# A review on Generative Learning in Computer Vision

Pathan Faisal Khan
Letterkenny Institute of Technology
Port Road, Letterkenny
Co. Donegal, Ireland
L00151142@student.lyit.ie

***Abstract*—Generative learning is a technique to generate artificial data. This paper has covered a brief review of different generative learning techniques applied in computer vision to generate visual data. With the advancement in computing power especially the utilisation of GPU for distributed computing, a lot of deep learning techniques has been developed in this field of computer vision. Generative Adversarial Networks and autoencoders are such unsupervised deep learning neural networks. GANs have garnered a lot of interest by computer scientists due to its high-quality image generation and also because of its novel yet simple architecture. A lot of variants of GANs are have been developed since the original GAN was introduced by Goodfellow et al in 2014 [1]. These variants have either tweak or have built upon the existing model. Some of the techniques covered in this paper are conditional GAN, progressive GAN, deep convolutional GANs, auxiliary GAN, InfoGANs, 3D-GAN, PacGAN, Pix2PixGAN, Cycle-GAN, Text-to-Image Synthesis (StackGAN), and Super-Resolution GAN. These models are just a few among others presented by researchers. This paper has also provided a brief critique of Variational Autoencoders which is a variant of the autoencoder neural network.**

***Index Terms*—generative learning, GAN, autoencoder, computer vision, conditional GAN, progressive GAN, deep convolutional GANs, auxiliary GAN, InfoGANs, 3D-GAN, PacGAN, Pix2PixGAN, Cycle-GAN, Text-to-Image Synthesis (StackGAN), Super-Resolution GAN, Variational Autoencoders**

## I. Introduction

Artificial intelligence has made it possible for computers to learn from experiences and perform simple tasks which a human can easily do. Visual perception by human-like object recognizing abilities is once such task which scientists have been able to teach computers. In the past few years due to advancement and innovation in computing power especially unlocking GPU based distributed computing, computer scientists have achieved success in able to teach computers to do complex tasks beyond classification like object detection, image segmentation, object tracking, and event detection. Computer Vision (CV) has since been used in numerous ways ranging from automated cars to quality check at factories which usually required an expert to manually check the production line.

With such advancements, the need for data is never ending. CV models running on neural networks require huge amount labeled training data to train them. But the problem lies in finding suitable high-quality data. Manual scavenging and labelling of data is not an ideal approach as it is costly to do so. The only option left for computer scientists is to produce high quality artificial data either from scratch or by manipulating existing data. Generative learning is one such popular way to generate artificial data. This paper will look at different generative methods introduced lately.

The rest of the paper is organized as follows– Section II will provide an overview of generative learning. Section III will cover recent techniques used for generative learning. Finally, section IV will provide some remarks to conclude the paper.

## II. Generative Learning

### *A. Background*

Computer scientists have contributed a lot of research towards generating synthetic visual data. With such techniques, computer scientists have been able to generate data which is almost indistinguishable by a human eye. This fast availability of generated data has made it possible to solve a lot of modern problems in deep learning.

There have been some generative techniques around in this field of deep learning. In 2014, Goodfellow et al. published a paper on Generative Adversarial Networks (GANs) [1]. This state-of-the-art technique proved to be a major breakthrough in deep learning, especially in the CV. GANs has since then been applied in numerous fields including Natural Language Processing (NLP) and computer vision. This family of deep learning methods has become quite popular due to its good results. This review paper provides a survey on recent adaptations of GAN along with the first-original version often called as vanilla GAN. Autoencoders is a class of unsupervised neural networks which is also a popular method to generate data with some applications dating back to the 80s [2], [3]. Variational Autoencoders (VAEs) in specific is a generative technique.

### *B. Taxonomy*

The following Table I presents taxonomy of different generative learning techniques–

## III. Techniques

### *A. Generative Adversarial Networks (GANs)*

Goodfellow et al. developed this state-of-the-art class of neural networks with two key components; a generator and a discriminator. The generator is a neural network which

TABLE I
TAXONOMY AND CITATIONS OF GENERATIVE LEARNING TECHNIQUES

| Technique | | Citations |
|---|---|---|
| Generative Adversarial Networks | Conditional GAN | [4] |
| | Progressive GAN | [12] |
| | Deep Convolutional GANs (DCGAN) | [5] |
| | Auxiliary Classifier GAN | [7] |
| | Info GANs | [6] |
| | 3D-GAN | [8] |
| | PacGAN | [9] |
| | Pix2PixGAN | [10] |
| | Cycle-GAN | [11] |
| | Text-to-Image Synthesis (StackGAN) | [13] |
| | Super-Resolution GAN | [14] |
| Autoencoders | Variational Autoencoders (VAEs) | [15] |

takes a random vector input usually a noise and generates a new plausible output out of it in the domain. This vector input is taken from a Gaussian distribution. A Gaussian or normal distribution is a probability distribution around a mean, which signifies the data around the mean is more frequent as compared to data points away from the mean. A Gaussian distribution shows a bell curve when plotted. The vector is referred to as a latent space which means significant variables but not explicitly visible. Since it uses a random vector sample every time, the model becomes stochastic, the output will be non-deterministic every time. The generator makes unlimited samples out of it. The discriminator, a classification model, then takes input from the training dataset as well as output from the generator model and attempts to identify the source of the input, whether it is from the training dataset or the generator generated output. Both models are meant to compete with each other, the generator has to fool the discriminator into incorrectly classifying its sample as a real sample from the training set and the discriminator has to make itself perfect incorrectly classifying input samples. The generator and the discriminator are both trained along with. When the discriminator predicts the class of the input sample, it is tweaked to perform better for the next sample, likewise, the generator is updated to create samples which will fool the discriminator. Even though GAN is an unsupervised learning technique but this architecture of it is designed as a supervised learning technique. The generator model is then used to discarding the discriminator model. The following figure 1 describes the working of a GAN.

*1) Conditional GAN:* Conditional GAN is an extension of GAN proposed by Mirza et al. in 2014 [4]. It works on the principal of conditional probability which is defined as the probability of an event given another event mathematically represented as $P(A|B)$ which translates to probability of $A$ given $B$. Conditional GAN is conditioned using extra information which is passed to the generator as well as to the discriminator. This extra information can be any supplementary information like image labels. The discriminator gets the extra information alongwith the input from generator. One of the applications of conditional GAN is generating image of a person given its gender. The following figure 2 describes the working of a conditional GAN.

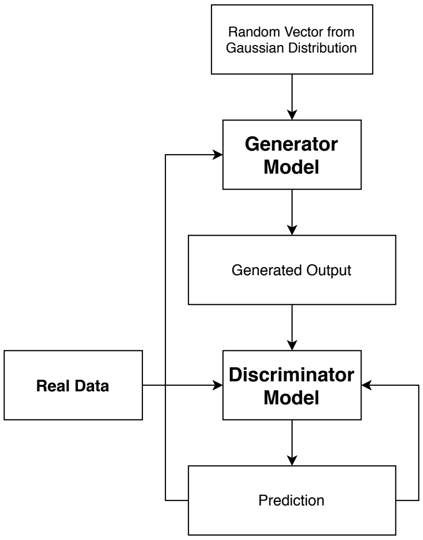

Fig. 1. End-to-end process of the Vanilla GAN

*2) Progressive GAN:* Karras et al in 2017 published a paper introducing progressive GAN [12]. According to the authors of the paper, they proposed a novel training technique where the generator and discriminator grow progressively as they train themselves. Growing progressively means adding more layers to themselves to fine tune the model. With this approach, the training time decreases. The generator produces low quality images and as more layers are added it fine tunes the model resulting in high quality images. This progressive learning technique allows the model to gradually discover large scale composition of the image and then focus on improving the fine details of the image. New layers are added in such a way that it does not drastically affect the fine tuned network. Figure 3 shows how layers are added to both the generator and discriminator together.

*3) Deep Convolutional GAN (DCGAN):* Deep Convolutional GAN (DCGAN), one of the first convolutional GANs proposed by Radford et al in early 2016 [5], relies on the

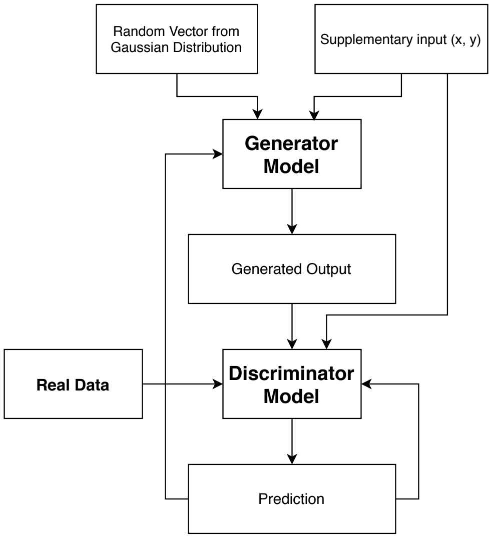

Fig. 2. End-to-end process of the conditional GAN

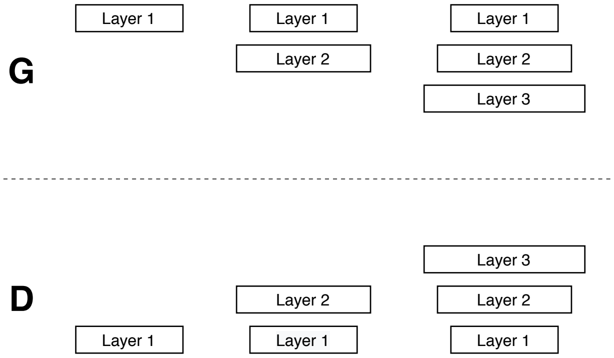

Fig. 3. Addition of layers in every iteration of progressive GAN

architecture of CNN. The authors proposed a few architecture guidelines which describe the composition of a DCGAN. They have taken a CNN and replaced the pooling layers with convolutions with fractional-strides in the generator to reduce feature maps resulting from the sample Gaussian input. They also have replaced the pooling layers in discriminator with convolutions with strides resulting in more parameters to train the model. They then used batch normalization for both the models resulting in stable training. Then to lower the training parameters, they removed hidden layers. Then finally they have used ReLU and LeakyReLU as the activation function in the generator and the discriminator respectively.

*4) Auxiliary Classifier GAN:* Auxiliary classifier GAN (ACGAN), proposed by Odena et al in 2017 [7], is able to generate high-resolution images using a technique called label conditioning. At that time, the experiment resulted in images of 128 x 128 resolution images. They also were able to identify the collapsing behavior of a GAN. The model the presented is a conditional model which was able to take a lot of lables/classes as supplementary input. They were able to create highly different sample spaces at a time. Unlike CGAN, the discriminator, in this case, learnt the supplementary information on its own, this technique is referred to as reconstruction loss.

*5) 3D-GAN:* 3D-GAN, is a 3D-object generation GAN proposed by Wu et al [8] in 2016. This was a novel technique which focused on the 3D space generation using GAN using CNN. 3D object generation is not an easy task due to the high-dimensionality of the object. With high dimensionality, a large number of parameters are needed. To solve this high dimensionality problem, CNN is the best bet out there. A CNN with the apropriate configuration helps in increasing the dimensions of the sample space while also being able to reduce the parameters/features. The authors proposed a CNN which is similar to DCGAN. With this architecture they have managed to reduce the features by a great amount. They added a Sigmoid layer in the end. Due to the fact that it is difficult to generate a 3D object as compared to be able to classify them, the authors had to come up with a simple training method. They used a bigger batch with very low learning rate. This method proved to be able to generate good-quality shapes.

*6) InfoGAN:* InfoGAN was proposed by Chen et al in 2016 [6]. InfoGAN is a GAN which utilises the information shared among a subset of the latent space. It also uses an unsupervised learning technique called disentangled representation which allocated different sets of dimensions by breaking down features of the data. Disentangled information represents a low representation of the data. With the introduction of this technique, it makes the model truly unsupervised. InfoGAN, as claimed by the authors, contributes comparatively less computation cost when used with GAN also making it easier to train.

### *B. Autoencoders*

Autoencoders are a type of neural networks which learns in an unsupervised way. As the name suggests, it encodes the input and then tries to reconstruct the input to generate similar or better output. This way it learns to reduce unnecessary features while being able to produce the same input data it was fed. This technique is often used for dimension reduction, fix noisy data, or compression without losing significant parts of the data. Autoencoders generally use a feedforward architecture which uses backpropagation while training.

In the above figure 4, an autoencoder is shown. It has 2 main components: an encoder which takes in the input and encodes it and a decoder which uses the encoded output of the encoder as an input and tries to generate the original input of the model. Due to the backpropagation nature of this model, the training is quite fast.

*1) Variational Autoencoders (VAEs):* Variational Autoencoders (VAEs) are generative models. It is based on vanilla autoencoders but with a distinct property of picking up input from a continuous latent space. It avoids overfitting by regularizing its training. Unlike GAN, where a discriminator is used to classify the data which in turn helps the generator to tune itself to fool the discriminator, VAEs learn how to generate the

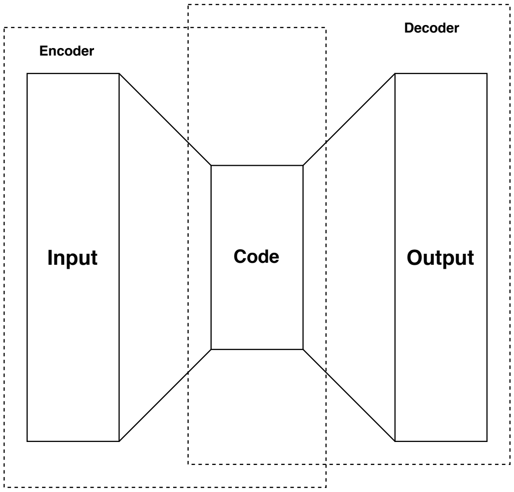

Fig. 4. Architecture of an autoencoder

data. The model first encodes the input over the latent space as distribution, then it picks up a point from the resultant latent space and then decoded. The error in decoding is calculated and backpropagated in the network. It uses Stochastic Gradient descent to train itself. This enables the model to generate distinct samples with similar characteristics to the input. VAEs can work with a surprisingly diverse data forms, continuous or discrete, labelled or unlabelled, sequential or non-sequential, making it an extremely efficient generative learning technique. VAEs has proven to produce images out of latent space but it often performs poorly. This is the reason why VAEs are not used anymore for generative learning in computer vision.

## IV. Conclusion

This review paper has covered a brief review on generative learning in computer vision focusing more towards GANs. It has been noted that GANs was introduced as a basic architecture and computer scientists have built upon different variants of GANs on this original GAN. Some GANs performed the same thing but with better accuracy as compared to others with novel architectural changes like DCGAN and Conditional GAN, while some GANs did different tasks like 3D-GAN and StackGAN. These variants of GANs has proved to be a starting point for other GANs like 3D-GAN is based on the architecture of DCGAN. The paper has also shown the inefficient generative method being used before GANs were introduced, Variational Autoencoders. A brief overview of autoencoders and how VAEs are different from autoencoders was also presented. The generative methods presented in this paper are not exhaustive, more methods have been presented and proposed by researchers in this field. The paper has covered some popular methods used in the CV. Currently, GANs have proved to be a major break-through in generative learning, it is not hard to say that GANs will be around for a while and many such researchers will not stop innovating over this model.

## References

[1] Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y., 2014. Generative Adversarial Networks. arXiv:1406.2661 [cs, stat].

[2] Ballard, D.H., 1987. Modular learning in neural networks, in: Proceedings of the Sixth National Conference on Artificial Intelligence - Volume 1, AAAI'87. AAAI Press, Seattle, Washington, pp. 279–284.

[3] Rumelhart, D.E., Hinton, G.E., Williams, R.J., 1986. Learning internal representations by error propagation, in: Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Vol. 1: Foundations. MIT Press, Cambridge, MA, USA, pp. 318–362.

[4] Mirza, M., Osindero, S., 2014. Conditional Generative Adversarial Nets. arXiv:1411.1784 [cs, stat].

[5] Radford, A., Metz, L., Chintala, S., 2016. Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. arXiv:1511.06434 [cs].

[6] Chen, X., Duan, Y., Houthooft, R., Schulman, J., Sutskever, I., Abbeel, P., 2016. InfoGAN: Interpretable Representation Learning by Information Maximizing Generative Adversarial Nets. arXiv:1606.03657 [cs, stat].

[7] Odena, A., Olah, C., Shlens, J., 2017. Conditional Image Synthesis With Auxiliary Classifier GANs. arXiv:1610.09585 [cs, stat].

[8] Wu, J., Zhang, C., Xue, T., Freeman, W.T., Tenenbaum, J.B., 2016. Learning a Probabilistic Latent Space of Object Shapes via 3D Generative-Adversarial Modeling.

[9] Lin, Z., Khetan, A., Fanti, G., Oh, S., 2018. PacGAN: The power of two samples in generative adversarial networks, in: Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., Garnett, R. (Eds.), Advances in Neural Information Processing Systems 31. Curran Associates, Inc., pp. 1498–1507.

[10] Isola, P., Zhu, J.-Y., Zhou, T., Efros, A.A., 2018. Image-to-Image Translation with Conditional Adversarial Networks. arXiv:1611.07004 [cs].

[11] Zhu, J.-Y., Park, T., Isola, P., Efros, A.A., 2018. Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks. arXiv:1703.10593 [cs].

[12] Karras, T., Aila, T., Laine, S., Lehtinen, J., 2018. Progressive Growing of GANs for Improved Quality, Stability, and Variation. arXiv:1710.10196 [cs, stat].

[13] Zhang, H., Xu, T., Li, H., Zhang, S., Wang, X., Huang, X., Metaxas, D., 2017. StackGAN: Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks. arXiv:1612.03242 [cs, stat].

[14] Ledig, C., Theis, L., Huszar, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J., Wang, Z., Shi, W., 2017. Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network. arXiv:1609.04802 [cs, stat].

[15] Doersch, C., 2016. Tutorial on Variational Autoencoders. arXiv:1606.05908 [cs, stat].