Transactions on Computer Science and Intelligent Systems Research
ISSN: 2960-1800, eISSN: 2960-2238 | Volume 5
AIDML 2024

Exploiting Feature Space Manipulation for Image Generation in Deep Convolutional Generative Adversarial Networks

Yujie Yang *

College of Electrical Engineering and Automation, Jiangsu Normal University, Xuzhou, Jiangsu, 221116, China

* Corresponding Author Email: 3020210546@jsnu.edu.cn

Abstract. Image generation and manipulation are important research directions in the field of computer vision and graphics. Traditional methods of image generation and manipulation often necessitate substantial manual intervention or are constrained by the quality and diversity of the dataset, leading to a lack of authenticity and control in the images produced. This paper provides an extensive examination of image generation and manipulation using Deep Convolutional Generative Adversarial Networks (DCGAN) and feature weighting operations. The model successfully captured the basic structure of a face. Furthermore, the color and texture of the images are quite realistic, indicating that the model has learned some basic features of the face. In terms of feature weighting operations, specific features are successfully transferred from one image to another. This research contributes valuable insights to the field of computer vision and graphics and paves the way for future advancements in image generation and manipulation. Ethical considerations and efficient training methods are also discussed as important aspects for future research.

Keywords: Image generation; image manipulation; deep convolutional generative neural networks.

1. Introduction

Image generation and manipulation are important research directions in the field of computer vision and graphics. Their goal is to create or change the content and style of images using computer technology, thereby achieving various applications such as artistic creation, entertainment, education, and medical treatment. For example, image generation can be used to generate virtual scenes, characters, objects, etc., providing materials for games, animations, movies, etc.; image manipulation can be used to beautify, repair, enhance, transform, etc. images, providing services for photography, design, advertising, etc.

Nonetheless, conventional methods of image generation and manipulation typically necessitate significant manual intervention, or they are constrained by the dataset's quality and diversity, leading to a deficiency in the authenticity and control of the images produced. For example, template-based image generation methods need to predefine the structure and content of the image, which cannot adapt to complex and variable needs; retrieval-based image generation methods rely on a large amount of annotated data, which cannot generate novel and diverse images; deep learning-based image generation methods can use neural networks to automatically learn the distribution of data, but they often struggle to balance the quality and diversity of the generated images, or to control the features and style of the generated images.

In the past few years, the advancement of deep learning has led to the emergence of Generative Adversarial Networks (GANs). As a potent generative model, GANs have introduced innovative concepts and techniques for image generation and manipulation [1]. The original concept of GANs was introduced by Goodfellow and his team in the year 2014. It is an adversarial training-based method that can learn complex data distributions and thus generate high-quality images. The fundamental principle of GAN involves a pair of neural networks, with one being the generator, which is tasked with creating images from random noise; the other called the discriminator, responsible for judging whether the input image is real or generated. The two networks are in a state of competition, with the generator striving to fool the discriminator, and the discriminator working to identify the

generator's output. Through ongoing iterations, the generator progressively enhances the quality of the images it produces, while the discriminator incrementally improves its ability to judge images. Ultimately, they reach a state of Nash equilibrium, where the discriminator is unable to distinguish between the images generated by the generator [1].

The Deep Convolutional Generative Adversarial Network (DCGAN) is a variant of GAN that incorporates convolutional layers into the architecture of generative adversarial networks [2]. Through this fusion, DCGAN leverages the depth and complexity of CNN, enhancing the generation and analysis capabilities of complex images, resulting in unprecedented accuracy and quality. DCGAN introduces several methods to dissect the internal workings of the network: (1) Feature Activation Adjustment: Use an oscillator to adjust specific feature activations and observe their impact on the generated image. This helps to understand the role of each filter. (2) Occlusion Experiments: By systematically obscuring certain sections of the image and analyzing the model's response, it could be identified that the key components of the image. This technique provides a deeper understanding of how the model perceives and processes visual data, enhancing the performance and interpretability of image-based models. At the same time, check the feature maps that cause the most activation in the generator, revealing the specific image features that each layer is trained to capture. These methods not only promote the generation of realistic images but also deepen the understanding of how DCGAN learns complex image representations layer by layer. This knowledge is very valuable for improving model architecture and improving the learning efficiency of GAN-based learning. It is widely applied to medical image processing, virtual reality, artistic creation, and data enhancement [2, 3].

The model in this paper essentially adheres to the conventional DCGAN architecture. This includes the application of convolutional transpose layers in the generator and convolutional layers in the discriminator. Additionally, it incorporates batch normalization and ReLU activation functions in the generator, and LeakyReLU activation functions in the discriminator. On this basis, the following features have been added: (1) Image Enhancement: During the data preprocessing phase, this model employs various image enhancement techniques. These include random horizontal and vertical flips, as well as color jitter. Indeed, the application of these image enhancement techniques during the preprocessing stage can significantly bolster the model's robustness. This allows the model to more effectively manage a diverse range of images, thereby improving its performance and adaptability. (2) Vector Operations: This model uses a method of vector operations to generate new latent variables, which is a method of operation in the latent space, which can be used to explore and manipulate the characteristics of the generated images. (3) Model Saving and Loading: This model saves some latent variables during the training process and loads these latent variables in the subsequent image generation process. This allows for the generation of consistent images at any point after training, or for further latent space operations.

2. Related Work

Within the realm of image generation, there are many different types of methods and models, including:

Generative Adversarial Networks (GAN) [1]. It's composed of two main components: a generator and a discriminator. The generator's role is to create lifelike images, whereas the discriminator's task is to differentiate between authentic images and those produced by the generator. It generates high-quality images, with high realism. Indeed, it has a wide range of applications, including but not limited to, tasks like image synthesis and image transformation. This versatility makes it a powerful tool in the field of image generation and manipulation. However, the training is unstable and prone to mode collapse, and requires a lot of data and computing resources.

Variational Autoencoder (VAE) [4]. It is a probabilistic generative model that combines autoencoders and probability distributions. It generates images by learning the latent representation of data. It can generate diverse images and it learns latent space that benefits for image interpolation and editing.

Nonetheless, the images produced tend to be of lower quality and exhibit a high degree of blur. Additionally, controlling the attributes of the generated images presents a significant challenge [5].

Adversarial Autoencoder (AAE) [6]. AAE combines GAN and autoencoder, using adversarial learning to train the autoencoder. It combines the characteristics of autoencoders and the generative ability of GAN, and it can generate high-quality images. However, the training process is complex and requires adjustment of multiple hyperparameters [7].

Variational Adversarial Network (VAGAN) [8]. VAGAN combines VAE and GAN, using adversarial learning to train VAE. It combines the advantages of VAE and GAN, and the standard of the images produced by the model is high. Nonetheless, the training process is intricate and necessitates the tuning of numerous hyperparameters.

Moreover, there are many other models, such as Adversarially Learned Inference (ALI), Variational Autoencoder with GAN (VEEGAN), Adversarial Generator-Encoder Networks (AGE), etc [9]. Each model has its unique advantages and limitations, and the specific choice depends on the application requirements and problem background.

In summary, investigations and studies conducted in the domain of image generation is constantly promoting the understanding and application of image generation models, providing more possibilities for creating better images and applications.

3. Method

3.1. Dataset

The Diverse Faces Dataset is a comprehensive collection of 5484 facial images representing various ethnicities and age groups. This dataset encompasses a wide range of human diversity, making it valuable for research and applications in fields such as face recognition, age estimation, and emotion analysis. Key features of the dataset include: (1) Ethnic Diversity: The images include individuals from different racial backgrounds, ensuring representation across various ethnicities. (2) Age Variation: The dataset covers a broad age spectrum, from children to seniors. (3) Privacy Protection: To safeguard individual privacy, all facial images have been anonymized and processed to blur identifying features.

Researchers and practitioners can leverage this dataset to advance the development of robust and inclusive facial analysis algorithms. Whether you're exploring gender recognition, facial expression classification, or other related tasks, the Diverse Faces Dataset provides a rich resource for experimentation and evaluation.

3.2. Mechanism of DCGAN

DCGAN is composed of two components: a generator and a discriminator. The generator, a deconvolutional neural network, utilizes a random noise vector (drawn from a latent space) as its input and upscales it to produce images. Upsampling is accomplished via a sequence of deconvolutional layers that incrementally enhance the spatial resolution while altering the depth of the feature maps, as depicted in Fig. 1. The ReLU activation function is commonly employed in all layers, with the exception of the output layer, which utilizes the Tanh activation function to yield pixel values within the range of [-1, 1]. The primary objective of the generator is to fabricate images that the discriminator cannot distinguish from authentic images.

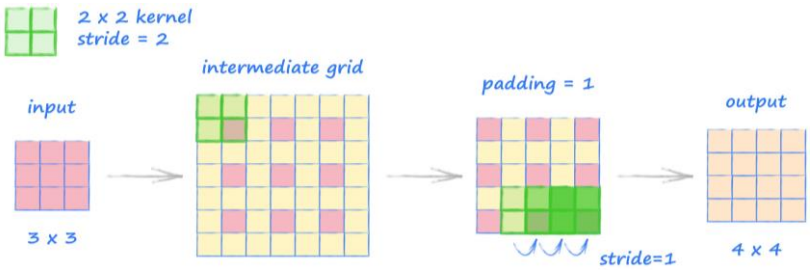

Fig. 1 Principle of deconvolutional layer [10].

The discriminator, a convolutional neural network, accepts an image as input and produces a single scalar value. This value represents the likelihood that the input image is authentic. The structure of generator and discriminator is shown in Fig. 2. The input image can either be from the real dataset or generated by the generator. The discriminator uses stridden convolutional layers to reduce the spatial dimension while increasing the depth of feature maps, capturing the hierarchical features necessary to discriminate between real and fake images. Leaky ReLU activation functions are used in the discriminator to prevent the dying ReLU problem and maintain gradient flow during training.

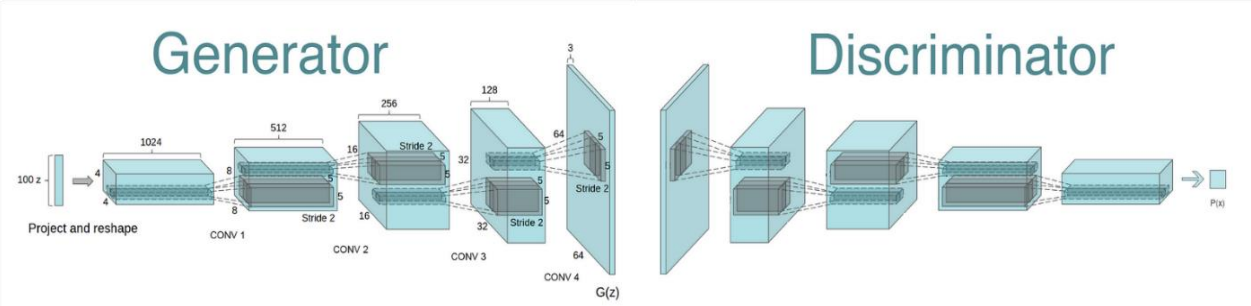

Fig. 2 Structure of generator and discriminator [11].

During the adversarial training of DCGAN, the generator and discriminator compete in a game-theoretic scenario. The discriminator strives to optimize its capacity to differentiate between authentic and counterfeit images, whereas the generator works to reduce the likelihood of its generated outputs being identified as counterfeit. This process is facilitated by a loss function that measures the discriminator's performance in classifying real and generated images (1) and the generator's performance in fooling the discriminator (2).

$$L_D = -\frac{1}{2}\left[\mathbb{E}_{x\sim P_{data}} logD(x) + \mathbb{E}_{z\sim P_z(z)} log(1 - D(G(z)))\right] \tag{1}$$

$$L_G = -\mathbb{E}_{z\sim P_z(z)} logD(G(z)) \tag{2}$$

During backpropagation, the weights of the network are updated via the method of gradient descent. The weights in the discriminator are updated as $\theta_D = \theta_D - \alpha\nabla_{\theta_D} L_D$, and the weights in the generator are updated as $\theta_G = \theta_G - \beta\nabla_{\theta_G} L_G$ where $\alpha, \beta (\in R)$ are the learning rates.

In addition to the deconvolutional and convolutional layers, batch normalization layers are integrated following each layer, except the generator's output layer and both the input and output layers of the discriminator. Batch normalization scales the data to the same scale, stabilizing and accelerating the training process. To be more specific, the output of the batch normalization layer for feature k is expressed as

$$y^{(k)} = \gamma^{(k)}\hat{x}^k + \beta^{(k)}, \hat{x}^k = \frac{x^{(k)} - \mu_B{}^{(k)}}{\sqrt{\sigma_B{}^{(k)^2} + \epsilon}} \tag{3}$$

Where $x^{(k)}$ is the input value for feature k, ${\mu_B}^{(k)}$ is the mean of feature k over mini-batch B, is the standard deviation of feature k over B, ϵ is a small constant added for numerical stability, and $\gamma^{(k)}$ and $\beta^{(k)}$ are learnable parameters that represent the scale and shift, respectively.

The discriminator employs Leaky ReLU as its activation function, while the generator utilizes ReLU, consistent with the original DCGAN study. This setup is determined through empirical observation rather than theoretical demonstration, hypothesized to be effective in the adversarial dynamics between the generator and discriminator, which parallel a "win-lose" scenario. During the initial phases of training, the discriminator is more likely to outperform the generator, effectively distinguishing real images from the low-resolution or nearly random outputs of the generator. Consequently, the discriminator's weights may experience less substantial updates compared to those of the generator. This discrepancy can potentially make the discriminator more susceptible to the vanishing gradient problem.

$$ReLu(x) = max(0, x) \tag{4}$$

$$LeakyReLu(x) = \begin{cases} x & if\ x > 0 \\ \alpha x & otherwise \end{cases} \tag{5}$$

3.3. Mechanism of Feature Weighting

The images generated by the model can be conceptualized as vectors within a latent space. By manipulating these vectors, specific features of the images can be isolated and modified. Feature vector arithmetic allows for the isolation and combination of features, such as transferring facial expressions between images. This process can be expressed mathematically as:

$$Z_{smiling\ woman} = Z_{smiling\ woman} - Z_{smiling\ woman} + Z_{smiling\ woman} \tag{6}$$

Where z represents the vector representations of the generated images in the latent space.

4. Results

4.1. Implementation Details

The following are the key parameters involved:

(1) Dimension of the latent variable (z_dim): This is the dimension of the random noise vector used by the generator to generate fake images. In this model, this value is set to 100.

(2) Batch size (batch_size): This is the number of images used in each training iteration. In this model, this value is set to 64.

(3) Image size (img_size): This is the size (height and width) of the generated fake images. In this model, this value is set to 64.

(4) Number of training epochs (num_epochs): This is the number of times the entire training dataset is traversed. In this model, this value is set to 500.

(5) Learning rate of the generator and discriminator (g_lr, d_lr): This is the step size used by the optimizer when updating model weights. In this model, both of these values are set to 0.0002.

(6) Momentum parameters of the optimizer (beta1, beta2): These are the momentum parameters used by the Adam optimizer. In this model, beta1 is set to 0.5, and beta2 is set to 0.999.

4.2. Results

The loss function results are shown in Fig. 3 and the generated images are demonstrated in Fig. 4.

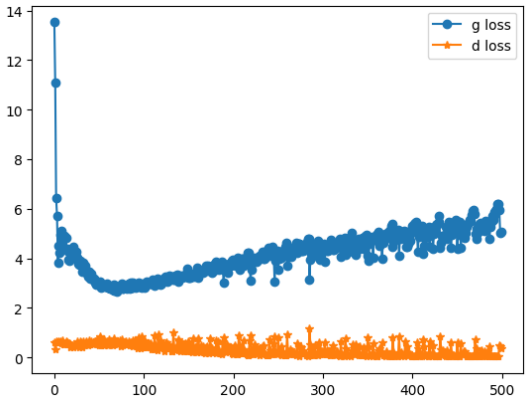

Fig. 3 Loss function of generator and discriminator (Figure Credits: Original).

In the implementation of the Deep Convolutional Generative Adversarial Network (DCGAN), the training process is guided by two distinct loss functions for the Generator and the Discriminator, components of the network.

The loss function of the Discriminator is structured to evaluate its proficiency in differentiating between authentic images from the dataset and the fabricated images produced by the Generator. This loss function is bifurcated into two segments: the initial segment calculates the Binary Cross Entropy (BCE) loss correlating the Discriminator's predictions for authentic images and a matrix of ones. The subsequent segment computes the BCE loss correlating the Discriminator's predictions for fabricated images and a matrix of zeros. The aggregate loss for the Discriminator is the summation of these two segments, signifying the cumulative error in categorizing both authentic and fabricated images.

On the other hand, the Generator's loss function is designed to measure how well it can deceive the Discriminator. More specifically, it calculates the Binary Cross Entropy (BCE) loss between the Discriminator's predictions for the fabricated images (which the Generator strives to have classified as authentic, i.e., 1) and a matrix of ones.

Throughout the training process, these loss functions provide a quantitative measure of the performance of the Generator and the Discriminator, guiding the optimization of the network parameters. The evolution of these loss functions during training provides valuable insights into the learning dynamics of the DCGAN model.

Fig. 4 Examples of generated images (Figure Credits: Original).

4.3. Feature Synthesis

The visualization results of feature synthesis are shown in Fig. 5 and Fig. 6, respectively. They include examples of three-feature synthesis and four-feature synthesis.

Fig. 5 Examples of three feature synthesis (Figure Credits: Original).

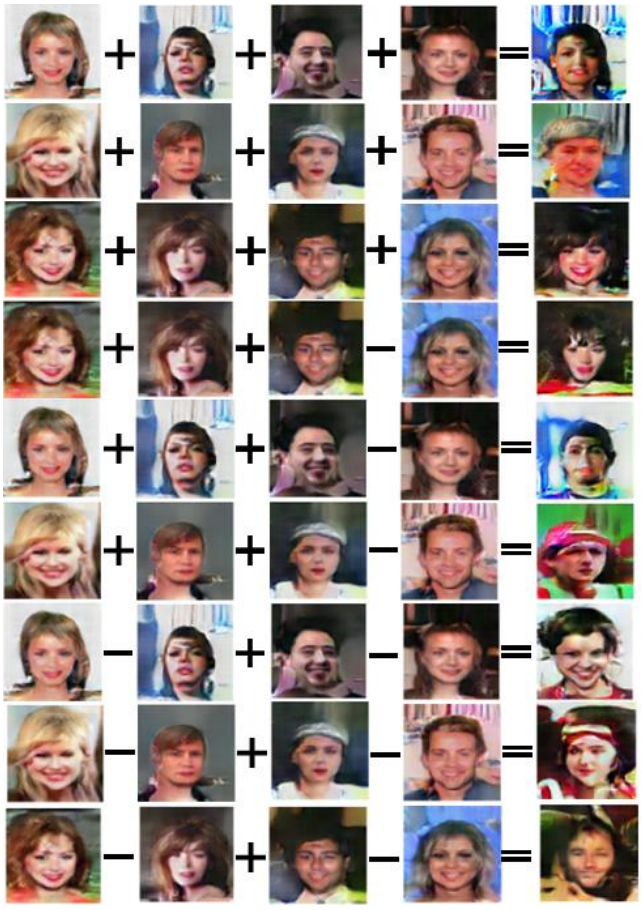

Fig. 6 Examples of four feature synthesis (Figure Credits: Original).

5. Discussion

In this study, DCGAN and feature weighting operations were successfully utilized for image generation and manipulation. These results demonstrate the potential and value of these techniques in the field of image generation. The model successfully captured the basic structure of a face, including the positions of the eyes, nose, and mouth. Furthermore, the color and texture of the images were quite realistic, indicating that the model has learned some basic features of the face. In terms of feature weighting operations, specific features (such as a smile) were successfully transferred from one image to another. However, the specifics of the face (such as the eyes, nose, and mouth) appeared indistinct, which could be attributed to the model's inadequate ability to grasp these intricate features

during the learning phase. Additionally, some parts of the face (such as the eyes and mouth) seemed to be irregular in shape, which may be due to some instability during the training process of the model.

To improve this model, the following methods could be attempted: (1) More Training Data: If possible, using more training data can help the model learn more features and details. (2) Depth and Width: Increasing the depth (more layers) and width (more nodes per layer) of the model can help the model learn more complex features. (3) More Complex Models: More complex generative models, such as StyleGAN, which is designed to generate high-quality facial images, could also be attempted.

These discoveries offer significant perspectives for forthcoming studies in the domain of image creation and alteration. Further studies are needed to address the identified weaknesses and explore the proposed improvements. The potential applications of these techniques are vast, promising exciting advancements in various domains.

In the future, subsequent directions could be further explored.

(1) Higher Quality Image Generation: Current generative models still face certain difficulties in generating high-resolution, realistic images. Future research can explore more powerful network structures and training techniques to enhance the quality of the images generated. For instance, the development of new architectures that can better capture the intricacies of natural images, or the application of novel training techniques that can stabilize the training process and prevent issues such as mode collapse.

(2) Adversarial Sample Defense: Generative models are susceptible to adversarial attacks, where small perturbations can cause the generated image to be misclassified. Future research can explore methods of adversarial sample defense to enhance the robustness of generative models. This could involve the development of new training regimes that make the model more resistant to adversarial perturbations, or the creation of new model architectures that are inherently more robust to such attacks.

(3) Aggregation of Multimodal Information: Combining information from multiple modalities (such as images, text, voice, etc.) to perform image generation is a challenging task. Future research can explore how to effectively aggregate multimodal information to achieve richer and more diverse image generation. This could involve the development of new models that can effectively fuse information from different modalities, or the creation of new datasets that contain rich multimodal information.

(4) Interpretable and Controllable Image Generation: While generative models can produce impressive results, understanding and controlling the generation process remains a challenge. Future work could focus on making these models more interpretable, such as by learning disentangled representations, or more controllable, such as by incorporating user guidance into the generation process.

(5) Ethical Implications: As the potency of generative models escalates, they concurrently evoke fresh ethical dilemmas, such as the possibility of exploitation in fabricating deepfakes. Upcoming investigations will necessitate contemplation of these matters, formulating techniques to identify and neutralize malevolent utilization of generative models, and instituting ethical norms for their application.

(6) Efficient Training: Training generative models can be computationally intensive and time-consuming. Future work could focus on developing more efficient training methods, such as by leveraging hardware accelerators, developing faster optimization algorithms, or designing models that require less data to train.

These are just a few of the possible future directions for research in image generation and manipulation. The field is rapidly evolving, and there are likely to be many exciting developments in the coming years. As always, the key to progress will be a combination of innovative thinking, careful experimentation, and rigorous evaluation.

6. Conclusion

This study has successfully demonstrated the application of DCGAN and feature weighting operations in image generation and manipulation. The model successfully captured the basic structure of a face, including the positions of the eyes, nose, and mouth. Furthermore, the color and texture of the images were quite realistic, indicating that the model has learned some basic features of the face. In terms of feature weighting operations, specific features (such as a smile) were successfully transferred from one image to another. However, the research also identifies challenges in generating high-resolution, realistic images and controlling the features and style of the generated images. The findings of this study fill a gap in the field of image generation and manipulation, demonstrating the potential of DCGAN and feature weighting operations. This research contributes valuable insights to the field of computer vision and graphics and paves the way for future advancements in image generation and manipulation. The successful application of these techniques could be beneficial for various applications such as artistic creation, entertainment, education, and medical treatment. While the results are promising, there are areas for improvement. Future work is suggested to explore more powerful network structures, training techniques, and methods for adversarial sample defense. Ethical considerations and efficient training methods are also discussed as important aspects for future research. The field is rapidly evolving, and there are likely to be many exciting developments in the coming years. As always, the key to progress will be a combination of innovative thinking, careful experimentation, and rigorous evaluation.

References

[1] Goodfellow, Ian, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. Advances in neural information processing systems, 2014, 27: 1-9.

[2] Radford, Alec, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. ArXiv Preprint, 2015: 1511.06434.

[3] Curtó, Joachim D., Irene C. Zarza, Fernando De La Torre, Irwin King, and Michael R. Lyu. High-resolution deep convolutional generative adversarial networks. ArXiv Preprint, 2017: 1711.06491.

[4] Kingma, Diederik P., and Max Welling. Auto-encoding variational bayes. ArXiv Preprint, 2013: 1312.6114.

[5] Higgins, Irina, Loic Matthey, Arka Pal, Christopher P. Burgess, Xavier Glorot, Matthew M. Botvinick, Shakir Mohamed, and Alexander Lerchner. Beta-vae: Learning basic visual concepts with a constrained variational framework. International Conference on Learning Representations, 2017, 3: 1-22.

[6] Makhzani, Alireza, Jonathon Shlens, Navdeep Jaitly, Ian Goodfellow, and Brendan Frey. Adversarial autoencoders. ArXiv Preprint, 2015: 1511.05644.

[7] Larsen, Anders Boesen Lindbo, Søren Kaae Sønderby, Hugo Larochelle, and Ole Winther. Autoencoding beyond pixels using a learned similarity metric. In International conference on machine learning, 2016: 1558-1566.

[8] Sinha, Samarth, Sayna Ebrahimi, and Trevor Darrell. Variational adversarial active learning. Proceedings of the IEEE/CVF international conference on computer vision. 2019: 5972-5981.

[9] Dumoulin, Vincent, Ishmael Belghazi, Ben Poole, Olivier Mastropietro, Alex Lamb, Martin Arjovsky, and Aaron Courville. Adversarially learned inference. ArXiv Preprint, 2016: 1606.00704.

[10] What are deconvolutional layers? URL: https://datascience.stackexchange.com/questions/6107/what-are-deconvolutional-layers/111532. Last Accessed: 2024/03/14

[11] Zero to Hero in Pytorch (ML+DL+RL). URL: https://www.kaggle.com/code/ashishpatel26/zero-to-hero-in-pytorch-ml-dl-rl/notebook. Last Accessed: 2024/03/14