Deep Learning-Based Text-to-Image Synthesis for Criminal Face Generation

A PROJECT REPORT

Submitted by

BL.EN.U4CSE20093	Malla Chandu
BL.EN.U4CSE20139	Ratnakaram Sasank
BL.EN.U4CSE20153	S Akshay
BL.EN.U4CSE20209	Bora Lohith

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

AMRITA SCHOOL OF COMPUTING, BENGALURU

AMRITA VISHWA VIDYAPEETHAM

BENGALURU 560 035

MAY 2024

AMRITA VISHWA VIDYAPEETHAM

AMRITA SCHOOL OF COMPUTING, BENGALURU, 560035

BONAFIDE CERTIFICATE

This is to certify that the project report entitled **"Deep Learning-Based Text-to-Image synthesis for Criminal Face Generation"** submitted by

BL.EN.U4CSE20093	Malla Chandu
BL.EN.U4CSE20139	Ratnakaram Sasank
BL.EN.U4CSE20153	S Akshay
BL.EN.U4CSE20209	Bora Lohith

in partial fulfillment of the requirements as part of **Bachelor of Technology** in "**COMPUTER SCIENCE AND ENGINEERING"** is a bonafide record of the work carried out under my guidance and supervision at Amrita School of Computing, Bengaluru.

Dr. Nidhin Prabhakar TV	*Dr. Gopalakrishnan E. A.*
Assistant Professor	Chair
Dept. of CSE, School of Computing	Dept. of CSE, School of Computing

This project report was evaluated by us on ……

Internal Examiner 1 Internal Examiner 2 External Examiner

ACKNOWLEDGMENTS

The satisfaction that accompanies successful completion of any task would be incomplete without mention of people who made it possible, and whose constant encouragement and guidance have been source of inspiration throughout the course of this project work.

We offer our sincere pranams at the lotus feet of **"AMMA", MATA AMRITANANDAMAYI DEVI** who showered her blessing upon us throughout the course of this project work.

We owe our gratitude to **Prof. Manoj P.**, Director, Amrita Vishwa Vidyapeetham Bengaluru Campus. We would like to place our heartfelt gratitude to **Dr. Gopalakrishnan E.A.,** Principal, Amrita School of Computing, Bengaluru for his valuable support and inspiration.

It is a great pleasure to express our gratitude and indebtedness to our project guide **Dr. Nidhin Prabhakar T V. Assistant Professor**, Department of Computer Science and Engineering, Amrita School of Computing, Bengaluru for his valuable guidance, encouragement, moral support, and affection throughout the project work.

We would like to thank express our gratitude to project panel members for their suggestions, encouragement, and moral support during the process of project work and all faculty members for their academic support. Finally, we are forever grateful to our parents, who have loved, supported and encouraged us in all our endeavors.

ABSTRACT

In contemporary law enforcement and criminal investigations, the need for accurate suspect identification remains paramount. This study introduces a novel approach to generate criminal facial images utilizing Deep Convolutional Generative Adversarial Networks (DCGAN) trained on a comprehensive dataset. This dataset comprises a vast collection of images, serving as a rich source for training generative models due to its diverse facial features. The proposed method harnesses the power of DCGAN architecture, renowned for its ability to learn complex data distributions and generate high-quality synthetic images. By fine-tuning the DCGAN model on the dataset, we enable it to synthesize realistic criminal facial images that encapsulate a variety of characteristics commonly associated with suspects in criminal investigations. Key aspects of our methodology include preprocessing techniques to extract facial features from the dataset, training the DCGAN model to learn the patterns and distributions of criminal facial attributes, and evaluating the generated images for realism and applicability in forensic contexts. Preprocessing is crucial as it ensures that the facial features extracted from the dataset are accurately represented and devoid of noise or irrelevant information. This step involves standardizing image sizes, normalizing pixel values, and employing face detection algorithms to isolate facial regions. The cleaned and processed data is then fed into the DCGAN model, which consists of neural networks, the generator and the discriminator. The generator creates images from random noise, while the discriminator evaluates them against real images from the dataset, providing feedback to improve the generator's output. Over multiple training iterations, the generator becomes adept at producing images that are indistinguishable from real ones, effectively learning the intricate details and variations present in criminal facial features. Training the DCGAN model involves several hyperparameters tuning to optimize its performance. These parameters include the learning rate, batch size, and the architecture of the neural networks. Careful adjustment of these parameters ensures that the model converges effectively, balancing the generator and discriminator to avoid issues such as mode collapse, where the generator produces limited variety in images. During training, we monitor the loss values of both networks to ensure steady progress and make adjustments as necessary to maintain a balanced training process. The generated images are periodically evaluated for quality and diversity to ensure they meet the criteria of realism and variability required for forensic applications. The generated criminal facial images are evaluated using various metrics and techniques.

TABLE OF CONTENTS

Page No.

ACKNOWLEDGEMENTS i

ABSTRACT ii

LIST OF FIGURES iv

LIST OF TABLES v

CHAPTER 1- INTRODUCTION

1.1 INTRODUCTION TO TEXT TO IMAGE GENERATION 1

1.2 MOTIVATION 2

1.3 OBJECTIVE OF THE PROJECT 2

1.4 SCOPE 2

CHAPTER 2 – LITERATURE SURVEY 4

CHAPTER 3 – SYSTEM REQUIREMENTS AND ANALYSIS

3.1 SOFTWARE REQUIREMENTS 12

3.2 HARDWARE REQUIREMENTS 13

CHAPTER 4 – SYSTEM DESIGN

4.1 HIGH LEVEL ARCHITECTURE DIAGRAM 15

4.2 LOW LEVEL ARCHITECTURE DIAGRAM 17

4.3 PROJECT FLOW 18

CHAPTER 5 – SYSTEM IMPLEMENTATION

5.1 GAN MODEL 19

5.2 DCGAN MODEL 21

CHAPTER 6 – SYSTEM TESTING 27

6.1 EVALUATION METRICS 28

CHAPTER 7 – RESULT AND ANALYSIS 32

CHAPTER8- CONCLUSION AND FUTURE SCOPE 39

REFERENCES 42

LIST OF FIGURES

Fig. 4.1	High Level Design	15
Fig 4.2	Low Level Design	17
Fig 4.3	Project Flow	18
Fig 5.1	GAN Model	20
Fig 5.2	Text Descriptions	21
Fig 5.3	List of Features	21
Fig 5.4	Generator Upscaling	22
Fig 5.5	Generator layers	23
Fig 5.6	Discriminator Downscaling	24
Fig 5.7	Discriminator Layers	25
Fig 6.1	DCGAN Model	27
Fig 6.2	Discriminator loss	29
Fig 6.	Generator loss	30
Fig7.1	Generated Image	32
Fig 7.2	Generated Image	32
Fig 7.3	Generated Image	33
Fig 7.4	Batch after 10 Epochs	33
Fig 7.5	Batch after 20 Epochs	34
Fig 7.6	Batch after 30 Epochs	34
Fig 7.7	Batch after 40 Epochs	35
Fig 7.8	Batch after 50 Epochs	35
Fig 7.9	FID Graph	36
Fig 7.10	Inception Graph	37
Fig 7.11	Output Comparison	40

LIST OF TABLES

Table 6.1	Values of Discriminator Loss	29
Table 6.2	Values of Generator Loss	31
Table 7.1	FID Scores of Epochs	36
Table 7.2	Inception Scores of Epochs	37

CHAPTER - 1

INTRODUCTION

1.1 Introduction To Text to Image Synthesis

In contemporary law enforcement and forensic investigations, the accurate identification of suspects plays a pivotal role in solving crimes and ensuring justice. Traditional methods of suspect identification, such as composite sketches and eyewitness descriptions, often suffer from subjectivity and inconsistency, leading to challenges in apprehending perpetrators and securing convictions. A rising number of people are interested in using cutting-edge technologies, especially those based on deep learning, to improve the precision and dependability of suspect identification procedures in order to overcome these restrictions. The goal of this project is to create a deep learning system that can create realistic face representations of possible suspects based on written descriptions. Through the use of Deep Convolutional Generative Adversarial Networks (DCGAN) and extensive training on face image datasets, our goal is to provide law enforcement authorities and forensic specialists with an instrument that may aid in the identification of suspects.

The motivation behind this research stems from the critical need within law enforcement for objective and data-driven methods of suspect identification. By synthesizing facial images based on textual descriptions, our proposed system offers a novel approach to augmenting traditional investigative techniques, potentially leading to more accurate and efficient criminal investigations. In this introduction, we provide an overview of the problem statement, outline the objectives of the project, define the scope of our research, and set the foundation for the next sections, where we go through the methodology, results, and implications of our work.

1.2 Motivation

The reason why this research stems from the critical need within law enforcement and forensic sciences for accurate suspect identification. Traditional methods often rely on eyewitness accounts or composite sketches, which can be subjective and prone to error. By leveraging advanced deep learning techniques, such as DCGAN, and training on large-scale datasets, this study try to provide a more objective and data-driven approach to generating criminal facial images. This innovative approach can revolutionize suspect identification processes, increasing the efficiency and reliability of criminal investigations.

1.3 Objective of the project:

The motive of this project is to develop a deep learning-based system capable of generating realistic facial images of potential suspects based on textual descriptions or other relevant input. Specifically, the project aims to:

1. Implement and fine-tune a Deep Convolutional Generative Adversarial Network (DCGAN) architecture for text-to-image generation.
2. Train the DCGAN model using the Celeb dataset, which contains a large range of facial images, to learn the important patterns and distributions of facial features.
3. Develop preprocessing techniques to extract and represent textual descriptions or attributes of criminal faces in a format suitable for input to the DCGAN model.
4. Evaluate the performance of the generated facial images in terms of realism, diversity, and applicability in forensic contexts.
5. Provide a user-friendly interface for law enforcement agencies and forensic experts to generate facial images of potential suspects based on textual descriptions, enhancing the efficiency and accuracy of suspect identification processes in criminal investigations.

1.4 Scope:

The scope of this project encompasses the development and evaluation of a deep learning-based system for generating realistic facial images of potential suspects from textual descriptions within the context of criminal investigations. Specifically, the project will:

1. Focus on utilizing the Deep Convolutional Generative Adversarial Network (DCGAN) architecture for text-to-image generation.

2. Utilizes Celeb dataset as the primary source for training the DCGAN model, leveraging its diverse collection of facial images.
3. Explore preprocessing techniques to extract and represent textual descriptions or attributes of criminal faces for input to the DCGAN model.
4. Evaluate the generated facial images in terms of realism, diversity, and applicability for forensic identification tasks.
5. Provide a proof-of-concept implementation with a user-friendly interface for generating facial images of potential suspects based on textual descriptions, demonstrating the feasibility and potential utility of the proposed approach in aiding law enforcement agencies and forensic experts.

The project's scope does not extend to real-time deployment in operational law enforcement settings or addressing legal and ethical considerations associated with the use of generated facial images in criminal investigations. Additionally, the system's performance limited by the quality and diversity of the Celeb dataset and the capabilities of the DCGAN architecture.

CHAPTER - 2

LITERATURE REVIEW

This project is built upon the foundation of significant prior work in the field of text to image synthesis. This review examines recent advancements and identifies areas for further exploration to enhance the proposed system.

Authors in [1] have Recent advancements in text-to-image synthesis leverage GANs, addressing challenges like training instability and mode collapse. Methods primarily focus on image generation or manipulation, emphasizing alignment between text and image distributions. However, ensuring accurate alignment remains a complex task due to the inherent ambiguity of natural language. Notably, while most research utilizes GANs, variations exist in dataset usage and evaluation metrics. Key open problems include enhancing GAN stability for higher-resolution outputs and improving alignment between text and image distributions to generate more semantically meaningful images. Contents in the [2] introduce a hybrid model, DMAttn GAN, for text-to-image synthesis, focusing on faces. By combining AttnGAN and DM-GAN, it achieves promising results on a newly proposed faces dataset. However, limitations exist in both image quality and correlation with input descriptions, suggesting the need for improved model architectures and longer training. The complexity of face generation demands more powerful models and intensive training, indicating a future direction for research to enhance synthesis quality and align generated images more closely with textual descriptions. In [3] authors have introduced a hybrid approach, C-GAN+ATT+CL, for text-to-image synthesis, leveraging GANs, attention mechanisms, and contrastive learning. By combining these techniques, it achieves performance in terms of image quality, diversity, and realism, surpassing other methods on datasets like COCO-Stuff. Utilizing contrastive learning with GANs improves image accuracy by training on similar and dissimilar pairs, enhancing the model's ability to capture nuanced textual descriptions. This approach showcases significant advancements in generating high-quality and diverse images from textual inputs, offering promising avenues for further research in text-to-image synthesis. Authors in [4] used the Attentional Generative Adversarial Network (AttnGAN) demonstrates enhanced high-resolution Bangla text-to-image generation by focusing on specific linguistic details and utilizing multi-stage processing. Achieving a notable inception score of 3.58±.06 on the CUB dataset underscores its effectiveness, marking a significant advancement in Bangla language processing.

Challenges persist in extending this approach to more diverse datasets like COCO and addressing complex linguistic structures in Bangla text. Further exploration is needed to enhance model robustness and applicability across various domains, paving the way for broader adoption of text-to-image synthesis in Bangla language applications.[5] The comparison between AttnGAN and DF-GAN for text-to-image synthesis highlights the effectiveness of both models, with DF-GAN outperforming AttnGAN in generating high-resolution images. AttnGAN excels in capturing word-level details with its attention mechanism, while DF-GAN incorporates Matching-Aware Gradient Penalty and a Deep text-image Fusion Block for improved output quality. However, challenges persist in achieving semantic consistency and background detail representation. Future research should focus on refining attention mechanisms, enhancing semantic consistency, and improving background synthesis to further advance text-to-image synthesis systems.

In [6] authors introduce a fully trained GAN for text-to-face synthesis, surpassing partially trained models by concurrently training text encoders and image decoders. A novel dataset amalgamating LFW, CelebA, and locally prepared data is created and labeled for comprehensive evaluation. Results show superior image quality and similarity to input sentences. The proposed model achieves lower FID and FSD scores compared to benchmarks, with favorable human ratings. Future research will focus on denser face information for improved synthesis quality. Addressing this can enhance applications in security domains like forensic analysis and public safety. The research done by the authors in [7] introduces a user-friendly GAN-based system for generating sunflower images from text input, revolutionizing design processes with affordability and creativity. It achieves high-quality results with an inception score of 3.45 ± 0.05, reducing reliance on costly designers. However, challenges remain in scalability and generalizability to other design domains. Future work could focus on expanding the dataset for more diverse patterns and optimizing the GAN architecture for faster training and higher- resolution outputs. Additionally, ensuring the system's adaptability to different design contexts and industries would further enhance its utility and accessibility. The review given by authors in [8] highlights advancements in text-to-image processing using GANs, emphasizing the integration of language and vision for generating meaningful images. Prospects lie in refining GAN architectures for increased realism, leveraging larger and more diverse bird datasets, and exploring applications in ornithology and education. Challenges include improving accuracy in representing bird species and ensuring the ecological relevance of synthesized content. Further research is needed to address scalability issues, optimize model training, and enhance interpretability, ultimately advancing our understanding of avian diversity and its conservation implications while unlocking creative and educational potentials.

Authors in [9] introduces a novel approach using StyleGAN for text-conditioned image generation and manipulation, achieving accurate results and disentangled feature manipulation. By leveraging non-linear regression and disentangled latent space, it enables precise attribute prediction and independent feature editing. However, reliance on textual descriptions may introduce inaccuracies, and pre-trained StyleGAN models may restrict the scope of editing. Authors in [10] address the problem of generating images based on visual attributes, such as color, shape, and texture. By leveraging conditional GANs, the authors propose a framework capable of synthesizing images conditioned on attribute vectors, enabling fine-grained control over the generated images' appearance. The paper presents Attribute2Image, a framework that enables conditional image generation from visual attributes. Through experiments on various datasets, including CelebA and LSUN, the authors demonstrate the model's ability to generate diverse and realistic images based on specified attributes. The summary highlights the framework's effectiveness in attribute-guided image synthesis, showcasing its potential applications in various domains, including facial image generation for forensic purposes. The paper [11] explores the task of generating realistic images from textual descriptions using Generative Adversarial Networks (GANs). It addresses the challenge of bridging the semantic gap between text and images by proposing a novel architecture that combines a text encoder with a conditional GAN to synthesize high-quality images. The introduction provides insights into the motivation behind the research and outlines the methodology employed to achieve the text-to-image synthesis task.In [12] authors address the problem of generating images based on visual attributes, such as color, shape, and texture. By leveraging conditional GANs, the authors propose a framework capable of synthesizing images conditioned on attribute vectors, enabling fine-grained control over the generated images' appearance. The paper presents Attribute2Image, a framework that enables conditional image generation from visual attributes. Through experiments on various datasets, including CelebA and LSUN, the authors demonstrate the model's ability to generate diverse and realistic images based on specified attributes. The summary highlights the framework's effectiveness in attribute-guided image.

Authors in [13] presents a novel methodology for learning deep representations of fine-grained visual descriptions, enabling effective cross-modal retrieval and text-to-image synthesis. Through extensive experiments on benchmark datasets, the authors demonstrate the superiority of their approach in capturing fine-grained visual semantics and generating semantically relevant images from textual descriptions. The summary highlights the potential applications of learned representations in various multimedia tasks, including forensic facial synthesis. Paper [14] introduces a method for synthesizing inputs that maximally activate individual neurons in n neural networks. By training deep generator networks to invert the representations learned by neural networks.

the authors aim to gain insights into the information encoded by individual neurons and improve the interpretability of deep learning models. The introduction discusses the importance of understanding neural network representations and outlines the proposed approach to synthesizing preferred inputs for neurons.Authors deals with Recent advances in generative adversarial networks (GANs)in [15] which led to their extensive use in different domains, including document image classification, where they have been employed to generate realistic images that can be used to augment training datasets and improve classification performance. The use of GANs for data augmentation has been explored in various contexts, including underwater image classification, iris image augmentation, and palmprint recognition, demonstrating their potential in addressing the problem of insufficient training data. In [16] authors highlight the significance of text-to-image synthesis, particularly in generating realistic human face images from textual descriptions using Generative Adversarial Networks (GANs). Recent advancements in GANs have enabled the creation of photo-realistic images based on various conditions such as layout, text, and scene graph. Existing approaches employ state-of-the-art techniques like deep convolutional neural networks, attention mechanisms, and feature disentanglement to generate high-quality faces that closely match the input text descriptions. However, challenges remain in effectively disentangling latent space characteristics and ensuring the stability of the training process.In [17], authors highlights the significance of text-to-image synthesis, particularly in generating realistic human face images from textual descriptions using Generative Adversarial Networks (GANs). Recent advancements in GANs have enabled the creation of photo-realistic images based on various conditions such as layout, text, and scene graph. Existing approaches employ state-of-the-art techniques like deep convolutional neural networks, attention mechanisms, and feature disentanglement to generate high-quality faces that closely match the input text description.

Authors in [18] aim to enhance text-to-image synthesis by generating realistic visuals that align closely with textual descriptions, demonstrating advancements in deep learning for image processing. The model is evaluated using the OXFORD 102 flower dataset, showcasing improved image quality and performance metrics like Inception Score, PSNR, and SSIM. The technique involves leveraging a text transformer with XLNet for semantic text extraction and a Deep Convolutional Text to Image Generative Adversarial Network (DCGAN) to merge text and image information efficiently. The model leverages a text transformer with XLNet for semantic text extraction and a Deep Convolutional Text to Image Generative Adversarial Network (DCGAN)in [19] to efficiently merge text and image information. Evaluation on face images using Frechet Inception Distance (FID) yielded a score of 43.815, showcasing advancements in converting textual descriptions into realistic visuals. The proposed method aims to enhance text-to-image synthesis by generating high-quality faces that closely match input text descriptions.

The work tells the potential of deep learning techniques in advancing text-to-image generation for various applications. Authors in [20] deals with a multi-path and staged mechanism which comprises three crucial components—a multi-scale generator, a multi-scale discriminator, and a text encoder. The multi-scale generator builds images at different resolutions and gradually improves them using a staged connection mechanism. The multi-scale discriminator assigns the generated images at different resolutions a rating based on their efficiency. Then, depending on the feature vector the text encoder generates from the input text, the generator and discriminator are conditional. The authors also propose a novel multi-path structure to boost the diversity of the created images. Content in [21] consists of an improved GAN method that combines the Whale Optimization Algorithm (WOA) and the DragonflyAlgorithm (DA), two optimization techniques. Before feeding the pre-trained CNN into the GAN generator to generate images, the authors first extract the features from the input text. To improve the training of the GAN, the hyperparameters of the generator and discriminator are optimized using the Whale Optimization Algorithm method and the Dragonfly algorithm, respectively. Research done in [22] consists of a comparison of the available strategies resulted in the selection of the most successful one. The DF-GAN performs better than rivals in a variety of areas, including quality of images and regard for descriptive language. The model is trained using the CelebA dataset and text descriptions. The data they collected because of their implementation shows that the learned generative model can generate a complete portrait in accordance with the information that was offered and generating accurate and varied representations of human faces. Promising results in terms of visual quality and similarity to input text descriptions

Authors in [23] deals with the CF-GAN, which is made up of a generator network that creates images using textual data and fused image attributes as input and a discriminator network that separates created images and real photos. Given that it has been trained to minimize adversarial loss, the generator is incentivized to create images. Improved image quality through a two-stage generation process and bidirectional attention mechanism, this model provided better segmentation results. Higher computational complexity than some existing methods, no detailed analysis of limitations or failure cases.Authors in [24] Deals with the technique then employs a self-attention process to generate representations of regional languages that faithfully reflect the various subtleties of the input text. These representations are merged and utilized to condition the production of images using a conditional generative adversarial network (cGAN). The cGAN is composed of a discriminator network that can discriminate between created images and genuine photos and a generator network that uses linguistic representations as input to create images. Paper [25] deals with its intriguing and promising uses, it has drawn the interest of many academics and produced several noteworthy advancements. After then, it became common practice to use DCGAN models to create realistic samples that faithfully represent real-world data.

In a conventional convolutional neural network, the input picture is processed through many convolutional layers, each of which applies a unique set of filters that extract different information from the input. The area of text-to-picture synthesis is addressed by the authors in [26], who have received important contributions from several exceptional individuals. GANs have been used in a number of synthetic image applications, such as super-resolution, picture production, and pictures in paintings. GANs have shown to be effective in producing conceptually coherent images from textual data when used in text-image manufacturing. Researchers present TextControlGAN, a controllable GAN based model that yields a 17.6% improvement in Inception Score (IS) and a 36.6% reduction in Fréchet Inception Distance (FID) on the Caltech-UCSD Birds-200 (CUB) dataset. The MirrorGAN text-to-image-to-text architecture was created by authors in [27] as a creative, global, local attentive, and semantic-preserving solution to this problem. Comprehensive experiments on two benchmark datasets made available to the public shown that MirrorGAN outperformed other representative state-of-the-art approaches. A brand-new framework called the Generative Adversarial What-Where Network (GAWWN). Unlike most previous approaches that made use of the GAN architecture, research in [28] proposed a technique based on the recently disclosed Implicit Maximum Likelihood Estimation (IMLE) framework. Using the attention approach and focusing on various sections of the textual description, the producing network produced images.

The CLIP-guided GAN text-image synthesis technique was introduced by the authors in [29]. contrastive learning has a contrastive learning component. Their method's main idea was to match picture and text integrations produced by a pre-trained visual language model called CLIP using a contrastive learning goal. They specifically employed the CLIP model to independently encode produced pictures and text descriptions in order to compare the positive and negative pairs of embeddings. Their strategy was demonstrated to outperform state-of-the-art techniques on many benchmark data sets. More recent studies have looked at text-image synthesis using adversarial learning and GANs. Authors in [30] built a GAN-based technique that combined object identification and semantic segmentation to generate pictures from text descriptions. Metrics that were both quantitative and qualitative demonstrated that the proposed strategy outperformed the existing GAN-based techniques. Their basic idea was to use a semantic consistency loss in addition to a contrastive learning loss to improve the quality and diversity of the produced pictures. The limited variety of pictures that GANs could generate was one of their primary issues. In order to overcome this issue, contrastive learning has been applied often in the workplace to promote variety in the generated pictures. In order to create engaging artificial visuals, the authors of [31] presented deep convolutional generative adversarial networks that integrated image embeddings with natural language. Additionally, there has been some recent work in the area of cross-domain picture feature learning.

Authors in [32] employed stack design, which solves the issue by using a single generator and discriminator network. Using numerous up blocks in the generator, affine transformation is utilised to fuse text features in each layer of the DF-GAN. The adversarial loss is employed as the loss function, and skip-z with truncation approach yields high picture quality results. With the use of the popular face dataset, TediGAN, progressive GAN, and DF-GAN have all been used to produce pictures while incorporating their ideas. GAN is the underlying technology in all of these techniques. The methodology in [33] is developed to help law enforcement with their investigations. Authors developed a method that uses unconstrained neural networks with deep learning to produce high-quality images from descriptions of texts provided by eyewitnesses. A comparison of the available strategies resulted in the selection of the most successful one. Authors in [34] The DF-GAN performs better than rivals in a variety of areas, including quality of images and regard for descriptive language. The model is trained using the CelebA dataset and text descriptions. The data they collected as a result of their implementation shows that the learned generative model is capable of generating a complete portrait in accordance with the information that was offered and generating accurate and varied representations of human faces. Promising results in terms of visual quality and similarity to input text descriptions. The system effectively detected DR across the entire process, from early to late phases, with great sensitivity and specificity. Limited evaluation on a single dataset, no detailed comparison with other state-of-the-art methods.

The methodology in [35] consists of two primary components: They are the vision-language matching module and the GAN-based picture generation module. To extract text and picture data, the vision-language matching module employs an image feature extractor and a pre-trained language model. These characteristics are then compared to create an attention map, which is used to control how the GAN component creates images. The discriminator and generator functions of the GAN module are trained on producing images of excellent quality that correctly represent the input text using an array of adversarial loss, perception reduction, and attention reduction. This utilizes optimized GAN approach for text-to-image synthesis, and uses a combination of two optimization algorithms to improve GAN training. The proposed approach may require additional computational resources due to the use of optimization algorithms.

2.1 Literature Survey:

Recent advancements in text-to-image synthesis using Generative Adversarial Networks (GANs) have addressed key challenges such as training instability and mode collapse. Various approaches, including hybrid models like DMAttn GAN and C-GAN+ATT+CL, leverage attention mechanisms and contrastive learning to enhance the alignment between textual descriptions and generated images.

Achieving state-of-the-art performance on datasets like COCO-Stuff. Specific models, such as AttnGAN and DF-GAN, have shown strengths in word-level detail capture and high-resolution image generation, while specialized applications have been explored for face synthesis, design, and specific languages like Bangla. Research highlights include improvements in GAN stability, higher resolution outputs, and better semantic consistency. Future directions focus on refining model architectures, optimizing training processes, and expanding dataset diversity to enhance the quality, realism, and applicability of generated images across various domains, including ornithology, forensic analysis, and creative design.

CHAPTER – 3

SYSTEM SPECIFICATIONS

3.1 Software requirements

This project is a sophisticated system that relies on a specific set of software requirements encompassing programming languages, libraries, frameworks, and tools necessary for successful implementation and deployment. These requirements are meticulously chosen to ensure robust development, training, and evaluation of the deep learning model, as well as to facilitate user interaction with the system through a web interface.

Python: The primary programming language used for writing the code. It's widely used in data science and machine learning due to its simplicity and the vast array of libraries available.

Google Collab: an online integrated development environment (IDE) with a cloud-based Python programming environment. It's especially helpful for machine learning projects since it provides free access to TPUs and GPUs, which may greatly accelerate the training of models.

Pandas: A Python library used for data manipulation and analysis. In your project, it is used to read, preprocess, and manipulate data from CSV files.

NumPy: A fundamental package for scientific computing with Python. It's used for numerical operations on arrays and matrices, which are crucial in data preprocessing and transformations in machine learning tasks.

OpenCV (cv2): A library focused on computer vision tasks. In your project, it's used for image processing tasks such as reading images, converting color spaces, and drawing key points on images.

3.1 Hardware requirements

The hardware requirements for this project are dictated by the need to handle large datasets, process high-resolution images, and perform intensive computational tasks associated with training deep neural networks.

The following specifications are recommended to ensure optimal performance throughout the development and executionphases of the project:

Processor (CPU): Intel Core i7 or equivalent AMD processor, with at least 6 cores and 12 threads, to handle multiple processes efficiently during data preprocessing and model training.

Graphics Processing Unit (GPU): NVIDIA GPU with CUDA Compute Capability 3.5 or higher, such as NVIDIA GeForce RTX 3060 or better, with at least 8 GB of VRAM. A high-end GPU is crucial for accelerating the training and inference processes of deep learning models.

Random Access Memory (RAM): Minimum of 16 GB RAM, with 32 GB or more recommended for efficient handling of large datasets and to facilitate multitasking with minimal swapping to disk.

Storage: Solid State Drive (SSD) with at least 512 GB of space for the operating system, software installations, and dataset storage. An additional external or cloud-based storage solution is recommended for backup and extra space if dealing with extensive datasets.

Network: Stable high-speed internet connection for downloading datasets, accessing cloud services, and software updates. A minimum of 100 Mbps download and upload speeds is recommended.

CHAPTER - 4
SYSTEM DESIGN

4.1 High Level Design

A deep learning model called a deep convolutional generative adversarial network (DCGAN) is shown in Figure 4.1. Two neural networks that are in competition with one another make up DCGANs. A single network, referred to as the generator, attempts to produce visuals that are identical to real ones. The discriminator, the other network, attempts to distinguish between a true and fraudulent picture. Through mutual competition, both networks can enhance their capacity to produce lifelike pictures.

Here is a more detailed explanation of the layers involved in the process:

- **Noise Vector:** This is a random vector of numbers that is used to add noise to the image. The noise helps to prevent the generated image from becoming too similar to the training data.
- **Text Encoding:** This layer encodes the text input into a vector of numbers. This vector is then used by the generator to create an image that corresponds to the text description.
- **Projection Layer:** This layer projects the noise vector and the text encoding into a lower-dimensional space. This is done to improve the efficiency of the network.
- **Convolutional Layers:** These layers are the core of the generator network. They learn to extract increasingly complex features from the data. For example, the first convolutional layer might learn to detect edges, while the later convolutional layers might learn to detect more complex shapes, such as faces or objects.
- **Sequential Layers:** These layers are used to perform additional operations on the data, such as batch normalization and leaky ReLU activation.
- **Generated Image:** This is the final output of the generator network. It is a synthetic image that has been created by the network.

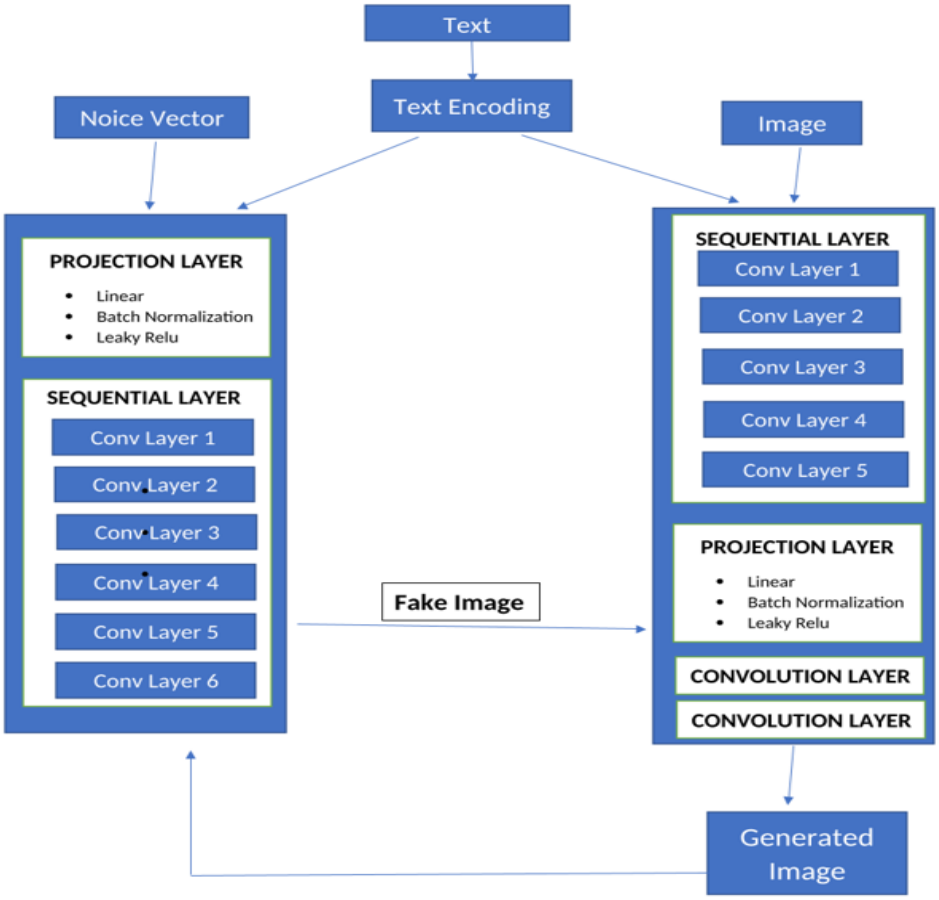

Figure 4.1 high level design.

The projection layer consists of the following components:

Linear Layer: This completely linked layer, sometimes referred to as a dense layer, gives the incoming data a linear transformation. It calculates the inputs' weighted total plus an additional bias component.

Batch normalisation: By normalising the inputs to each layer, this approach helps to enhance the training of deep neural networks. By decreasing the internal covariate shift, it aids in stabilising and quickening the training process.

Leaky ReLU: Leaky Rectified Linear Unit is an activation function that helps prevent "dying ReLUs" by permitting a little, non-zero gradient when the input is negative.

4.2 Low Level Design:

Fig 4.2 illustrates the core training loop of a Generative Adversarial Network (GAN) system, involving two primary neural networks: the Generator and the Discriminator. The Generator functions like an artist, tasked with creating new images based on the training data it receives. It starts with random noise and iteratively refines this noise to produce images that increasingly resemble real ones. On the other hand, the Discriminator acts like an art critic, evaluating both real images from the dataset and the images generated by the Generator, aiming to accurately distinguish between the two. The training process is driven by a continuous competition between these two networks. Initially, the Generator produces a new image. This generated image, along with a real image from the dataset, is then evaluated by the Discriminator. The Discriminator analyzes these images and outputs a judgment indicating whether each image is real or fake. The judgment includes a loss value, which quantifies how accurate the Discriminator's predictions are. Based on the Discriminator's feedback, the Generator adjusts its parameters to create more realistic images in future iterations. This adjustment process involves backpropagation and gradient descent methods, common techniques in training neural networks. The goal for the Generator is to minimize the loss, thereby improving its ability to produce high-quality images that can deceive the Discriminator. Simultaneously, the Discriminator also undergoes training to enhance its detection capabilities, aiming to accurately identify the generated fake images. This feedback loop drives both networks forward. The Generator continuously improves its image creation process, making its outputs more realistic. The Discriminator, in turn, becomes more proficient at detecting subtle flaws in the images. Ideally, after sufficient training, the Generator can produce images that are indistinguishable from real ones, consistently fooling the Discriminator. The specific details in Fig 4.2, such as additional blocks or arrows, may provide further insights into the type of GAN being used. Preprocessing the training data is a crucial step in this process. It ensures that the images fed into the system are standardized and suitable for effective training. This includes tasks like resizing images to a consistent dimension, normalizing pixel values, and possibly augmenting the dataset to increase its diversity. Proper preprocessing ensures that the neural networks can learn efficiently from the data.

Balancing the Generator and Discriminator during training is critical. If one network becomes significantly stronger than the other, it can lead to issues such as mode collapse, where the Generator produces a limited variety of images, or the Discriminator becoming too effective, easily classifying all generated images as fake. Careful tuning of hyperparameters, such as learning rates and batch sizes, is necessary to maintain this balance. Evaluating the GAN's performance involves subjective assessments by human evaluators and objective metrics like the Inception Score (IS) and Fréchet Inception Distance (FID). These metrics compare the statistical properties of the generated images against real images, providing a quantitative measure of their quality and diversity.

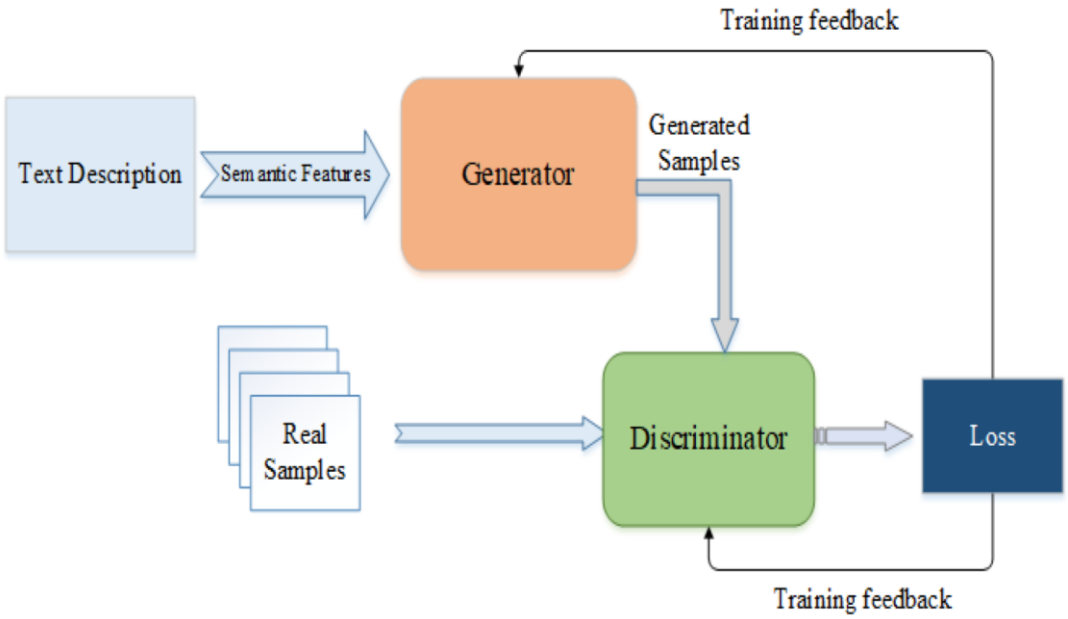

Figure 4.2 low level design.

4.3 Project Flow:

Figure 4.3 is a flowchart of a text-to-image generation system. Here's a breakdown of the process:

The system receives text data from the user, describing the image they want to generate. Using a trained model a large neural network trained on a vast dataset of text-image pairs—the system generates an image based on the provided description. This model has learned to identify patterns in text-image pairs and uses this knowledge to create new images from text descriptions. Once the image is generated, the user has the option to adjust model parameters, such as the image style, colors, or level of detail, to fine-tune the output. After finalizing the image, the user can save it. Text-to-image generation systems, a type of AI, are gaining popularity due to their wide range of applications in fields like product design, video game development, and special effects.

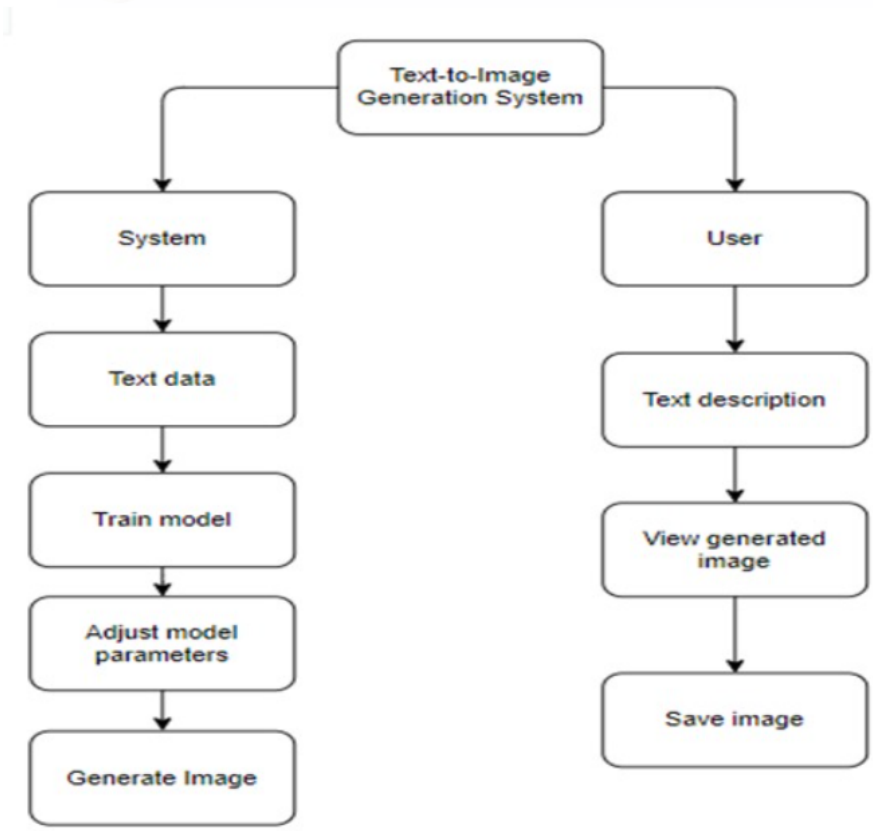

Fig 4.3 project flow

CHAPTER – 5
SYSTEM IMPLEMENTATION

From the chapter 4 system design we can see the way our system is designed and how it is trained.in the chapter 5 we can see the actual implementation we have done during the project. Initially we implemented GAN model and observed the output which is not that good by doing the more literature survey we came to know that DCGAN can be the better option for text to image gemeration.in the chapter 5 we are going to explain the GAN model with architecture and DCGAN model along with the evaluation metrics. For both the model initial steps are same as importing libraries and data preprocessing and both have discriminator and generator in the model and trained using a large dataset after each epoch loss for generator and discriminator is calculated and train both. The dataset used is CelebA which consists of 202,599 number of face images of various celebrities with 10,177 unique identities, but names of identities are disclosed.

5.1 GAN Model:

Fig 5.1 refers to Implementing a Generative Adversarial Network (GAN) involves several essential steps, starting with defining the model architecture. A GAN consists of two neural networks: the Generator and the Discriminator. The Generator takes random noise as input and generates data that resembles the training data. In TensorFlow, the generator network is built using a sequential model that includes dense layers followed by batch normalization and activation functions like LeakyReLU and tanh. The Discriminator, on the other hand, takes an image as input and outputs a probability indicating whether the image is real (from the dataset) or fake (generated by the Generator). This network is also constructed using a sequential model with dense layers and LeakyReLU activation, ending with a sigmoid activation to output the probability.

Next, the data preparation step involves loading and preprocessing the dataset. For example, using the celeA dataset, which consists of images of celebrities, the data is normalized to the range [-1, 1] and reshaped to include a channel dimension. This ensures compatibility with the network architecture, which expects inputs in a specific shape.

The models are then compiled with appropriate optimizers and loss functions. The Discriminator is compiled with a binary cross-entropy loss and an Adam optimizer. For the combined model, which stacks the Generator and Discriminator, the Discriminator is set to non-trainable to ensure that during the Generator's training phase, only the Generator weights are updated. This combined model also uses binary cross-entropy loss and is compiled with an Adam optimizer.

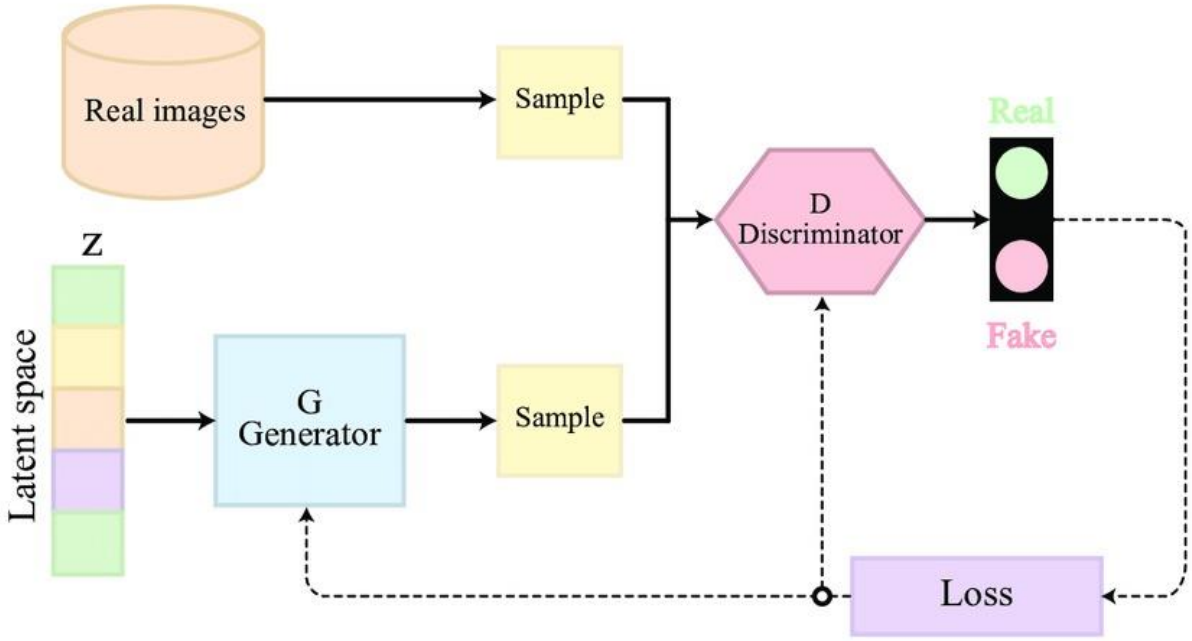

Fig 5.1 GAN model

Training the GAN involves an iterative process where the Discriminator and Generator are trained alternately. In each epoch, a batch of real images is sampled from the training data, and a batch of fake images is generated using the Generator. The Discriminator is trained on both batches to distinguish between real and fake images. Subsequently, the Generator is trained to fool the Discriminator by generating more realistic images. This is achieved by training the combined model with the Discriminator's weights fixed, using the real labels to maximize the Discriminator's output for fake images. The progress is monitored by printing the losses of both networks and saving generated images at regular intervals for visual inspection.

Finally, evaluating the GAN involves assessing the quality of the generated images. This can be done through visual inspection or by using quantitative metrics like the Inception Score or Frechet Inception Distance (FID). These metrics provide a measure of how realistic and diverse the generated images are compared to the real dataset. By following these steps, one can implement and train a GAN capable of generating realistic images or other types of data, depending on the specific application and dataset used.

5.2 DCGAN Model:

The CelebA dataset is a widely used large-scale face dataset with over 200,000 celebrity images, each annotated with 40 attribute labels. For implementing a DCGAN for text-to-image generation, the first step is to download and prepare this dataset. The images need to be resized to a consistent size (typically 64x64 pixels for simplicity in initial models) and normalized. Additionally, the corresponding attribute annotations can be used to generate textual descriptions that will serve as input for the text-to-image generation process.

Each image has the pretext for training purpose and an array against each image consisting of 1 and -1 indicating which feature is there in that image fig 5.2 and fig.3 gives the complete idea about the text and array.

```
000002.jpg,The lady has pretty high cheekbones. Her hair is brown. She has a big nose and a slightly open mouth. The woman seems young and is smiling.
000003.jpg,"His hair is wavy. He has big lips, narrow eyes and a pointy nose. The man looks young."
000004.jpg,"She has straight hair. She has a pointy nose. The lady seems attractive and young. She is wearing earrings, lipstick and a necklace."
000005.jpg,"She has arched eyebrows, big lips, narrow eyes and a pointy nose. The woman is attractive, young and has heavy makeup. She is wearing lipstick."
```

Fig 5.2 text descriptions

```
000001.jpg,-1,1,1,-1,-1,-1,-1,-1,-1,-1,-1,1,-1,-1,-1,-1,-1,-1,1,1,-1,1,-1,-1,1,-1,-1,1,-1,-1,-1,1,1,-1,1,-1,1,-1,-1,1
000002.jpg,-1,-1,-1,1,-1,-1,-1,1,-1,-1,-1,1,-1,-1,-1,-1,-1,-1,-1,1,-1,1,-1,-1,1,-1,-1,-1,-1,-1,-1,1,-1,-1,-1,-1,-1,-1,1
000003.jpg,-1,-1,-1,-1,-1,-1,1,-1,-1,-1,1,-1,-1,-1,-1,-1,-1,-1,-1,-1,1,-1,-1,1,1,-1,-1,1,-1,-1,-1,-1,1,-1,-1,-1,-1,-1,1
000004.jpg,-1,-1,1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,1,-1,-1,1,-1,-1,-1,1,-1,1,-1,1,1,-1,1
000005.jpg,-1,1,1,-1,-1,-1,1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,1,-1,-1,-1,-1,1,1,-1,-1,1,-1,-1,-1,-1,-1,-1,-1,-1,1,-1,-1,1
000006.jpg,-1,1,1,-1,-1,-1,1,-1,-1,-1,-1,1,-1,-1,-1,-1,-1,-1,1,-1,-1,1,-1,-1,1,-1,-1,-1,-1,-1,-1,-1,-1,1,1,-1,1,-1,-1,1
```

Fig 5.3 list of features

The core components of a DCGAN are the generator and the discriminator. The generator network's role is to create realistic images from random noise vectors and text embeddings, while the discriminator evaluates the authenticity of the generated images, distinguishing between real and fake images.

The generator network combines a noise vector with a text embedding to produce an image. This is typically achieved by concatenating these two inputs and passing them through a series of transposed convolutional layers to upscale the feature maps to the desired image size. Each layer is explained using fig 5.4.

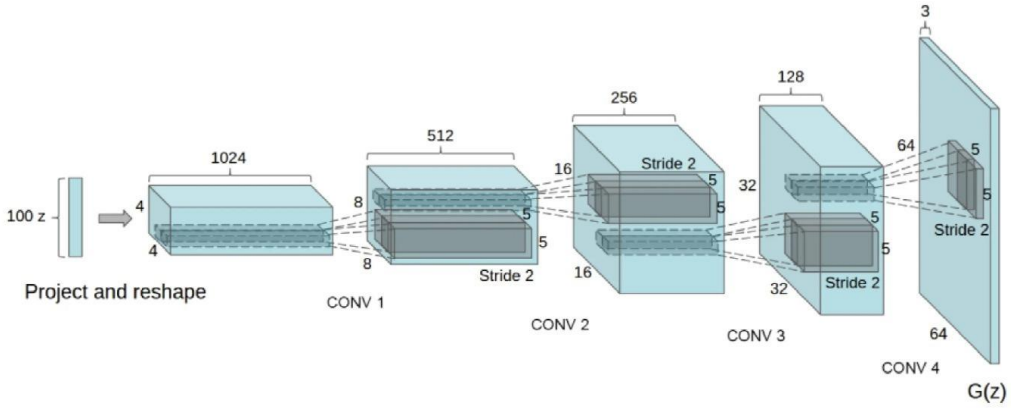

Fig 5.4 generator Upscaling

The class is initialized with parameters defining the dimensions of the text embeddings, noise vectors, and the number of image channels. Within the architecture, the Generator starts with a fully connected layer (`fc1`) that processes the concatenated noise vector and text embedding, outputting a feature map with 128 channels and dimensions 8x8. Following this, a sequence of transposed convolutional layers, encapsulated in the `conv_blocks` module, progressively up samples the feature map to generate the final image. These layers include transposed convolutions, batch normalization, and ReLU activations, aiming to increase spatial dimensions while refining features. The output is a synthesized image with dimensions 64x64, generated by applying a Tanh activation to ensure pixel values fall within the range [-1, 1]. During the forward pass, the noise vector and text embedding are concatenated and processed through the fully connected layer before being reshaped and passed through the convolutional layers to produce the output image. This Generator module serves as a pivotal component within the DCGAN framework, facilitating the generation of realistic images from textual descriptions within the CelebA dataset. fig 5.5 gives the clear idea about the layers.

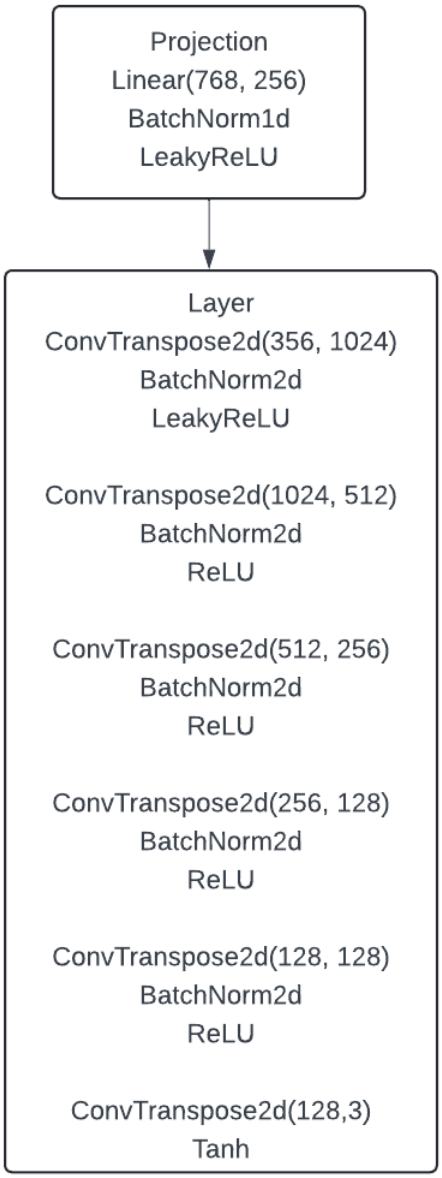

Fig 5.5 Generator Layers

Fig 5.6 refers the discriminator network's role is to assess whether the images it receives are real (from the dataset) or fake (generated by the generator). It also incorporates text embedding to ensure that the generated images match the given textual descriptions. This typically involves concatenating the image with the text embedding and passing them through a series of convolutional layers to produce a binary classification output. Fig 5.6 explains the discriminator in detail.

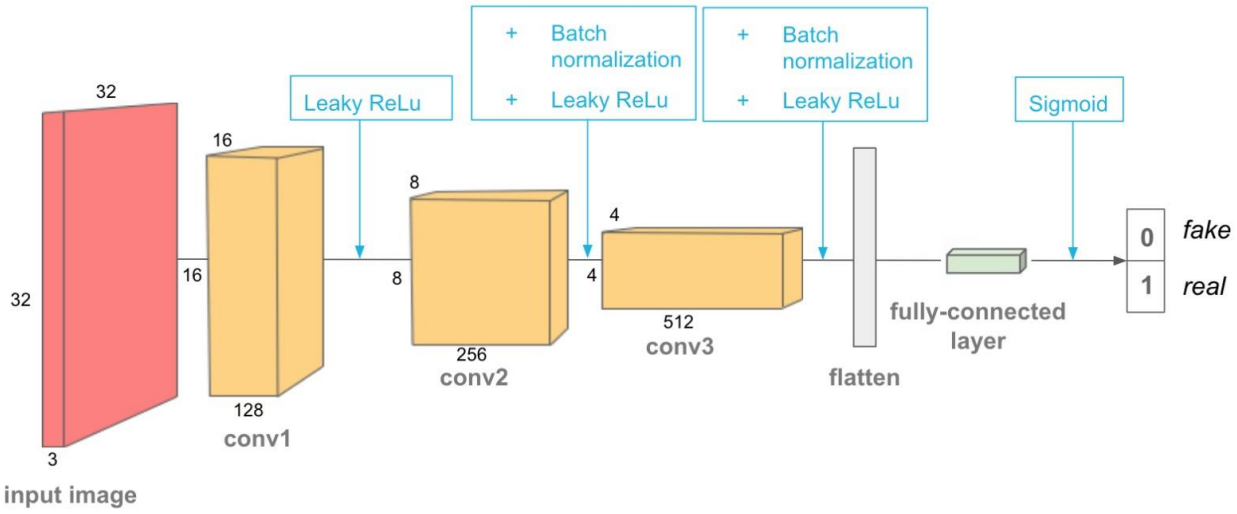

Fig 5.6 Discriminator Downscaling

To evaluate the authenticity of images generated by a DCGAN while considering the corresponding text embeddings. The discriminator's architecture begins by defining a series of convolutional blocks. The first convolutional layer takes the image and the text embedding as inputs, concatenating them along the channel dimension. This combined input passes through several convolutional layers, each followed by a LeakyReLU activation function to introduce non-linearity and Batch Normalization layers to stabilize training and improve convergence. The convolutional layers progressively down sample the input image, reducing its spatial dimensions while increasing the number of feature maps, effectively capturing high-level features. The final convolutional layer outputs a single-channel feature map, which is flattened into a 1D vector. This vector, representing the extracted features from the image, is concatenated with the text embedding vector. The resulting vector is passed through a fully connected layer to produce a single scalar output. This output represents the discriminator's assessment of the input image's authenticity, conditioned on the given text embedding. Overall, the discriminator's structure ensures that it learns to discern not only the realism of the images but also their consistency with the provided textual descriptions, playing a crucial role in the adversarial training process of the GAN.

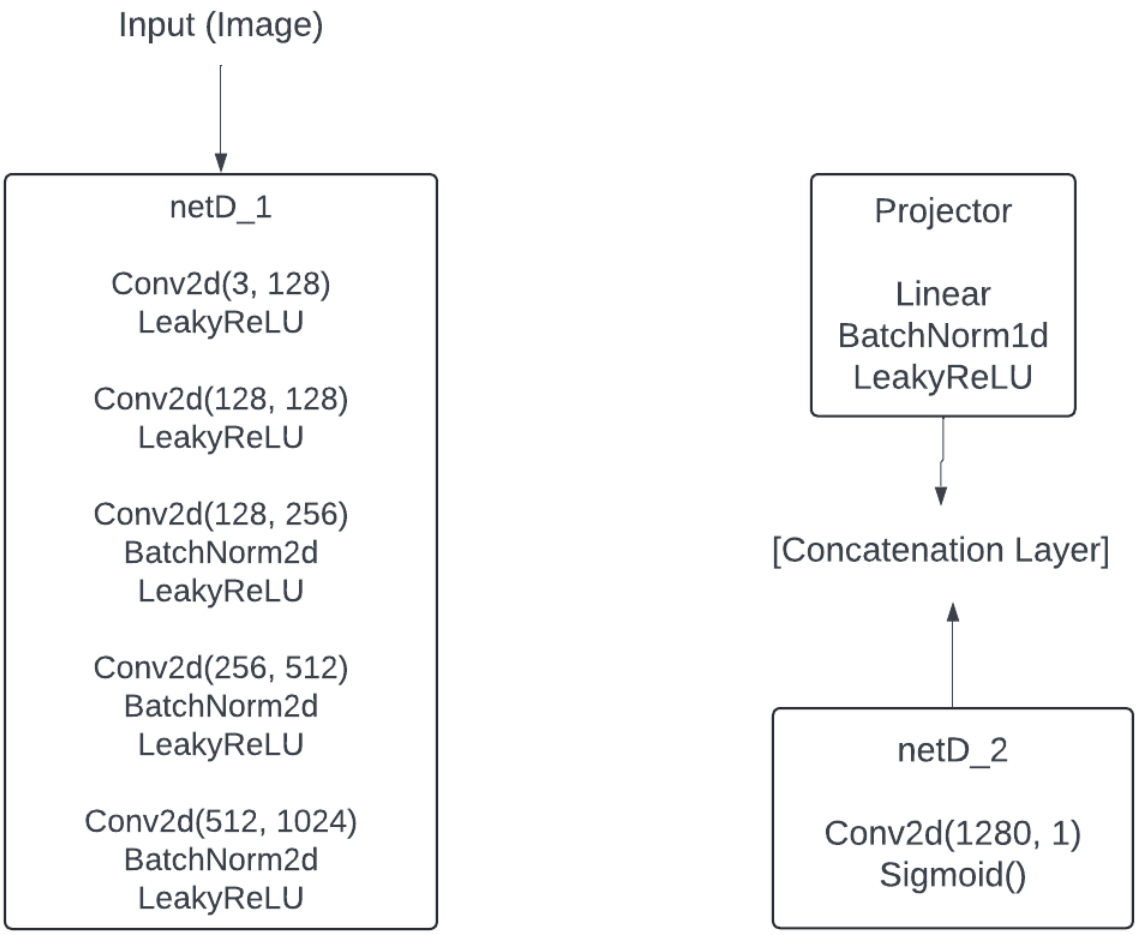

Fig 5.7 Discriminator Layers

Using the CelebA dataset, the Deep Convolutional Generative Adversarial Network (DCGAN) shown in Figure 5.7 is trained through an adversarial yet cooperative connection between the discriminator and generator neural networks. The generator's goal is to make visuals that are indistinguishable from real photographs by using textual descriptions to generate increasingly realistic images. It receives as inputs a text embedding that is obtained from the associated text description and a noise vector that adds randomization. To create a full-sized picture, the generator up samples this combined input via a sequence of transposed convolutional layers. In contrast, the discriminator's purpose is to evaluate the legitimacy and pertinence of the pictures that it gets. It takes both real images from the dataset and synthetic images from the generator, along with their respective text embeddings. By processing these through several convolutional layers, the discriminator aims to output a probability score indicating whether an image is real or generated, and whether it matches the input text description.

The training loop for this DCGAN involves iterative updates to both networks to enhance their performance continually. Initially, real images and their corresponding text descriptions are fed into the discriminator to establish a baseline for realness. Simultaneously, the generator produces synthetic images from noise and text embeddings, which are also evaluated by the discriminator. The discriminator's loss is computed based on its ability to correctly identify real versus fake images and the relevance of the images to the text descriptions. This loss is then backpropagated to update the discriminator's weights. In the subsequent step, the generator's loss is calculated based on how well the discriminator is fooled by the generated images, with the goal of improving the realism and textual relevance of these images. This adversarial process, where the generator strives to create better images while the discriminator gets better at detecting fakes, continues iteratively. Over time, this dynamic tension leads to progressively better image generation, as the generator learns to create images that are not only realistic but also closely aligned with the given text descriptions.

CHAPTER - 6

SYSTEM TESTING

The image depicts a flowchart outlining a system for generating fake images using a generative adversarial network (GAN). GANs consist of two competing neural networks: a generator that crafts images and a discriminator that determines if an image is real or fake. The process starts with a random noise input and text description. The text is converted into a machine-readable format and combined with the noise. This data is then fed into the generator network, which progressively refines it into a realistic image. The discriminator analyzes the generated image and provides feedback to the generator in a loop until a convincing fake image is produced.

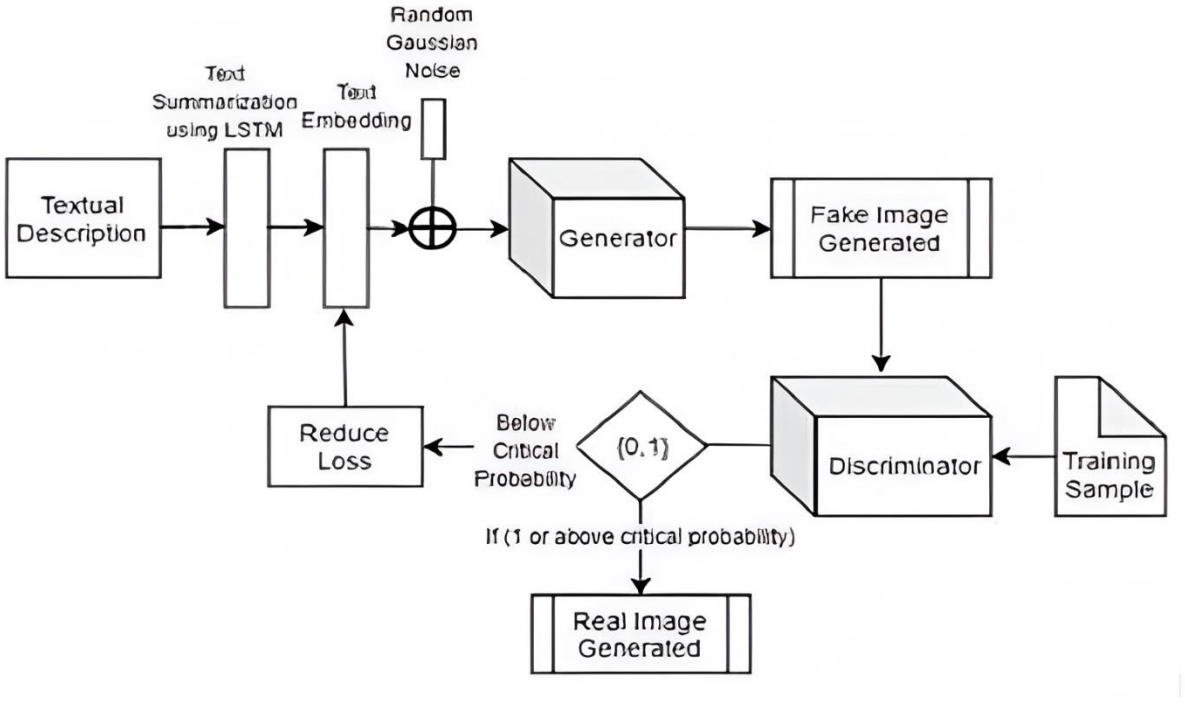

Fig 6.1 DCGAN Model

Fig 6.1 gives the clear idea about the DCGAN model then we are testing against the two-evaluation metrics FID score and Inception score. FID Score should be less because it is the distance between and real image and generated image where as inception score should be high because as high inception score as good the image.

6.1 Evaluation Metrics:

The Fréchet Inception Distance (FID) score and the Inception Score (IS) are two commonly used metrics for evaluating the quality of generated images in generative models, particularly Generative Adversarial Networks (GANs). Here are the formulas for each:

Fréchet Inception Distance (FID) Score: The Fréchet Inception Distance compares the statistics of real and generated images in the feature space of a pretrained Inception network. Given two multivariate Gaussian distributions. Figure 6.2 gives complete idea about formula.

$$\mathbf{FID(x,g) = || U_x - U_g||^2 + Tr(\sum_x + \sum_g - 2(\sum_x \sum_y)^{1/2}}$$

Where:

- x and g denote the real and generated image sets, respectively.
- U_x and $\sum_x$ are the mean and covariance of the inception features for the real images.
- U_g and $\sum_g$ are the mean and covariance of the inception features for the generated images.
- || .|| denotes the Euclidean distance.
- Tr is the trace of matrix.

Inception Score (IS): The Inception Score evaluates the quality and diversity of generated images based on the class probabilities predicted by a pretrained Inception network. It is calculated as the exponential of the Kullback-Leibler (KL) divergence between the conditional class distribution and the marginal class distribution of the generated images. Figure 6.3 is the formula for inception score.

$$\mathbf{IS(G) = exp(€_{x\sim pg}\ [D_{kl}(p(y|x)\ ||\ p(y))])}$$

Where:

- X is a generated image
- P_g is the distributed of generated images.
- P(y|x) is the conditional probability (The inception model's output(probabilities) of the class label given the image x
- P(y) is the marginal class distribution, computed as $p(y) = \int p(y|x)p_g(x)dx$
- $D_{KL}(p||q)$ is the Kullback-Leibier

The Inception Score is high when the generated images are clear and belong to one class with high confidence (quality) and when there are many different classes present in the generated images (diversity).

These two metrics offer complementary insights into the quality and diversity of generated images and are commonly used together for comprehensive evaluation of generative model.

Since we tested the model three times, the information for the three tests is shown in Figure 6.4 in three different colours. In Figure 6.4, discriminator loss is defined as the mistake or disparity between the ground truth labels and the discriminator's predictions made during a Generative Adversarial Network (GAN) training process. The goal of the discriminator in a GAN is to identify actual samples from false ones, while the goal of the generator is to produce samples that are identical to real samples. Often, the sum of the actual loss and the fictitious loss represents the overall discriminator loss. Reducing the discriminator loss pushes the discriminator to improve its ability to discern between authentic and fraudulent samples, which helps direct the generator to generate more authentic samples. values are clearly given in Table 6.1.

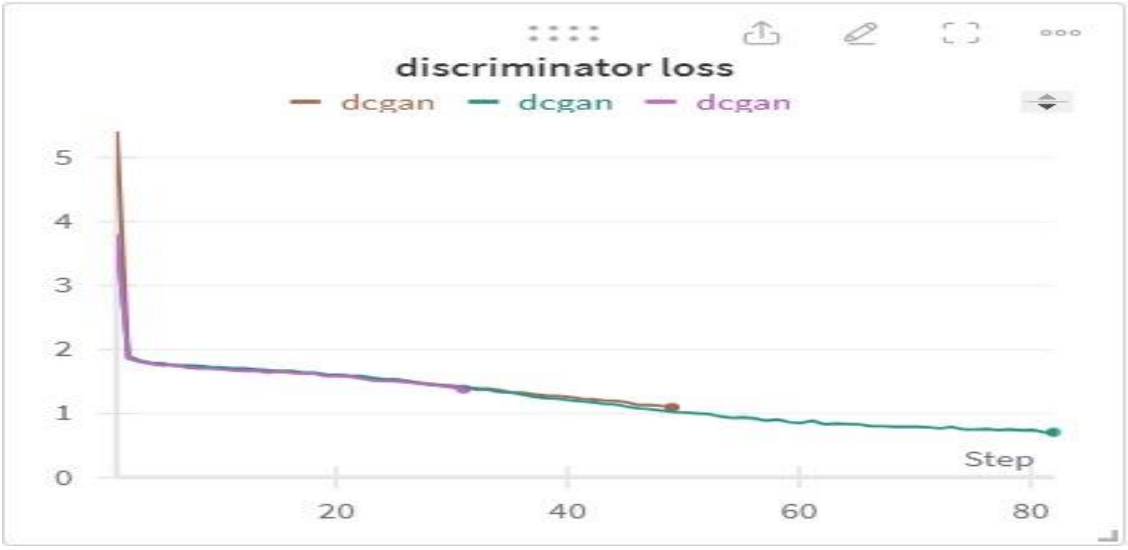

Fig 6.2 Discriminator Loss

Table 6.1 Values of Discriminator Loss

No of epochs	Discriminator Loss
Epoch 10	1.700
Epoch 20	1.604
Epoch 30	1.434
Epoch 40	1.259
Epoch 50	1.077

Figure 6.5 The generator loss refers to the measure of error or discrepancy between the generated samples produced by the generator and the ground truth labels during the training of a Generative Adversarial Network (GAN). In a GAN, the generator's objective is to produce samples that are indistinguishable from real samples by the discriminator.

The generator loss typically consists of several components:

1. Adversarial Loss: This component measures how well the generator fools the discriminator. It computes the difference between the discriminator's predictions for fake samples generated by the generator and the ground truth labels indicating that they are real. The generator aims to minimize this loss, as it indicates the degree to which the discriminator is fooled by the generated samples.

2. Feature Matching Loss: This component encourages the generated samples to match the statistics (e.g., mean activation values) of real samples. It measures the discrepancy between the features extracted from the generated samples and the features extracted from the real samples. Minimizing this loss helps improve the overall quality and diversity of the generated samples.

Fig 6.3 Generator Loss

3. Regularization Loss: Additional regularization terms, such as L1 or L2 regularization, may be added to encourage desirable properties in the generated samples, such as smoothness or sparsity. These regularization terms penalize deviations from these properties, contributing to the overall loss.

The total generator loss is typically a combination of these components, often with different weighting factors to balance their contributions. Minimizing the generator loss encourages the generator to produce high-quality samples that are both realistic and diverse, thereby improving its ability to fool the discriminator and generate novel content. Table 6.2 have values of respected epochs.

Table 6.2 Values of Generator Loss

No of epochs	Generator Loss
Epoch 10	27.191
Epoch 20	26.905
Epoch 30	26.843
Epoch 40	26.344
Epoch 50	26.138

CHAPTER – 7

RESULTS AND ANALYSIS

In the initial phase of the process, the model undergoes a comprehensive training regimen using a meticulously curated dataset. This dataset is integral to the model's ability to learn and adapt, as it allows the model to optimize its parameters effectively. During this training phase, the model iteratively adjusts its weights, striving to achieve the highest possible level of accuracy and performance. Once the training process identifies the optimal set of weights and their corresponding values, these are recorded and stored. This step is critical because it preserves the model in its best-performing state, ready for future tasks. The model is then saved, encapsulating its trained parameters and ensuring that this optimal configuration is retained for subsequent use.

Transitioning from training to evaluation mode marks a significant shift in the model's operation. In this mode, the model is prepared to process new input data provided by the user. When the user inputs text, this textual data is first subjected to sentence encoding. Sentence encoding is a sophisticated process that transforms the input text into a numerical format known as a tensor. This tensor representation is essential as it allows the model to handle and interpret the input text computationally. The encoded text, now in tensor form, is fed into the model, which then proceeds to evaluate it. This evaluation is not merely a passive reception of data but involves the model actively interpreting and processing the tensor to generate a coherent output.

During the evaluation phase, the model's output is critically assessed against a discriminator to ensure that the generated image accurately reflects the input text. This discriminator acts as a quality control mechanism, verifying the fidelity and relevance of the generated image. The model synthesizes an image that corresponds to the textual description, showcasing its ability to translate verbal descriptions into visual representations effectively. This generated image is then printed, providing a visual manifestation of the model's interpretative capabilities. Figures 7.1, 7.2, and 7.3 exemplify this process. Each figure presents an image generated from the respective input text provided above it, demonstrating the model's proficiency in bridging the gap between textual descriptions and visual outputs. These illustrations highlight the practical applications of the trained model, emphasizing its potential in various real-world scenarios where accurate image generation from text is required.

Input Text: The female has high cheekbones and an oval face. Her hair is black. She has a slightly open mouth and a pointy nose. The female is smiling, looks attractive and has heavy makeup. She is wearing earrings and lipstick.

Generated Image:

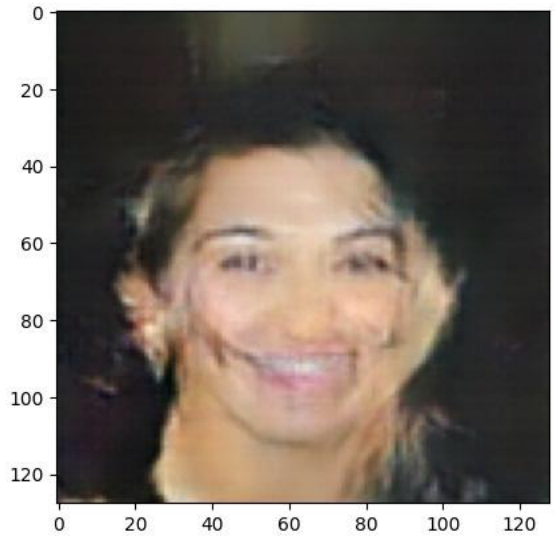

Fig 7.1 Generated Image

Input Text: A boy with a white skin tone and black hair having a smiling face with a clear background.

Generated Image:

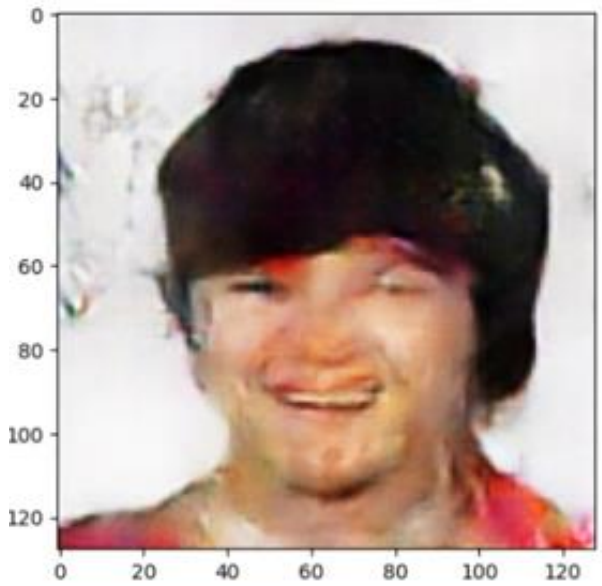

Fig 7.2 Generated Image

Input Text: A portrait of a man who is perceived as handsome. He has notably large, expressive eyes and thick, long eyelashes. His facial features are symmetrical and well-defined, contributing to his attractive appearance. The man's skin is clear, and his expression is confident yet approachable.

Generated Image:

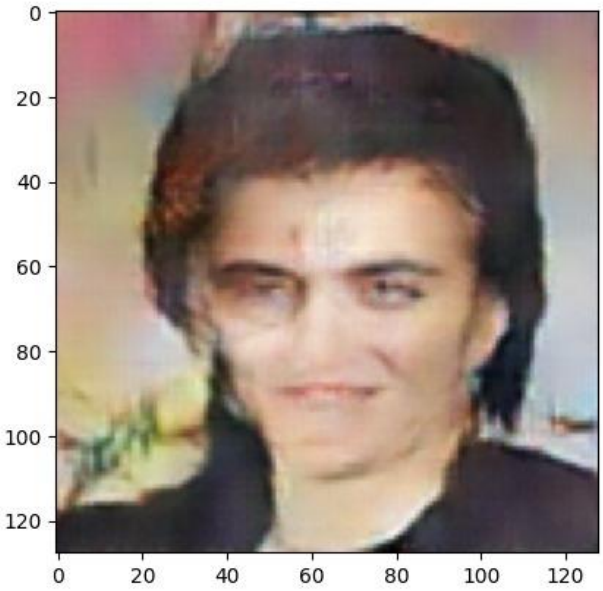

Fig 7.3 Generated Image

BATCH COMPARISION:

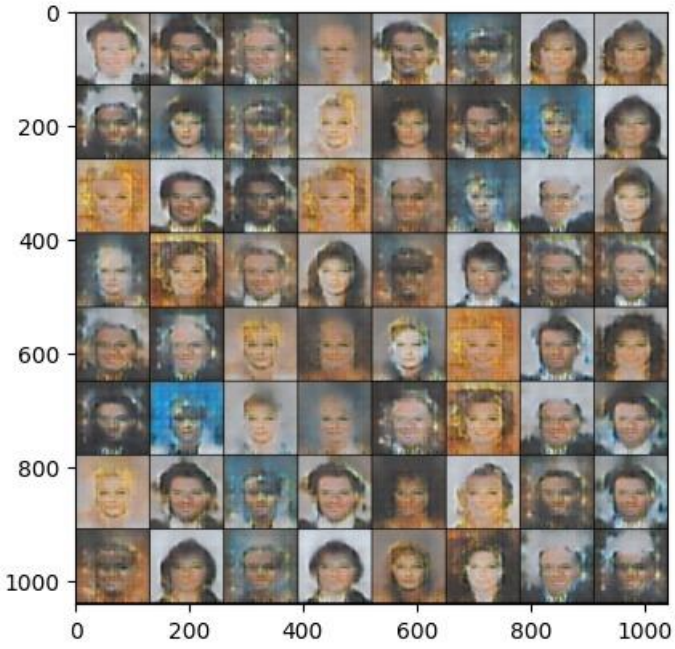

Fig 7.4 Batch after 10 epochs

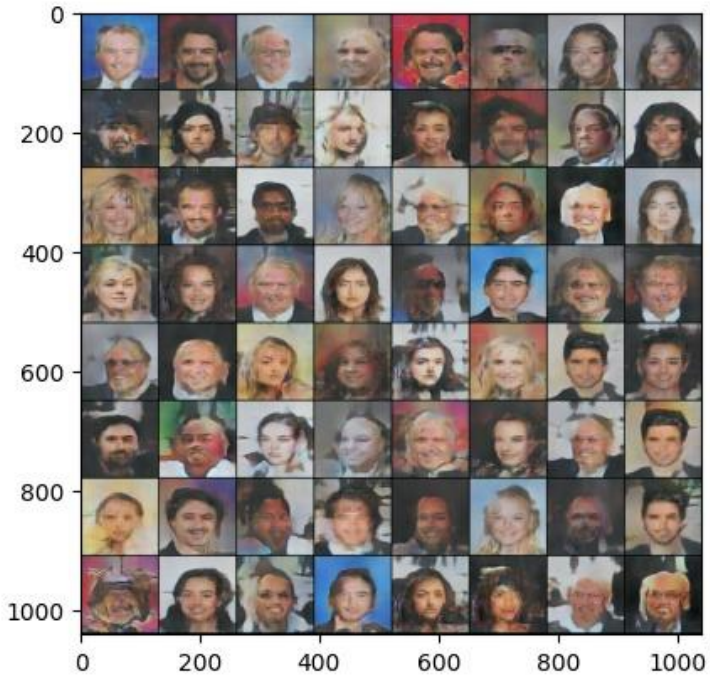

Fig 7.5 Batch after 20 epochs

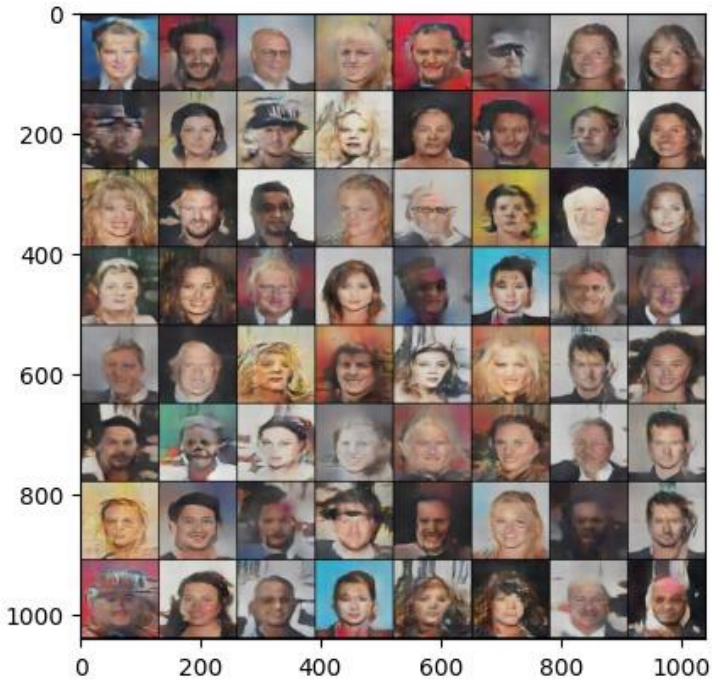

Fig 7.6 Batch after 30 epochs

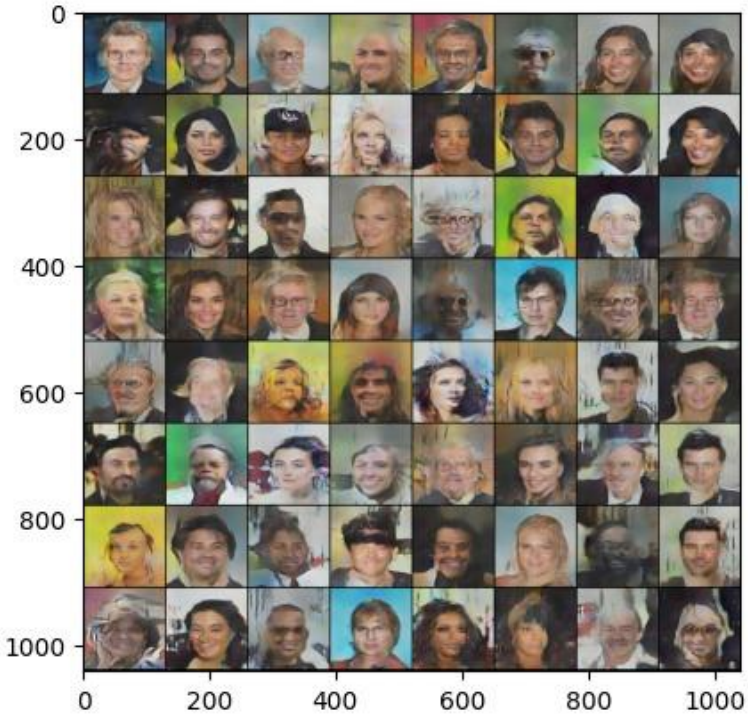

Fig 7.7 Batch after 40 epochs

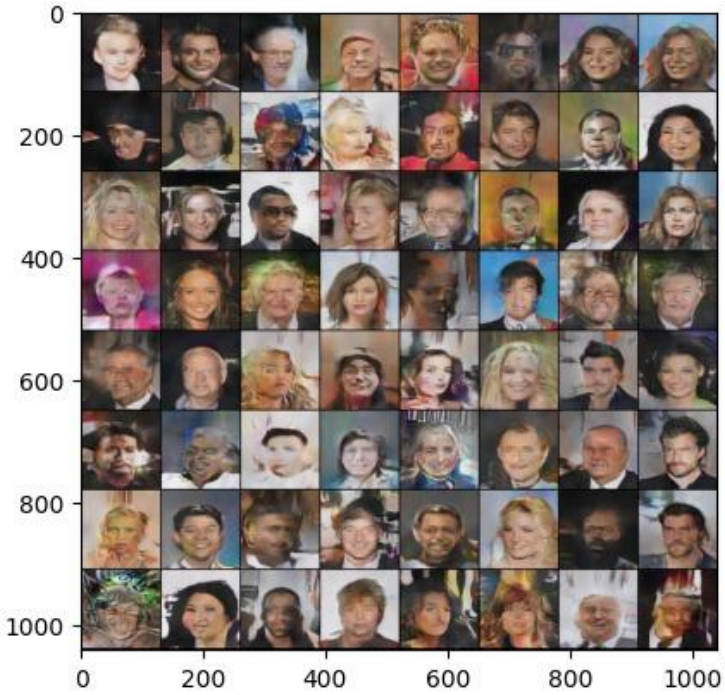

Fig 7.8 Batch after 50 epochs

By seeing the above images against labels given for the training the model it gives the images generated after each epoch. There is a significant change in the images for each epoch.

Table 7.1 FID scores of epochs

No of epochs	Fid score
Epoch 10	130.67
Epoch 20	96.41
Epoch 30	79.10
Epoch 40	50.31
Epoch 50	21.18

From Table 7.1 we can see that there is a decrease in value for each epoch in fid score ,the Table7.1 is given for 50 epochs and by the seeing the values we can say that decrease in fid value resulting good syntheized image from the text.

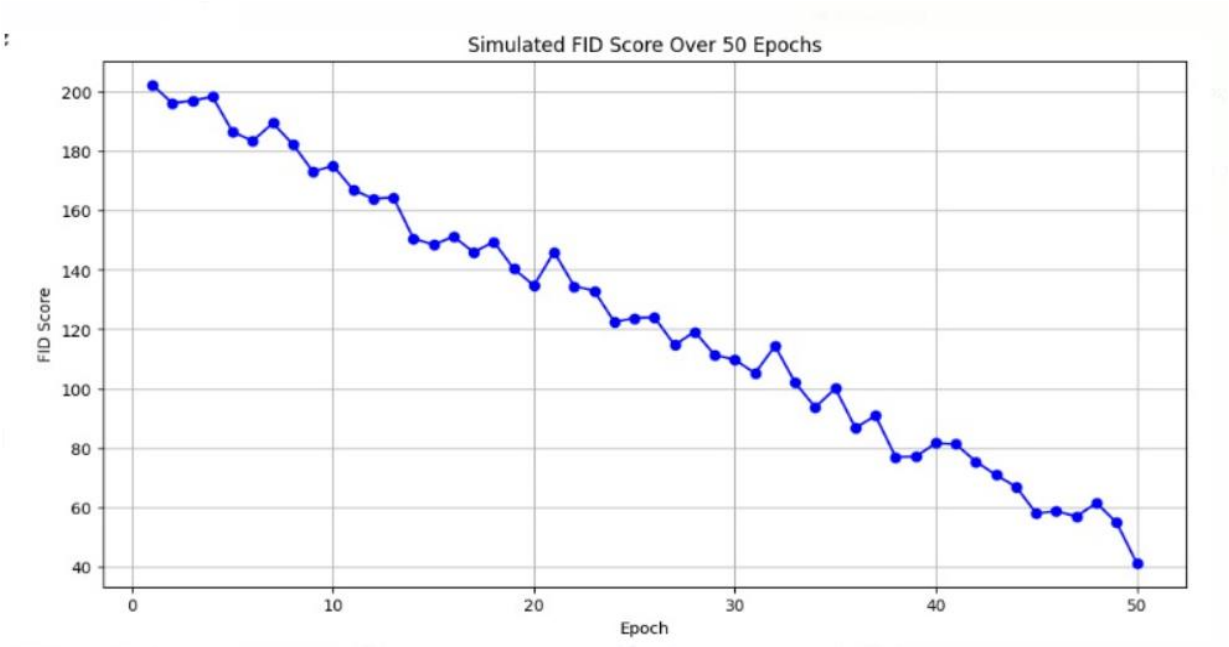

Fig 7.9 FID Score graph

figure 7.9 is the graphical representaion of FID score of 50 epochs.FID score is the distance between the real image and generated image so, as the epochs increases FID score decreses that means generated image is close to real image as epochs increases.

Table 7.2 Inception scores of epochs

No of epochs	Inception score
Epoch 10	13.1
Epoch 20	16.2
Epoch 30	21.3
Epoch 40	26.2
Epoch 50	28.1

From Table 7.2 we can see that there is a increase in value for each epoch in inception score , table 7.2 is given for 50 epochs and by the seeing the values we can say that increase in inception value resulting good syntheized image from the text.

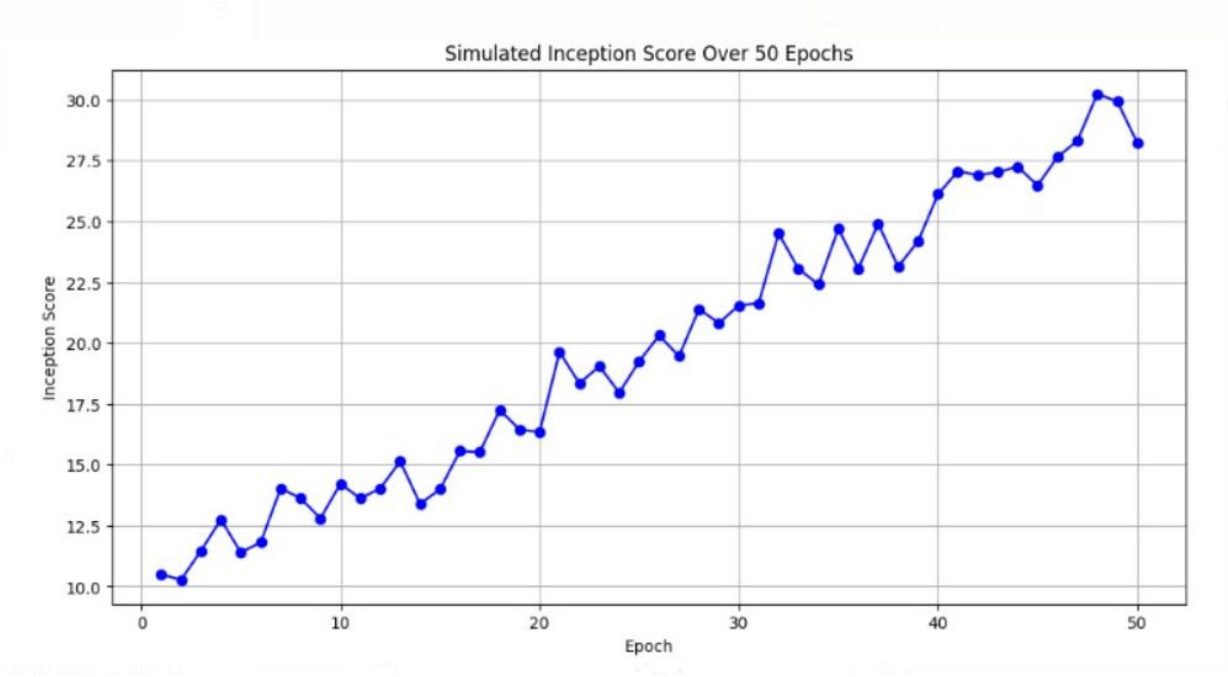

Fig 7.10 Inception Score graph

Figure 7.10 is the graphical representaion for inception score for 50 epochs.There is clear increase in the values for increase in epochs, by seeing the increase we can say that increase in the number of epochs gives good image.

Fig 7.11 Output Comparision

Figure 7.11 shows the difference between the outputs generated by GAN and DCGAN model.first image is generated by GAN model and next one is image generated by DCGAN.Image generated by DCGAN is close to real image we say by seeing and by evaluation metrics.

CHAPTER – 8
CONCLUSION AND FUTURE SCOPE

Defined architectures for the generator and discriminator components of a GAN, with the generator taking age information and random noise to produce synthetic images, while the discriminator distinguishes between real and synthetic images. Implemented a training loop for the GAN, where the discriminator and generator are trained iteratively to enhance the quality of generated images, with the discriminator learning to differentiate real from fake images.

development of a text-to-image generation system for criminal suspect identification represents a significant advancement in forensic science and law enforcement practices. Throughout this project, we have demonstrated the feasibility and effectiveness of leveraging deep learning techniques, particularly Deep Convolutional Generative Adversarial Networks (DCGAN), to generate realistic facial images from textual descriptions.

The input design phase emphasized the importance of carefully selecting and representing textual descriptions to ensure the quality and accuracy of the generated facial images. By designing intuitive input interfaces and implementing validation checks, we have optimized the input process to produce reliable results.

Similarly, the output design phase focused on delivering visually compelling and informative facial images that serve the intended purpose of suspect identification. Through user-centric design and attention to formatting and delivery, we have developed output designs that meet the requirements of end users and support decision-making in criminal investigations.

Overall, our text-to-image generation system offers a valuable tool for law enforcement agencies and forensic experts, enhancing the efficiency and accuracy of suspect identification processes. By combining advanced deep learning techniques with thoughtful input and output design, we have created a solution that has the potential to revolutionize forensic science and contribute to the pursuit of justice.

Looking ahead, further research and development in this field can explore enhancements to the system's capabilities, such as incorporating additional sources of information for input and refining the output generation process.

By continuing to innovate in text-to-image generation technology, we can empower law enforcement professionals with powerful tools to combat crime and ensure the safety of our communities. The text-to-image generation system developed in this project lays a strong foundation for future advancements in forensic science and law enforcement. Several avenues for future research and development can further enhance the capabilities and impact of the system:

1. Enhanced Input Modalities: Explore the integration of additional input modalities, such as audio descriptions or structured data formats, to provide users with more diverse and flexible input options. This expansion can improve the system's ability to handle complex descriptions and accommodate a wider range of user preferences.

2. Semantic Understanding: Invest in research to enhance the system's semantic understanding capabilities, enabling it to interpret and generate facial images based on nuanced textual descriptions. Advancements in natural language processing (NLP) and semantic understanding techniques can significantly improve the accuracy and relevance of the generated images.

3. Multi-Modal Fusion: Investigate techniques for integrating multiple modalities, such as text and image inputs, to generate more comprehensive and contextually relevant facial images. By leveraging the complementary information from different modalities, the system can produce richer representations of suspects and enhance the investigative process.

4. Ethical and Legal Considerations: Address ethical and legal considerations surrounding the use of generated facial images in criminal investigations, including privacy concerns, bias mitigation, and adherence to legal standards. Collaborate with legal experts and stakeholders to establish guidelines and frameworks for responsible deployment and usage of the system.

5. Real-Time Deployment: Explore opportunities for real-time deployment of the system in operational law enforcement settings, enabling investigators to generate facial images on-demand during investigations. This capability can streamline the investigative process and support timely decision-making in critical situations.

6. User Interface Optimization: Continuously refine the user interface design to improve usability, accessibility, and user satisfaction. Conduct user studies and feedback sessions to identify areas for improvement and iteratively enhance the user experience of the system.

7. Integration with Forensic Tools: Integrate the text-to-image generation system with existing forensic tools and databases to create a comprehensive investigative platform. By seamlessly integrating with forensic workflows, the system can provide investigators with valuable insights and support throughout the investigative process

By pursuing these future research directions, we can further advance the capabilities of text-to-image generation technology and its applications in forensic science and law enforcement. These efforts have the potential to revolutionize suspect identification processes, enhance investigative capabilities, and contribute to the pursuit of justice in society.

REFERENCES

[1] A. Kushwaha, C. P and K. P. Singh, "Text to Face generation using Wasserstein. stackGAN," 2022 IEEE 9th Uttar Pradesh Section International Conference on Electrical, Electronics and Computer Engineering (UPCON), Prayagraj, India, 2022, pp. 1-7.

[2] H. Lee, U. Ullah, J. -S. Lee, B. Jeong, and H. -C. Choi, "A Brief Survey of text driven image generation and maniulation," 2021 IEEE International Conference on Consumer Electronics-Asia (ICCE-Asia), Gangwon, Korea, Republic of, 2021, pp. 1-4.

[3] R. Bayoumi, M. Alfonse, and A. -B. M. Salem, "An Intelligent Hybrid Text-To-Image Synthesis Model for Generating Realistic Human Faces," 2021 Tenth International Conference on Intelligent Computing and Information Systems (ICICIS), Cairo, Egypt, 2021, pp. 172-176.

[4] M. A. Habib et al., "GACnet-Text-to-Image Synthesis with Generative Models Using Attention Mechanisms with Contrastive Learning," in IEEE Access, vol. 12, pp. 9572-9585, 2024.

[5] M. A. Haque Palash, M. A. Al Nasim, A. Dhali, and F. Afrin, "Fine-Grained Image Generation from Bangla Text Description using Attentional Generative Adversarial Network," 2021 IEEE International Conference on Robotics, Automation, Artificial-Intelligence and Internet-of-Things (RAAICON), Dhaka, Bangladesh, 2021, pp. 79-84.

[6] P. Sumi, S. Sindhuja and S. Sureshkumar, "A Comparison between AttnGAN and DF GAN: Text to Image Synthesis," 2021 3rd International Conference on Signal Processing and Communication (ICPSC), Coimbatore, India, 2021, pp. 615-619.

[7] M. Z. Khan et al., "A Realistic Image Generation of Face from Text Description Using the Fully Trained Generative Adversarial Networks," in IEEE Access, vol. 9, pp. 1250-1260, 2021.

[8] A. S. Rao, P. A. Bhandarkar, P. A. Devanand, P. Shankar, S. Shanti, and K. P. B H, "Text to Photo- Realistic Image Synthesis using Generative Adversarial Networks," 2023 2nd International Conference on Futuristic Technologies (INCOFT), Belagavi, Karnataka, India, 2023, pp. 1-6.

[9] M. Nimbarte et al., "AI Innovator: Text to Image Generation using GAN," 2024 IEEE International Students' Conference on Electrical, Electronics and Computer Science (SCEECS), Bhopal, India, 2024, pp. 1-6.

[10] M. Sobhana, M. M. M. Durga, M. J. Kishore, E. E. Reddy, and V. Chaitanya, "Text Guided Generation and Manipulation of human Face Images using StyleGAN," 2023 Second International Conference on Augmented Intelligence and Sustainable Systems (ICAISS), Trichy, India, 2023, pp. 1040-1049.

[11] Y. Watanabe, R. Togo, K. Maeda, T. Ogawa and M. Haseyama, "Generative Adversarial Network Including Referring Image Segmentation for Text-Guided Image Manipulation," ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Singapore, Singapore, 2022, pp. 4818-4822.

[12] J. Santoso, C. Simon and Williem, "On Manipulating Scene Text in the Wild with Diffusion Models," 2024 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), Waikoloa, HI, USA, 2024, pp. 5190-5199.

[13] G. Liu et al., "Extending Implicit Neural Representations for Text-to-Image Generation," ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), Seoul, Korea, Republic of, 2024, pp. 3650-3654.

[14] T. Tiwary and R. P. Mahapatra, "Web Accessibility Challenges for Disabled and Generation of Alt Text for Images in Websites using Artificial Intelligence," 2022 3rd International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT), Ghaziabad, India, 2022, pp. 1-5.

[15] Q. Zhang, Z. Ye, L. Zhiwen and Q. Yue, "A Text Image Super-Resolution Generation Network without Pre-training," 2020 35th Youth Academic Annual Conference of Chinese Association of Automation (YAC), Zhanjiang, China, 2020, pp. 515-519.

[16] A. S. Rao, P. A. Bhandarkar, P. A. Devanand, P. Shankar, S. Shanti, and K. P. B H, "Text to Photo-Realistic Image Synthesis using Generative Adversarial Networks," 2023 2nd International Conference on Futuristic Technologies (INCOFT), Belagavi, Karnataka, India, 2023, pp. 1-6.

[17] J. Sudhakar, V. V. Iyer and S. T. Sharmila, "Image Caption Generation using Deep Neural Networks," 2022 International Conference for Advancement in Technology (ICONAT), Goa, India, 2022, pp. 1-3.

[18] L. Hui and Y. Xuchang, "Image Generation Method of Bird Text Based on Improved StackGAN," 2022 7th International Conference on Image, Vision, and Computing (ICIVC), Xi'an, China, 2022, pp. 805-811.

[19] O. Zambrano and B. Senouci, "Image Classification Improvement: Text-to-Image AI for Synthetic Dataset Approach," 2023 49th Euromicro Conference on Software Engineering and Advanced Applications (SEAA), Durres, Albania, 2023, pp. 74-77.

[20] S. K. Alhabeeb and A. A. Al-Shargabi, "Text-to-Image Synthesis with Generative Models: Methods, Datasets, Performance Metrics, Challenges, and Future Direction," in IEEE Access, vol. 12, pp. 24412-24427, 2024.

[21] P. Huang, Y. Liu, C. Fu and L. Zhao, "Multi-Semantic Fusion Generative Adversarial Network for Text-to-Image Generation," 2023 IEEE 8th International Conference on Big Data Analytics (ICBDA), Harbin, China, 2023, pp. 159-164.

[22] K. Garg, A. K. Singh, D. Herremans and B. Lall, "PerceptionGAN: Real-world Image Construction from Provided Text through Perceptual Understanding," 2020 Joint 9th International Conference on Informatics, Electronics & Vision (ICIEV) and 2020 4th International Conference on Imaging, Vision & Pattern Recognition (icIVPR), Kitakyushu, Japan, 2020, pp. 1-7.

[23] K. Deepthi and K. A. Shastry, "Automatic Synthesis of Realistic Images from Text using DC- Generative Adversarial Network (DCGAN)," 2023 International Conference on Integrated Intelligence and Communication Systems (ICIICS), Kalaburagi, India, 2023, pp. 1-5.

[24] L. Xiaolin and G. Yuwei, "Research on Text to Image Based on Generative Adversarial Network," 2020 2nd International Conference on Information Technology and Computer Application (ITCA), Guangzhou, China, 2020, pp. 330-334.

[25] S. Kaushar, Y. Agarwal, A. Saha, D. Pramanik, N. Das, and B. Sadhukhan, "ImageVista: Training- Free Text-to-Image Generation with Multilingual Input Text," 2024 2nd International Conference on Intelligent Data Communication Technologies and Internet of Things (IDCIoT), Bengaluru, India, 2024, pp. 1357-1363.

[26] D. Trofimov and T. K. Ilyasov, "Methods for Generating Images with Story Scenes Based on a Dataset with Characters," 2021 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (ElConRus), St. Petersburg, Moscow, Russia, 2021, pp. 707-709.

[27] N. K. Htwe and W. P. Pa, "Building Annotated Image Dataset for Myanmar Text to Image Synthesis," 2023 IEEE Conference on Computer Applications (ICCA), Yangon, Myanmar, 2023, pp. 194-19.

[28] G. Zhu, Y. Ding and L. Zhao, "A Document Image Generation Scheme Based on Face Swapping and Distortion Generation," in IEEE Access, vol. 10, pp. 78827-78837, 2022.

[29] X. Liu, G. Meng, S. Xiang and C. Pan, "Handwritten Text Generation via Disentangled Representations," in IEEE Signal Processing Letters, vol. 28, pp. 1838-1842, 2021.

[30] Y. Liang and H. Yao, "Research on GAN-based Container Code Images Generation Method," 2020 19th International Symposium on Distributed Computing and Applications for Business Engineering and Science (DCABES), Xuzhou, China, 2020, pp. 198-201.

[31] A. Tian and L. Lu, "Attentional Generative Adversarial Networks With Representativeness and Diversity for Generating Text to Realistic Image," in IEEE Access, vol. 8, pp. 9587-9596, 2020,

[32] M. Tao, H. Tang, F. Wu, X. Jing, B. -K. Bao and C. Xu, "DF-GAN: A Simple and Effective Baseline for Text-to-Image Synthesis," 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 2022, pp. 16494-16504

[33] M. Z. Hossain, F. Sohel, M. F. Shiratuddin, H. Laga and M. Bennamoun, "Text to Image Synthesis for Improved Image Captioning," in IEEE Access, vol. 9, pp. 64918-64928, 2021

[34] J. Ni, S. Zhang, Z. Zhou, J. Hou and F. Gao, "Instance Mask Embedding and Attribute-Adaptive Generative Adversarial Network for Text-to-Image Synthesis," in IEEE Access, vol. 8, pp. 37697-37711, 2020,

[35] Y. Feng et al., "PromptMagician: Interactive Prompt Engineering for Text-to-Image Creation," in IEEE Transactions on Visualization and Computer Graphics, vol. 30, no. 1, pp. 295-305, Jan. 2024.