackGANs [24] with the help of the text description given to the system. The StackGAN is applied to the dataset of Oxford [25] for the generation of flower images.

Fig. 5.15 displays the example of the images generated by StackGANs [24] with the help of the text description about the rooms on the COCO dataset [26].

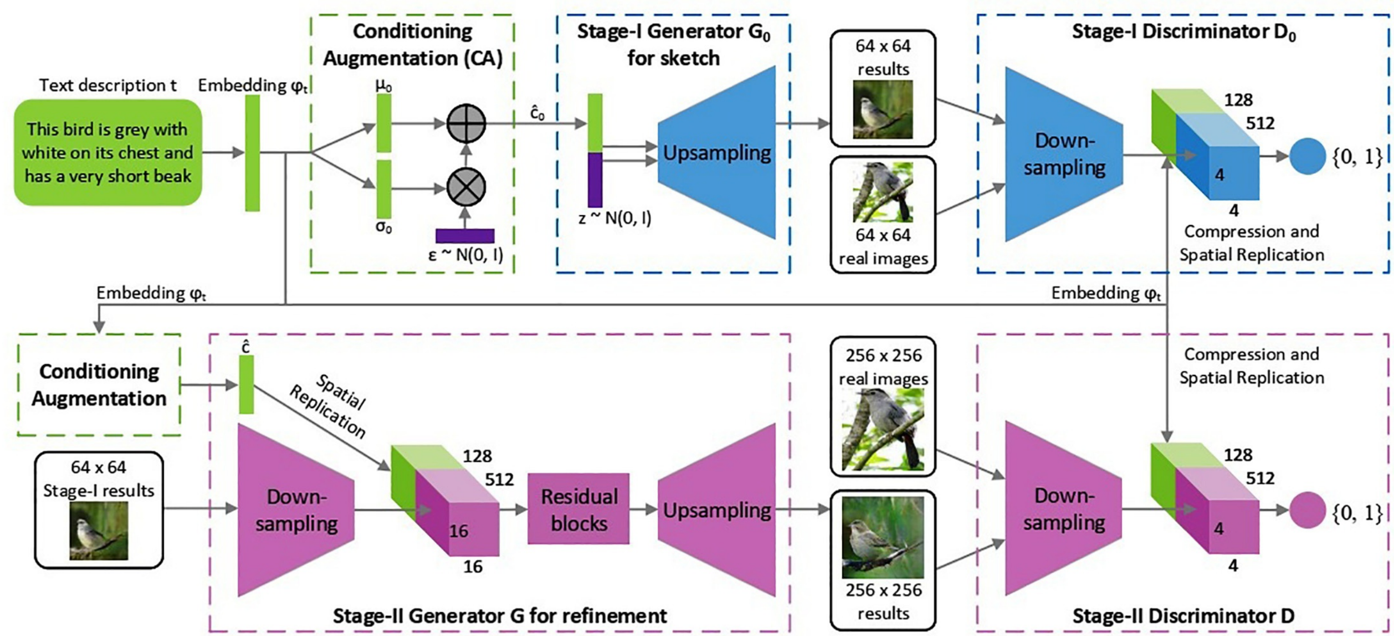

Fig. 5.13 StackGAN architecture: Stage-1 take inputs from the given text and applies rough sketching to produce low-resolution images by sketching a rough shape. Then, stage-2 generates more prominent high-resolution images by correcting the defects [24].

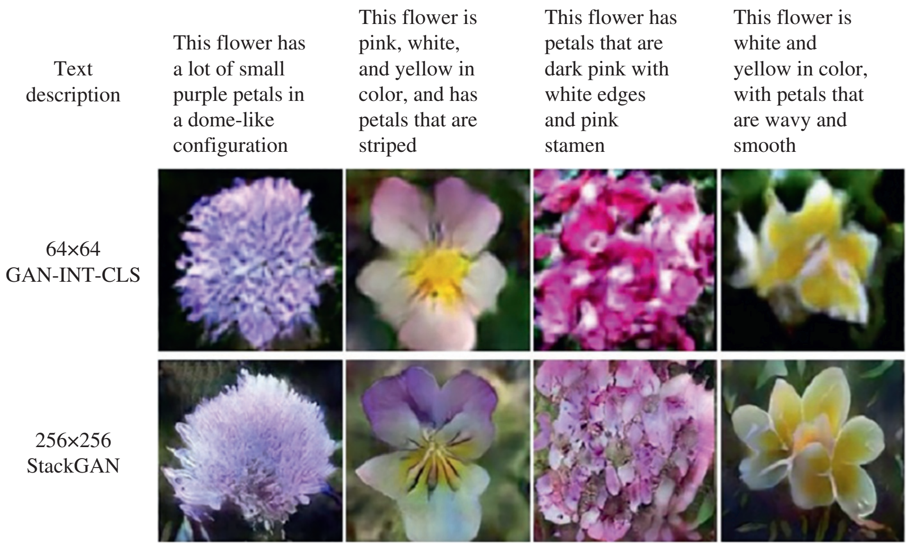

Fig. 5.14 Text-to-image generation using StackGAN [24].

Fig. 5.15 Results on COCO dataset [26] using StackGAN [24].

Thus, the method [24] performs much better concerning the other methods in this domain and produces high-resolution images that are incredibly close to the realistic images.

5.2.6 CycleGANs

CycleGAN [27] is one of the models used for training the image-to-image translation, in which the GAN architecture is used. It is an enhancement of the GAN model that simultaneously trains two generator and discriminator models.

In this model, two domains of images are formulated. The CycleGAN [28] in a simplified manner is shown in Figs. 5.16 and 5.17. The first generator is fed the images from the first domain, and the output of the first generator serves as input for the second domain of images. In contrast, the second generator takes input from the second domain of the images and outputs the images that feed as input to the first domain of images. The discriminator model checks how believable the images are from both the generators and then it fine tunes the generator models accordingly.

The model described earlier can check the correctness of the images generated for each domain, but it is not sufficient to translate the images. Therefore, for the purpose of the image-to-image translation, the CycleGAN has an add-on extension with the name of cycle consistency. In this, the output of the first generator is attached to the second generator's input. The output thus generated by the second generator is matched with the initial image fed to the first generator. Likewise, the reverse operation also holds true wherein the second generator's output can serve as input to the first generator, and the result produced is the same as the input fed to the second generator. The cycle

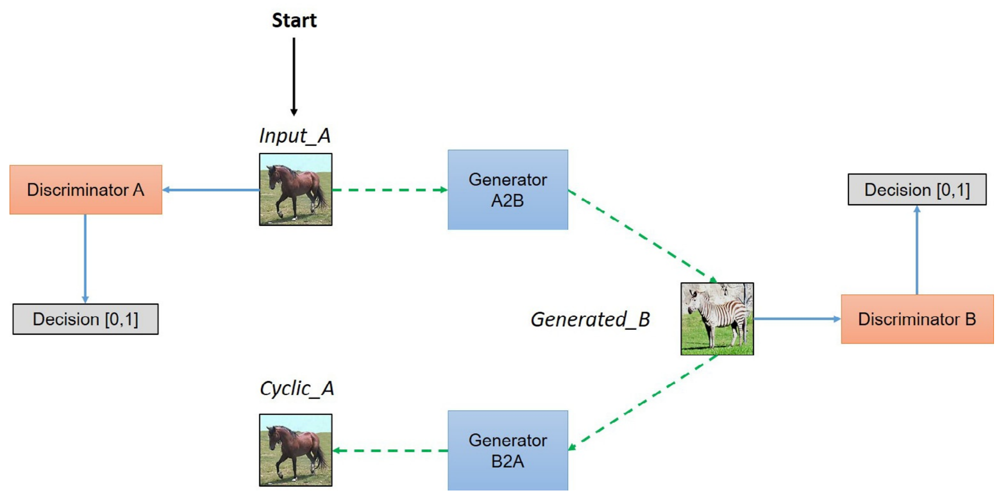

Fig. 5.16 Flow A-B-A starts from input in domain A [28].

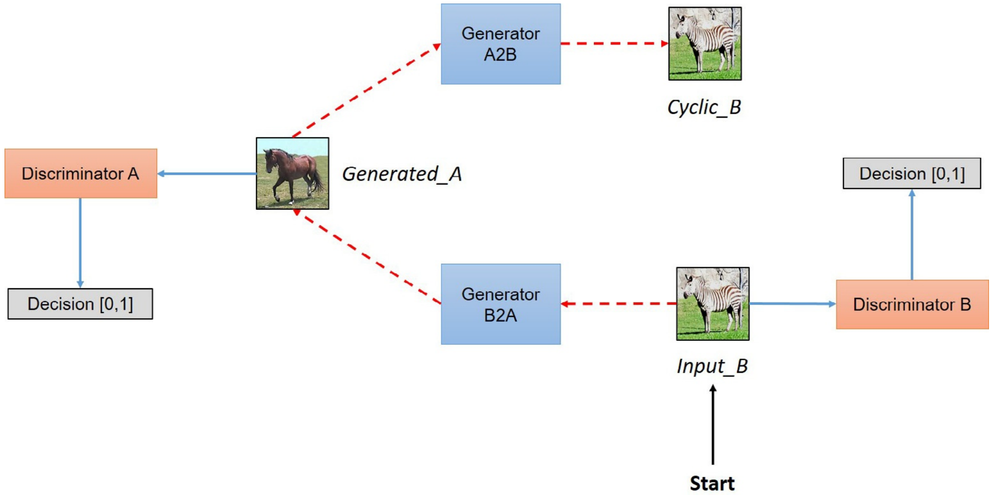

Fig. 5.17 Flow B-A-B starts from input in domain B [28].

consistency is used as a regularization measure for the generator models that help the image generation process for the image-to-image translation.

The CycleGAN model can be explained with the help of an example, where the aim is to translate the images of winter scene landscapes to the images of summer scene landscapes. It is well known that both the seasons will have different images for both the landscapes. So, in this case, the images of the two domains will be the images of the winter scene landscape, and others will be the images of summer scene landscapes, which is depicted through Fig. 5.18.

CycleGAN has an architecture of two GANs, and each GAN has a discriminator and a generator model, meaning there are four models in total in the architecture. So the system has two GAN generators; one will be taking images of winter scene landscape and generating images of summer scene landscape while the other will be taking the images of the summer scene landscape and generating the images of the winter scene landscape. Then the discriminator model will be checking if both the models are generating images as intended or not; based upon the discriminator's judgment, the generators will be further trained to get the exact translation.

The CycleGANs can be used in varied domains such as for style transfer, object transfiguration, season transfer, generating photographs from paintings, or photography enhancement.

5.2.7 Wasserstein GANs

The idea of the WGANs or the Wasserstein GAN was given by Arjovsky et al. [30]. It can be described as an augmentation to the existing GAN architecture. The main aim of the

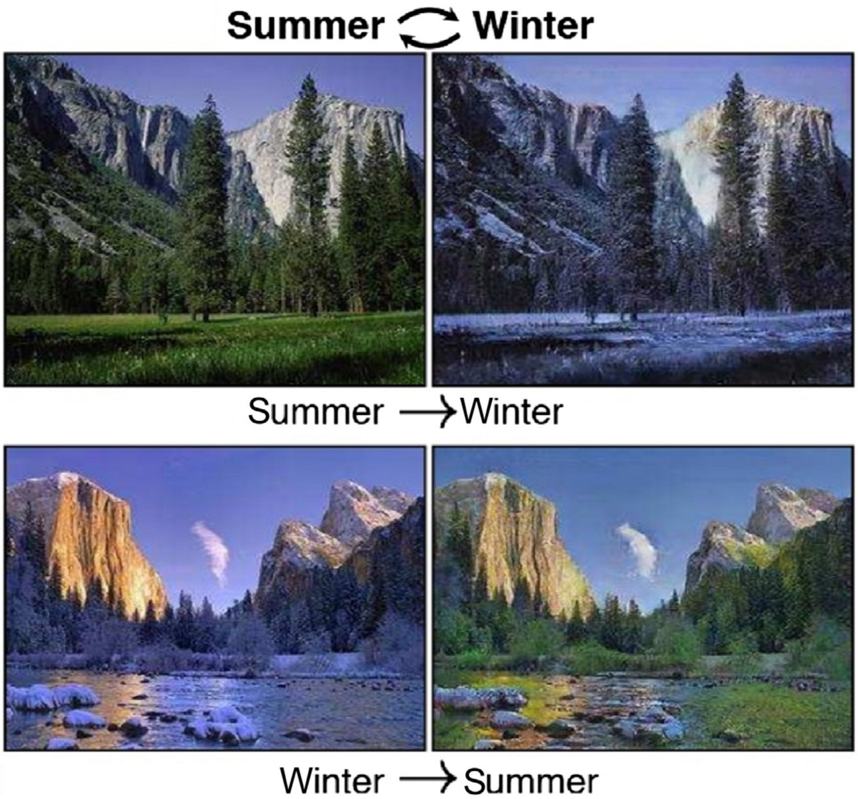

Fig. 5.18 Example of CycleGAN for summer to winter translation [29].

WGANs is to provide support for the model to improve the stability for the training of the given model and also provides a loss function to analyze the standards of the images generated by the model.

The WGAN uses an approach to perform a better approximation of the data provided in the dataset for training purposes. The WGAN proposes to use a critic in place of the discriminator, which decides the fakeness (or realness) of the given image with the help of the score given by that critic. The whole theory for the WGAN is based on the mathematical calculation about the distances. It states that the generator must search for minimization of the distance between the distribution of the data observed in the training dataset and the distribution observed in generated examples.

In the paper by Arjovsky et al. 30, the discussion consists of the various distribution distance measures, such as Jensen-Shannon (JS) divergence [31], Kullback-Leibler (KL) divergence [32], and the Wasserstein distance (Earth-Mover [EM] distance). The ability of each distance is based on the convergence of sequences of probability distributions. So, it was proved that WGAN could effectively train the generator using the properties of the Wasserstein distance more optimally compared to the other distribution distances.

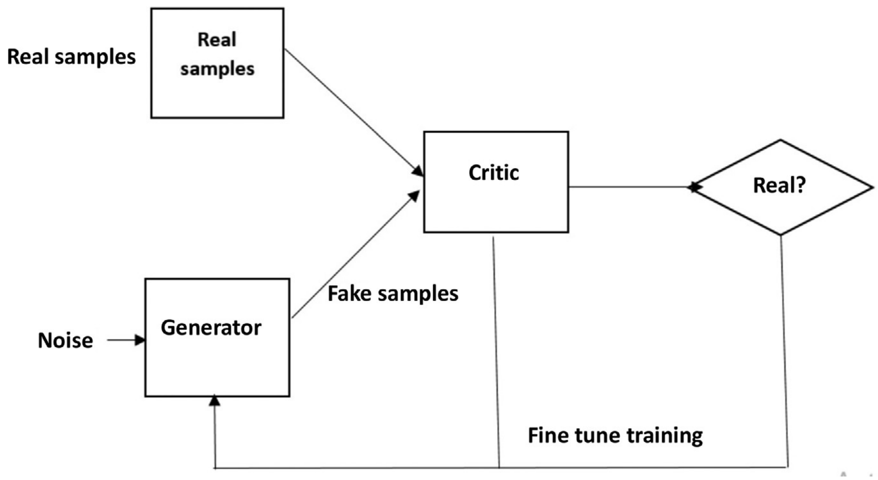

Fig. 5.19 A simple WGAN architecture [34].

Fig. 5.19 depicts the simple WGANs architecture and the concept of the WGANs revolves around the fact that Wasserstein distance is differential and continuous, which means that the training can be performed till it achieves optimal value. It is based on the argument that the longer the process of the training is carried out for the critic, then it will provide the reliable gradient of the Wasserstein. It happens due to the differentiable nature of the Wasserstein distance. Whereas in the case of the JS divergence (distance), the critic becomes reliable, but the true gradient becomes zero as the JS is locally saturated, and vanishing gradients are obtained. The critic in the WGANs does not get saturated. The critic gives a clean gradient as it reduces to the linear function compared to the discriminator that may learn the difference between real and fake quickly. Still, it does with almost no reliable linear-gradient information. The most crucial advantage of using the WGAN is that it makes the training process stable and ensures that the training process is not sensitive to the choice of the hyperparameter configurations. The WGAN aims to decrease the critic's loss, and achieving this can have a good quality of the generated images. The WGANs always try to make the lower the generating loss, whereas the other GANs try to achieve an equilibrium between two generative and discriminate models. The applications of the WGANs include the simulation of the isolated electromagnetic showers in a realistic setup of a multilayer sampling calorimeter [33]. Similarly, one critical step in the analysis of the medical images is the structure-preserved denoising of 3D magnetic resonance imaging (MRI) images.

Ran et al. [35] presented the use of the Wasserstein generative adversarial network (RED-WGAN) for the MRI denoising method.

The next section discusses some of the open issues and research gaps in the domain of the applications related to the GANs.

5.3 Discussion on research gaps

There are many open problems and research gaps where the GANs can be applied to achieve better results as compared to the traditional machine learning approach. The work by Barua et al. [6] emphasizes using the fully connected GANs for the unsupervised training. So, the effect of using fully connected convoultional (FCC) GANs on semisupervised training can be studied. Similarly, the inclusion of the CGANs can improvise the results. The complex networks, such ResNet [11], can be examined with the help of the FCC-GANs.

Zhao et al. [36] propose a method using adversarially regularized autoencoders for training deep latent variable models on the simple discrete structures. These simple structures consist of short sentences and the binary digits. Therefore, the scope is available to apply the training to model complex structures such as documents and old manuscripts.

Balabka [37] proposes a model using AAEs to recognize human activity with the help of semisupervised learning. The semisupervised learning is applied to use the unlabeled data with the help of the AAE training. The challenge open in the domain of the AAEs is to explore many hyperparameter that can be tuned to improve the performance of the model aggressively.

Ruiz-Garcia et al. [38] suggested a new method that uses a generative adversarial stacked autoencoder, which helps in mapping the facial expressions to an illumination invariant facial representation. The open research problem in this domain includes developing a method that can handle the scenario in which there is no labeling of the multipose datasets.

Lu et al. [39] discuss a method based on deep learning-based (DA-DCGAN) for practical domain-shifting DC series arc fault detection in photovoltaic systems. The GANs serve the purpose of the generation of the dummy arc shifting data. But this problem of implementation of the GANs in the area of the application-specific integrated circuits with low cost and improvement in the reliability remains open.

Padala et al. [40] proposed the idea to study the effect of variation of the input noise applied to the GANs. The inference of the method states that the noise has a remarkable contribution to the images' generation. But the gap in this study is to make a theoretical analysis between the high-dimensional data and low-dimensional distribution.

Kim [41] proposes a new variation of the GANs called Bool GAN and applied on the dataset containing the images of cars, that put the baseline model proposed by Radford et al. [18]. The inclusion of the dropout and convolution layers improves the efficiency of the model. The open issue in this study is to perform more experiments regarding the addition of a number of layers to find out the optimum hyperparameter and the scheduling of the learning rate. This study may give new dimensions about the performance of the model.

Durall et al. [22] proposed the use of Octave GANs, which states that Bayesian optimization can be explored as future work.

Cheng et al. [42] presented a novel method called SeqAttnGAN for creating images with the help of interactive image editing software. This method is implemented on the two benchmark datasets, DeepFashion-Seq and Zap-Seq. These datasets have images that are attached to the proper description in the text. The method gives excellent results as compared to the baseline methods. This method addresses future work to create human faces with an interactive image editor and to explore the generation of consistent image sequences by given attributes and other factors.

Vougioukas et al. [43] proposed a novel and innovative method to generate the video signals generated by speech. The method is achieved by applying temporal GANs. The performance of the method is evaluated on the GRID [44] and the TCD TIMIT [45] dataset. The method can capture the videos with proper facial expressions, including blinking, etc. This method is open to the problem of capturing the possible mood and gesture of the speaker and showing it using the facial expression.

Zhao et al. [46] suggested a novel idea dependent on the CGANs to retrieve the lost and missing information from the images of the solar observation. This missing information occurs due to the overexposure of the images in the solar observation process due to the violet solar burst. This novel idea uses CGANs and includes integration of the edge mass loss, masked L1 loss, and adversarial loss. The model uses training for the new dataset for the overexposed images. The work is still open to the problem where the images are of high texture and have large overexposure areas.

Zhu et al. [47] proposed a novel method by implementing the CGANs to solve the issue of producing the multiple outputs of the image-to-image translation using a single input. The mapping ambiguity is resolved by randomly sampling the low-dimensional latent vector. The generator used in this method achieves to map the input, with latent code to the possible output. Thus, the method encourages the bijective consistency between output modes and latent encoding. The future work in this method caters to produce the image-to-image translation with the help of controlling different user parameters along with meaningful attributes.

Nataraj et al. [48] discussed a model for detecting the fake images generated by the GANs. It is achieved by using the combination of deep learning and cooccurrence matrices of the pixels. These types of matrices are obtained by performing the computation on the different color channels. Further, a deep convolution neural network is used for training purposes to discriminate the real images from the fake images generated by the GAN. The future work in this domain is to make use of the pixels' location manipulated in the fake images generated by GANs and rectify them.

Liu et al. [49] used the idea of the Coupled GANs to perform the image-to-image translation in the unsupervised environment. The aim is to use the information about the images in two different domains and perform translation. The open issue that need to be addressed in this is to prevent the stability of the training system due to the saddle point searching problem. Along with that, the issue is to remove the system's limitation as unimodal due to the assumption of the Gaussian latent space.

The other open problems in this domain are like the need for the automatic metrics for judging the performance of the different types of generative networks and also need to consider the nondeterministic training losses for future prediction.

The next section discusses some of the applications that can solved with the assistance of the GANs.

5.4 GAN applications

There are many areas in which GANs can be applied to have remarkable results, and these are as follows:

- *Generation of images*: The task of the generation is one of the prominent areas where GANs are applied and giving the researchers an ample amount of the datasets to carry out different experiments. These images are realistic in nature. The process of image generation includes some sample images, based on which the GAN can generate a large number of the images with the help of the generator and discriminator. These new images which are generated will be different as compared to the existing sample images. The process of the generation of the images is used extensively in the animation, social media, marketing, entertainment world, and generation of the logos of the digital world.
- *Synthesis of images using text*: The most exciting feature of the GANs is the synthesis of the images with the help of the text description. Such applications are used in the entertainment industry. With the help of the text (story), an animation character with their gestures can be created.
- *Aging of face*: The GANs with their variants called the CGANs can be used to predict the face of the images with the targeted ages. GANs architecture can create and predict people's faces at different times (age). Thus, such a system can be useful in companies for face verification of their employees. It works on the principle of semisupervised learning for aging and progression. There are datasets with faces as images and age as labels are available in the public domain for experiment purpose.
- *Image-to-image translation*: This feature of GANs states that the images can be translated into other images with the help of the generator and discriminator. The images taken at night can be translated into the day; similarly, the different drawings and sketches can be translated into beautiful paintings. The different aerial images can be translated into the satellite images, and the images of the zebra can be converted to horses, etc. CGANs can be applied for synthesizing photos from label maps as defined by Nayak [14], uses edge maps to create, and fill colors in the images as discussed by Wang et al. [50] and Isola et al. [51].

- *Synthesis of video*: The synthesis of the video can also be performed with the assistance of the GANs. They take less time to create the videos compared to the conditions if they are created manually or in real time. In this manner, this property of GANs can encourage the animation creators to make optimum use of the new technology to develop and promote their videos in less time and close to the real world. GANs can also be used to predict the frames to appear in the future in any video sequence as mentioned by Villegas et al. [52].
- *Generating high-quality images*: The GANs allow converting the low-resolution images taken by ordinary cameras to high-resolution and quality images. Thus, it helps to observe the minute details of the images that cannot be viewed in low-resolution images.
- *Missing part generation of images*: The GANs network can be used to generate the missing parts of the partially degraded images and thus recovers the original images.
- *Generating shadow maps*: Nguyen et al. [53] apply one conditional (sensitivity) parameter (CGANs) to the system (generator) to parameterize the loss of trained detector and is more efficient than other GANs.
- *Speech enhancement*: Phan et al. [54] propose two architectures ISEGAN and DSEGAN for the process of speech enhancement. The main motive behind speech enhancement is to remove the unnecessary background and irrelevant noises that create problems in the process of speech recognition. The speech enhancement will further help in the cochlear implants, hearing aids, and communication systems. Therefore, GANs play an important role in the domain of speech recognition by enhancing the sample of speeches.
- *Fault diagnosis*: The GANs system can be implemented in the detection and diagnosis of the DC arc faults that occur in the photovoltaic system as described in Lu et al. [39]. The source and the target domain data are available during the operation in the field, but the fault data are not available. Therefore, GANs can be used to generate the dummy data.

5.5 Conclusion

This chapter mainly covers the introduction to the GANs, its need, and the detailed architecture of the various models, that is, fully connected GAN, CGANs, and AAEs, deep convolution GANs, StackGANs, CycleGANs, and Wasserstein GANs. The advantages and disadvantages of the models are listed. The chapter also focuses on the various research gaps identified in the different architecture of the GANs. The research gaps identified will provoke the students and scholars in this domain to contribute to the development of algorithms in the GANs. Various applications fall into the area of GANs. These are listed in the last section of the chapter. These applications, if solved using the approaches of the GANs, will provide better results as compared to the traditional machine learning approaches. The chapter also covers the various examples of the image-to-image translations described by the researchers.

In the recent years, GANs have been emerged as one of the novel methodologies to generate the data from the rough information given. They are considered to be the robust and powerful class of the neural networks for unsupervised learning. With the GANs idea, a large number of the image dataset can be created, which are very close to the real image. Thus, it satisfies the need of the dataset among the researchers for implementing their models.

Along with the great advantage of generating huge images, GANs have limitations that they can generate better results if the input data is mapped into the learned subspace. Still, in case of the unseen data not mapped correctly, it may give poor results. Another problem associated with the GANs is the problem of mode collapse, which states that the generator always produces output from a small set of spaces. Similarly, the GANs in certain stages are challenging to converge in the training process. The machine and the resources required to implement the GANs training models have exceptionally high configuration and expensive. The GANs training and implementation process require extensive use of the GPUs along with the CPUs. The need for memory for accounting the large data is also an issue in the GANs. The researchers in this domain can work on the complex problems of artificial intelligence by implementing the advanced version of the GANs. It will enhance the capabilities of the machines and provides the human race with a new solution to existing problems in the different areas of science and engineering.

References

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

[2] A. Mittal, Generative Adversarial Networks (GAN), 2020. https://codeburst.io/generative-adversarial-networks-gan-3c8978ba99a6.

[3] J. Hui, GAN some cool applications of GAN, Medium (2020). https://medium.com/@jonathan_hui/gan-some-cool-applications-of-gans-4c9ecca35900.

[4] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, A.C. Courville, Improved training of Wasserstein GANs, in: Advances in Neural Information Processing Systems, 2017, pp. 5767–5777.

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

[6] S. Barua, S. Monazam Erfani, J. Bailey, FCC-GAN: a fully connected and convolutional net architecture for GANs, arXiv E-Prints, arXiv-1905 (2019).

[7] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, Proc. IEEE 86 (11) (1998) 2278–2324.

[8] A. Krizhevsky, Learning Multiple Layers of Features From Tiny Images (Master's thesis), Department of Computer Science, University of Toronto, 2009.

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