**How to Cite**:
Kumari, S., & Aggarwal, K. (2022). Scope of generative adversarial networks (GANs) in image processing: A review. *International Journal of Health Sciences*, *6*(S6), 724–733. https://doi.org/10.53730/ijhs.v6nS6.9664

# Scope of generative adversarial networks (GANs) in image processing: A review

**Sudesh Kumari**
Maharishi Markandeshwar (Deemed to be University), Mullana, Ambala, India | BPS Women University, Khanpur Kalan, Sonepat, India
Email: sudesh_30@rediffmail.com

**Karan Aggarwal**
Maharishi Markandeshwar (Deemed to be University), Mullana, Ambala, India
Email: karan.170987@gmail.com

***Abstract*---**Generative Adversarial Network is the topic of interest in today's research in the field of image processing and computer vision. A basic GAN model was introduced by Ian Goodfellow et al. in 2014. After that advancement in the field of research in GAN models has been application specific. In computer vision and image to image translation GANs are playing very effective role either in the case of face detection and recognition or in image resolution enhancement and image augmentation. This paper represents a concise overview of various GAN models along with their features and applications. Pix2Pix and conditional GAN models work upon paired datasets while other models like cycle GAN, discover GAN, dual GAN, info GAN, deep convolutional GAN etc. work upon unpaired datasets. Various image datasets which are commonly used for training of generator and discriminator networks are also discussed in this paper. Since partial mode collapse is a common problem to occur during training process for all models, therefore various normalization techniques are also preferred during the training of generator and discriminator networks. As the advancements in GAN models are increasing at a very fast rate, soon these models will also be preferred in commercial applications.

***Keywords*---**image to image translation, style transfer, neural networks, image generation.

## Introduction

GANs are generative models and used to generate new data from existing one. GANs have two models which are adversarial to each other: Generative model and discriminative model. The prior is used to generate new data/samples from the


*Manuscript submitted: 18 Feb 2022, Manuscript revised: 27 April 2022, Accepted for publication: 9 June 2022*

existing dataset and random signal while the later is used to compare the new generated data with the real data/image. The generated data is real or fake; it is decided by the discriminator by using the value between 0 and 1. If this value lies between 0.5 or greater, the generated image is real otherwise the image is not acceptable. The whole process follows self-correction method by using back propagation in which the generated image is reproduced to minimize the rate of error [1-3].

**Generic workflow of GANs based algorithms**

- GANs are used for semi-supervised and unsupervised learning. In deep learning three learning techniques are used which are supervised learning, semi-supervised learning and unsupervised learning. Supervised Learning is a learning technique in which the machine uses well labelled dataset, the data which are available with the correct answers. Unsupervised Learning looks for earlier unobserved patterns in a data set with a least human involvement. Semi-supervised Learning works on combination of both types of data in which a small quantity of labelled data and a great quantity of unlabelled data is used for the duration of training [4-6].
- Discriminative Network uses convolutional neural network for image detection, in which the neuron in each layer is connected to a small section of its previous layer, rather than all neurons in a fully connected mode. It uses convolution function instead of common matrix multiplication in at least one layer [7-8].
- Generative Network use deconvolutional neural networks and these neural networks work in opposite order of convolution neural networks [9].
- Back propagation is applied to both Networks which are used to compute the contribution of each parameter on the error term. Thereafter, gradient descent is used to update these parameters such that the next pass through should result in a lower loss rate. Picking the right loss function is essential for this process [10].
- Normalization techniques are required to stabilize the training process of generator and discriminator networks, it also improves the performance of discriminator [11-12].

Fig.1 shows the basic workflow diagram of GAN, in which the generated image is passed through discriminator network to calculate the generative and discriminative loss. Due to GANs feature of parallel generation of high-quality images, which may also be trained in an imaginary environment, it finds application in the field of image processing as well as in computer vision [13]. Some common applications are in object detection and face recognition, image to image translation, video games, reconstruct 3D models of objects from images, architecture designs, used in maps (augment street view imagery), speech recognition [14-15], speech2face, automatic aging or de-aging of photographs [16], image resolution enhancement (focusing on realistic texture rather than pixel-accuracy) [17], augmentation of images [18], generating new anime character like Pokémon, generating new logo, clothing etc., translating a photo of summer to winter or day time tonight time, translate a sketch to photograph, automatic image in-painting, generate sequence of images, image to photo generation, photo to emojis generation and photo blending etc. [19-20]. GANs also have many applications in medical image processing [21-23] and in image forgery detection [24-25].

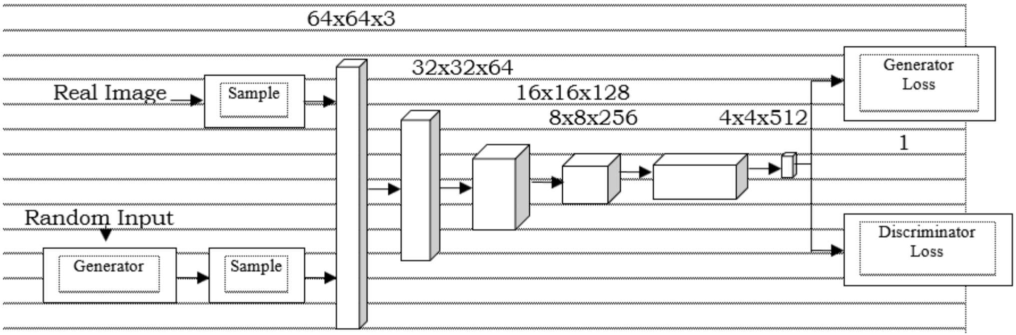


Fig. 1. Workflow diagram of Generative Adversarial Network

**Generative Techniques for Image Processing**

Generative algorithms used to generate data/images are classified in two groups: explicit density models and implicit density model. Explicit density models use true data to train the generator model for generating new data. These models include Markov chain methods, approximate inference and Maximum likelihood estimation etc. Implicit density models cannot directly estimate the data distribution and generate data samples from the given distribution.

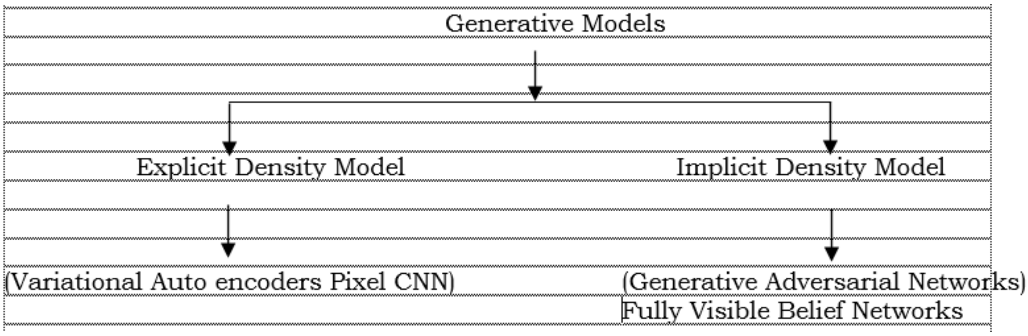


Fig. 2. Generative Models/Algorithms based on Maximum Likelihood Estimation

These generated samples are used to modify the model [26]. As figure 2 shows GANs use implicit density approximation of maximum likelihood, they can also generate samples from the random distribution of spatial data. The other most commonly used approaches of generative models based on the principle of maximum likelihood approximation are:

- *Variational Auto encoders:* These models come under the category of explicit density approximation of maximum likelihood. VAEs extract some main features from the input dataset and use them to produce output. Due to lack of feature extraction, the output signal is very weak and blurry, powerful neural networks are used to process the output further [27-28]. VAEs perform well in the state of equilibrium of two losses which are latent loss and reconstruction loss. These models are also used for unsupervised learning but require pre-training [29].

- *Fully Visible Belief Network (FVBN):* These models are mainly used to generate audio data. Similar to VAEs, these models are also a category of explicit density models and create a belief network [30]. These models use WaveNet neural network and chain rule of probability to generate raw audio signal. Due to WaveNet the signal generation is slow as it requires more computation time, so samples are not generated in parallel in these networks. It is also very difficult to accurately train these models [31-32].
- *Pixel CNN:* These are autoregressive models which use explicit distribution and generate an image pixel by pixel. It is easy to calculate log likelihood for these models and can be used with both continuous and discrete data. Instead of slow training process, these models can be directly applied in domains as compression and exploration etc. Due to the problem of blind spot, the performance of these models is not good [33-34].

**GANs based Image to Image Translation**

GANs can represent the image in single domain and multi domain. In single domain representation, the image existing in one domain is translated to a new domain and then it is again translated from new domain to the previous domain. In multi domain representation, the image from existing domain is translated to more than one domain [35-36]. GANs can work on both datasets: paired and unpaired. In paired datasets, the data in both domains (input/output) is of same nature/shape or dependent to each other. While in unpaired datasets, the data is independent in terms of shape and size. It learns the internal representation of data and can generate data with unlabelled data also. GANs are used for reinforcement learning as they can learn in an imaginary environment [37-38]. Some common GAN models used for image-to-image translation are as follows:

- *Pix2Pix GAN* is used to apply the style of one image according to the selected condition, on another image. Paired dataset is main requirement of this GAN and the non-availability of paired dataset limits its applications as compared to unpaired dataset. The generator and discriminator model uses simple GAN architecture including convolution layers, rectified linear activation function (ReLU) layers and batch normalization. Generator is trained to generate conditional images which are near to ground truth and to minimize the loss function; the discriminator is trained on paired dataset to find out the false generated image. Some common examples of this GAN are day to night conversion, summer to winter conversion, photo to scenery, areal to map, edges to photo, colorizing images etc [39].
- *Discover (DISCO) GAN* is basically used on unpaired dataset and measure the originality of reconstructed image following the translations of image from Domain 1 to Domain 2 and back to Domain 1. It has two reconstruction losses, separate for both the domains. To avoid the problem of mode collapse (in which generator draws a number of input values to the similar output), training in two diverse datasets is preferred. Disco GAN is used for Image generation, style transfer like face translation (hair colour, gender etc.), chair to car, edges to photos etc [40].

- *Dual GAN* generates realistic images in spite of the given input image by training both the generator and discriminator model with two different unlabelled datasets. The performance can be more improved by using labelled dataset. This model uses mini batches of similar size as a standard GAN model. The reconstruction error is used to train the generator again to minimize the loss in reconstructed image. Some common applications of Dual GAN are in conversion of day-night, photo-sketch, aerial-maps and label-facades [41].
- *Cycle GAN* is basically used to transfer style on images existing in different domains while preserving their contents. It is able to use a collection of photographs from different domains and extract the required style from images in the collection in order to perform the translation. This model calculates three losses to improve the reconstructed image: cycle-consistency loss, reconstruction loss and identity loss. Cycle GAN uses an additional hyper parameter to minimize the reconstruction/cycle-consistency loss. The main applications of Cycle GAN are in season transfer, style transfer, photograph generation from paintings, photograph enhancement etc. [42-43].
- *Conditional GAN* is used to generate an image with specific conditions of choice. In this encoded class label (features of choice) is also given to generator as well as discriminator beside the random signal as input. Then the generator generates a fake image of that particular class including the features of choice and discriminator gives output that the generated image belongs to that particular class or not. The code for conditional GAN is similar to DCGAN with some modifications. It is used to combine images with labels, combining noise and labels, and to generate images with some specific conditions like blue eyes, red hairs etc [44].
- *Stack GAN* is a combination of two or more GANs which are stacked together and generate a high-resolution image. This GAN works in two stages stage-I and stage–II. In stage–I a low-resolution image is generated by the generator using the random noise vector and in stage-II this low-resolution image is used to generate a high-resolution image. The second stage correct the defects and add compelling details to improve the resolution of image generated in stage-I. Stack GAN is just like a painter, which starts a painting by drawing lines and then move to shading and colouring. Stack GAN is mainly used to generate high resolution images [45].
- *Info GAN* provides disentangled representations which maximize the mutual information between a subset to represent the most important attribute or feature of data point. Info GAN can be used to extract the eye colour from a dataset of face to identify the person. It provides meaningful information by doing small modifications in the existing GANs. The objective function of info GAN is faster than normal GAN and in this discrete code constantly changes the digits of rows and columns to find meaningful attributes, while in regular GAN the digits are not changed frequently [46].
- *Deep Convolutional GAN (DCGAN)* provides network simplicity and eliminates fully connected layers in the convolutional network. It uses convolutional strides instead of max pooling and transposed convolution for up sampling. It uses ReLU in the generator and Leaky ReLU in the discriminator. It also uses batch normalization in all layers of generator network excluding the output layer and in input layer of discriminator network. Some common applications

of DC GAN are in anime character generation and data augmentation etc [47].

Table 1
An overview of various GAN models

| GAN Models | Authors | Year | Features |
|---|---|---|---|
| GAN | Goodfellow I et al. [4] | 2014 | Image generation |
| DCGAN (Deep Convolution GAN) | Radford et al. [47] | 2015 | Generate high quality images |
| LAPGAN (Laplacian pyramid) | Denton et al. [48] | 2015 | Generate High resolution image |
| Stack GAN | Zhang H. et al. [45] | 2016 | Text to image generation |
| Info GAN | Chen X. et al. [46] | 2016 | Use extra objective function |
| SR GAN (Super Resolution GAN) | Ledig C. et al. [17] | 2016 | Resolution enhancement |
| IGAN (Interactive GAN) | Zhu et al. [16] | 2016 | Generate photo realistic images |
| IAN (Introspective AN) | Brock et al. [18] | 2016 | Neural photo editor |
| Plug and Play GAN | Nguyen et al. [49] | 2016 | Improves diversity of samples |
| WGAN (Wasserstein GAN) | Martin A et al. [50] | 2017 | Remove mode collapse problem |
| Pix2Pix GAN | Phillip Isola et al. [39] | 2017 | Use paired dataset |
| DISCO GAN (Discover GAN) | Taeksoo K. et al. [40] | 2017 | Cross - domain generation |
| Cycle GAN | Zhu JY et al. [42] | 2017 | Style transfer |
| Conditional GAN | Isola P et al. [34] | 2017 | Conditional feature translation |
| Mean and Covariance GAN | Youssef et al. [51] | 2017 | Minimize loss function |
| Star GAN | Yunjey Choi et al. [52] | 2017 | Multi domain I – I Translation |
| LSGAN | Xudong M. et al. [53] | 2017 | Least-squares loss function |
| Perceptual GAN | Jianan Li et al. [54] | 2017 | Small object detection |
| Dual GAN | Zili Yi et al. [41] | 2018 | Image to image translation |
| Artsy GAN | Hanwen L. et al. [55] | 2018 | Improved quality and speed |
| Cartoon GAN | Yang Chen et al. [56] | 2018 | Image to cartoon conversion |
| Bi-flow GAN | Liuchun Y. et al. [57] | 2019 | Bi-directional image transformation |
| Attention GAN | Hao Tang et al. [58] | 2019 | Sharper and accurate images |
| Consistent Embedded GAN | FengXiong et al. [59] | 2019 | Generate Realistic images |
| Style GAN | Tero Karras et al. [60] | 2019 | Improved resolution |
| TuiGAN | Jianxin Lin et al. [61] | 2020 | Trained on two unpaired images |

The basic architecture of all GAN models is almost similar, while every model has some differences in designing and training as per their application. Table 1 shows a list of various GAN models along with their features.

## Conclusion

As various GAN models are discussed in previous sections, the major factors which need major attention are: Normalization and problem of Mode Collapse, which may occur in all models of GAN architectures. Normalization is required during the training of generator and discriminator to minimize the duplicity of data. Mode collapse is a problem when generator produces similar output images for several input images. Partial mode collapse is very common to occur, while complete is very rare. To minimize this problem, Minibatch feature is used which allows the discriminator model to compare a minibatch of real samples with minibatch of generated samples. Other method to overcome the problem of mode collapse is unrolled GANs which build computation graph of learning steps in discriminator and back propagate through all steps to calculate the gradient of features in generator network. Plug and play (PPGAN) provides a sampling approach to generate images with Markov Chain which estimates the gradients using denoising autoencoder that is trained with a number of losses including GAN losses.

Generative Adversarial Network is an immense area of research and new variants of GANs models are coming up frequently. GANs have so far shown very impressive results on tasks that were difficult to perform using conventional methods. In the research community, there is an open acceptance of GANs and their applications. There may be more advancement in deep learning techniques in future. As the advancement in GAN models is increasing at a very fast rate, soon these models will also be preferred in commercial applications. It may also be possible for everyone to create an SRGAN network and train it on their images.

**References**


1. I. J. Goodfellow, Y. Bengio, and A. Courville, “Deep Learning”, *MIT Press*, 2016. http://www.deeplearningbook.org.
2. I. J. Goodfellow, “On distinguishability criteria for estimating generative models”, *In International Conference on Learning Representations, ICLR’2015*.
3. I. J. Goodfellow, J. Shlens, and C. Szegedy, “Explaining and harnessing adversarial examples”, *ICLR’2015, arXiv: 1412.6572v3 [stat.ML]* 20 March, 2015.
4. I. J. 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 (NIPS’ 2014)*, pp 2672–2680.
5. Y. Ganin, V. Lempitsky, “Unsupervised Domain Adaptation by Backpropagation”, *arXiv: 1409.7495v2 [stat.ML]* 27 Feb 2015.
6. A. Aggarwal, M. Mittal, G. Battineni, “Generative adversarial network: An overview of theory and applications”, in *International Journal of Information Management Data Insights @Elsevier,* 2021.
7. Deep Learning with Azure , “Generative Adversarial Networks”, August 2018, pp 187-208.
8. V. Dumoulin and F. Vsin, “A guide to convolution arithmetic for deep learning”, *arXiv: 1603.07285v2 [stat.ML]* 11 June 2018.
9. A. Odena, V. Dumoulin and C. Olah, “Deconvolution and Checkerboard Artifacts”, *Distill*, 2016.
10. Y. Bengio, E. Thibodeau- Laufer, G. Alain, and J. Yosinski, “Deep generative stochastic networks trainable by backprop.”, *In ICML*'2014.
11. D. Kingma and J. Ba, “Adam: A method for stochastic optimization”, *arXiv preprint arXiv:1412.6980* (2014).
12. T. Salimansnet, I. J. Goodfellow, W. Zaremba, V. Cheung, A. Radford and X. Chen, “Improved Techniques for Training GANs”, *arXiv:1606.03498v1 [cs.LG]* 10 June 2016.
13. H. Alqahtani, M. Kavakli-Thorne and Gulshan Kumar, “Applications of Generative Adversarial Networks (GANs): An Updated Review”, *in Springer Link*, 2019.
14. V. Ringu, A. Eenaja, “A Proposal to use GAN for Speech Recognition in Natural Language Processing”, *international journal of creative research thoughts (IJCRT)*, volume 8, June 2020.
15. R. Anand, B. Singh and N. Sindhwani, “Speech Perception and Analysis of Fluent Digits’ String using Level-By-Level Time Alignment”, *International journal of information technology and knowledge management*, volume 2, issue 1, 2009, pp. 65-68.
16. J. Y. Zhu, K. Philipp, E. Shechtman, A. A. Efros, “Generative Visual Manipulation on the Natural Image Manifold”, in *European Conference on Computer Vision (ECCV)*, 2016.

17. C. Ledig, L. Theis, F. Huszar, J. Caballero, C. Andrew et al., "Photo-Realistic Single Image Super-Resolution using a Generative Adversarial Network", *arXiv:1609.04802[cs.CV],* 2016.
18. B. Andrew, L. Theodore, J. M. Ritchie, N. Weston, "Neural Photo Editing with Introspective Adversarial Networks", *arXiv: 1609.07093 [cs.LG].*
19. J. Gui, S. Zhenan, W. Yonggang, T. Dacheng, Y. Jieping, "A Rewiew on Generative Adversarial Networks: Algorithms, Theory, and Applications", *arXiv: 2001.06937 [cs.LG],* 2020.
20. G. Bakshi, R. Shukla, N. Sindhwani et al., "An Optimized Approach for Feature Extraction in Multi-Relational Statistical Learning", *Journal of scientific and industrial research (JSIR)*, volume 80, issue 6, 2021, pp. 537-542.
21. Y. Skandarani, P-M. Jodoin, A. Lalande, "GANs for Medical Image Synthesis: An Empirical Study", *arXiv:2105.05318v2 [eess. IV]* 19 Jul 2021.
22. K. Aggarwal, M. Singh Bhamrah and H. S. Ryait, "Detection of cirrhosis through ultrasound imaging by intensity difference technique", *EURASIP journal on image and video processing,* 2019.
23. K. Aggarwal, M. Singh Bhamrah and H. S. Ryait, "The identification of liver cirrhosis with modified LBP gray scaling and Otsu binarization", *Springer Plus*, 2016.
24. A. Islam, C. Long, A. Basharat and A. Hoogs, "DOA-GAN: Dual-Order Attention Generative Adversarial Network for Image Copy-Move Forgery Detection and Localization", proceeding of *IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, 2020, pp. 4676-4685.
25. R. Mehta, K. Aggarwal, D. Koundal et al., "Markov features based DTCWS algorithms for online image forgery detection using ensemble classifier in the pandemic", *International journal on expert systems with applications*, December 2021.
26. I. J. Goodfellow, "NIPS 2016 Tutorial: Generative Adversarial Networks", *arXiv: 1701.00160v4 [cs.LG].*
27. D. P. Kingma, M. Welling, "An Introduction to Variational Autoencoders", *arXiv: 1906.02691v3 [cs.LG]*, 2019.
28. D. P. Kingma, "Fast gradient-based inference with continuous latent variable models in auxiliary form", *Technical report*, *arxiv: 1306.0733* (2013).
29. D. J. Rezende, S. Mohamed and D. Wierstra, "Stochastic backpropagation and approximate inference in deep generative models", In *ICML'2014, Preprint: arXiv: 1401.4082*.
30. B. J. Frey, "Graphical models for machine learning and digital communication", *MIT Press,* 1998.
31. B. J. Frey, G. E. Hinton, and P. Dayan, "Does the wake-sleep algorithm learn good density estimators?" in proceedings of *8th International conference on Neural Information Processing Systems (NIPS'95)*, pages 661-670.
32. A. V. D. Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior and K. Kavukcuoglu, "Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499 (2016).*
33. A. V. D. Oord, N. Alchbrenner, O. Vinyals, L. Espeholt, A. Graves, K. Kavukcuoglu, "Conditional Image Generation with PixelCNN Decoders", *arXiv:1606.0328 [cs.CV]*, 2016.

34. L. Kunhua, Z. Peisi, Y. Zheng, Y. Kaige, M. Liu, "P_VggNet: A convolutional neural network (CNN) with pixel-based attention map", *PLOS ONE, https://doi.org/10.1371/journal.pone.0208497*, December 12, 2018.
35. X. Huang, M. Liu, S. Belongie, J. Kautz, "Multimodal unsupervised image-to-image translation" in Proceedings of the *European conference on computer vision (ECCV),* 2018, pp 172–189.
36. M. Egmont-Petersen, D. D. Ridder, H. Handels, "Image processing with neural networks-a review" in journal of *Pattern Recognition @ Elsevier Science Ltd,* 2002, pp 2279-2301.
37. L. A. Gatys, S. E. Alexander, M. Bethge, "Image Style Transfer using Convolutional Neural Networks", in *IEEE Explore,* 2016, pp 2414-2423.
38. L. Karacan, Z. Akata, A. Erdem and E. Erdem, "Learning to Generate Images of Outdoor Scenes from Attributes and Semantic Layouts", *arXiv:1612.00215v1 [cs.CV],* 1 Dec 2016.
39. P. Isola, J. Y. Zhu, T. Zhou, A. A. Efros, "Image-to-Image Translation with Conditional Adversarial Networks", *IEEE Explore*, 2017, pp 1125-1134.
40. T. Kim, M. Cha, "Learning to Discover Cross-Domain Relations with Generative Adversarial Networks", *arXiv: 1703.05192v2 [cs.CV],* 2017.
41. Z. Yi, H. Zhang, P. Tan, M. Gong, "Dual GAN: Unsupervised Dual Learning for Image-to-Image Translation", *arXiv:1704.02510 [cs.CV],* 2018.
42. J. Y. Zhu, T. Park, P. Isola, A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks", in Proceedings of the *IEEE international conference on computer vision*, 2017, pp 2223–2232.
43. J. Y. Zhu, T. Park, P. Isola, A. A. Efros, "Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks", *Computer Vision and Pattern Recognition,* 15 November 2018.
44. *P.* Isola, J. Y. Zhu, T. Zhou, A. A. Efros, "Image-to-image translation with conditional adversarial networks" In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp 1125–1134.
45. H. Zhang, T. Xu, H. Li, S. Zhang, X. Wang, X. Huang, D. Metaxas, "Stack GAN: Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks", *arXiv:1612.03242 [cs.CV],* 2017.
46. X. Chen, Y. Duan, H. Rein et al., "InfoGAN: Interpretable Representation Learning by Information Maximizing Generative Adversarial Nets", *arXiv: 1606.03657 [cs.LG]*, 2016.
47. A. Radford, L. Metz and S. Chintala, "Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks", *arXiv: 1511.06434v2 [cs.LG],* 7 Jan 2016.
48. E. Denton, S. Chintala, A. Szlam, R. Fergus, "Deep Generative Image Models using a Laplacian Pyramid of Adversarial Networks", *arXiv: 1506.05751[cs.CV],* 2015.
49. A. Nguyen, J. Y. Bengio et al., "Plug and Play Generative Networks: Conditional Iterative Generation of Images in Latent Space", *arXiv: 1612.00005 [cs.CV],* 2017.
50. M. Arjovsky, S. Chintala, L. Bottou, "Wasserstein GAN", *arXiv: 1701.07875 [stat.ML],* 2017.
51. Y. Mrouch, T. Sercu, V. Goel, "McGAN: Mean and Covariance Feature Matching GAN", *arXiv: 1702.08398v2 [cs.LG],* 2017.

52. Y. Choi, M. Choi, S. Kim, M. Kim, J. W. Ha and J. Choo, "Star GAN: Unified Generative Adversarial Networksfor Multi-Domain Image-to-Image Translation", *IEEE Xplore,* 2017, pp. 8789-8797.
53. X. Mao, Q. Li, H. Xie, Y. K. L. Raymond, Z. Wang and S. P. Smolley, "Least Squares Generative Adversarial Networks", *arXiv:16611.04076v3 [cs.CV],* 5 April 2017.
54. J. Li, X. Liang, Y. Wei, T. Xu, J. Feng and S. Yan, "Perceptual Generative Adversarial Networks for Small Object Detection", *arXiv: 1706.05274v2 [cs.CV]* 20 June 2017.
55. H. Liu, P. N. Michelini, D. Zhu, "Artsy–GAN: A style transfer system with improved quality, diversity and performance", *24$^{th}$ International Conference on Pattern Recognition (ICPR),* China, August 20-24, 2018.
56. Y. Chen, Y. U. Lai, Y. J. Liu, "Cartoon GAN: Generative Adversarial Networks for Photo Cartoonization", *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 9465-9474.
57. L. Yuan, D. Chen and H. Hu, "Unsupervised Object-Level Image-to-Image Translation using Positional attention Bi-Flow Generative Network", *IEEE Access*, volume 7, 2019.
58. H. Tang, X. Dan, N. Sebe and Y. Yan, "Attention-Guided Generative Adversarial Networks for Unsupervised Image-to-Image Translation", *arXiv: 1903.12296v3 [cs.CV]* 27 Aug 2019.
59. F. Xiong, Q. Wang and Q. Gao, "Consistent Embedded GAN for Image-to-Image Translation", *IEEE Access*, Volume 7, 2019.
60. T. Karras, S. Laine and T. Aila, "A Style-Based Generator Architecture for Generative Adversarial Networks", *arXiv: 1812.04948v3 [cs.NE]* 29 March 2019.
61. J. Lin, Y. Pang, Y. Xia, Z. Chen and J. Luo, "TuiGAN: Learning Versatile Image-to-Image Translation with Two Unpaired Images", *arXiv: 2004.04634v1 [cs.CV]* 9 Apr 2020.
62. Rinartha, K., Suryasa, W., & Kartika, L. G. S. (2018). Comparative Analysis of String Similarity on Dynamic Query Suggestions. In *2018 Electrical Power, Electronics, Communications, Controls and Informatics Seminar (EECCIS)* (pp. 399-404). IEEE.
63. Suryasa, I. W., Rodríguez-Gámez, M., & Koldoris, T. (2021). Get vaccinated when it is your turn and follow the local guidelines. *International Journal of Health Sciences*, *5*(3), x-xv. https://doi.org/10.53730/ijhs.v5n3.2938