University of Kerbala
College of Computer Science & Information Technology
Computer Science Department

Text - Conditioned Image Generation using Diffusion Models

A Thesis

Submitted to the Council of the College of Computer Science & Information Technology / University of Kerbala in Partial Fulfillment of the Requirements for the Master Degree in Computer Science

Written by
Sara Faez Abdulghani

Supervised by
Asst. Prof. Dr. Ashwan Anwer Abdulmunem

2024 A.D. 1445 A.H.

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

(فَتَعَالَى اللَّهُ الْمَلِكُ الْحَقُّ ۗ وَلَا تَعْجَلْ بِالْقُرْآنِ مِن قَبْلِ
أَن يُقْضَىٰ إِلَيْكَ وَحْيُهُ ۖ وَقُل رَّبِّ زِدْنِي عِلْمًا)

صَدَقَ اللهُ العَلِيُّ العَظِيمْ

طه :[114]

Supervisor Certification

I certify that the thesis entitled (**Text - Conditioned Image Generation using Diffusion Models**) was prepared under my supervision at the Department of Computer Science / College of Computer Science & Information Technology / University of Kerbala as partial fulfillment of the requirements of the degree of Master in Computer Science.

Signature:
Supervisor Name: Assist. Prof. Dr. Ashwan Anwer Abdulmunem
Date: 10 / 10/2024

The Head of the Department Certification

In view of the available recommendations, I forward the thesis entitled "**Text - Conditioned Image Generation using Diffusion Models**" for debate by the examination committee.

Signature:
Assist. Prof. Dr. Muhannad Kamil Abdulhameed
Head of Computer Science Department
Date: 10 /10 / 2024

Certification of the Examination Committee

We hereby certify that we have studied the dissertation entitled (**Text - Conditioned Image Generation using Diffusion Models**) presented by the student (**Sara Faez Abdulghani**) and examined her/him in its content and what is related to it, and that, in our opinion, it is adequate with (Excellent) standing as a thesis for the Master degree in Computer Science.

Signature:
Name: Dr. Baheeja Khudair Shukur
Title: Prof.
Date: 9 /10 / 2024
(Chairman)

Signature:
Name: Dr. Elham Mohammed Thabit
Title: Assistant Prof
Date: 9 / 10/ 2024
(Member)

Signature:
Name: Dr. Ihsan Ali Kareem
Title: Assistant Prof
Date: 9 /10 / 2024
(Member)

Signature:
Name: Dr. Ashwan Anwer Abdulmunem
Title: Assistant Prof.
Date: 10 / 10/ 2024
(Member and Supervisor)

Approved by the Dean of the College of Computer Science & Information Technology, Universi
of Kerbala.

Signature: Mowafak
Assist. Prof. Dr. Mowafak Khadom Mohsen
Date: ١٠ /١٠ / 2024
(Dean of College of Computer Science & Information Technology)

Dedication

This work is lovingly dedicated to my parents and my brother for their unwavering love and support throughout all the challenging moments of my academic journey and, whose unwavering support and encouragement have been my guiding light. Your belief in me has given me the strength to persevere through challenges and strive for excellence. I would like to thank my uncle, Professor Dr. Adel Al-Yasiri, for his invaluable advice and ongoing guidance. I would also like to extend my deepest gratitude to all my friends who generously shared their time and expertise to assist me in this research journey. Thank you for being a part of my life and helping me reach this milestone.

Acknowledgement

I would like to express my deep appreciation to my supervisor, [**Assist Prof. Dr. Ashwan Anwer Abdulmunem**], for her invaluable support and guidance throughout my research. Her knowledge and sharp observations have been crucial in enhancing the quality of my work, offering clear direction every step of the way. I would also like to thank [College of Computer Science & Information Technology] and [Department of Computer Science], represented by the Dean of the College, the Assistant Deans, and the Head of the Department, for providing the ideal academic environment and necessary support to complete this research. I must also express my deep appreciation and gratitude to all my professors in the department, who generously shared their knowledge and expertise with me and contributed to the development of my research and academic skills.

Abstract

Text-guided synthesis of images has made a giant leap toward becoming a mainstream phenomenon. With text-to-image generation systems, anybody can create digital images and artwork. This provokes the question of whether text-to-image generation is creative.

The generative systems have contributed much to the development of artificial intelligence (AI) generating rather realistic images from the text. Text-to-image generation systems have been used in various forms and areas in scope including, but not limited to, artworks and designs, data sampling, and entertainment. Many studies have been conducted on generating images from text and many AI techniques have been proposed. However, some critical issues have yet to be solved, especially with regard to the time consumption and the training time. Therefore, the proposed study utilized the Stable Diffusion Model (SDM) to conduct iterative feedback (if the metrics of the evaluation namely Inception Score (IS) and Fréchet inception distance (FID) do not improve then the hyper-parameters are tuned and the model is trained again). In this study, the fine-tuning of the SDM results in a considerable improvement in generating images that are more akin to reality. As well, there are trade-offs between image quality and flexibility in performance metrics. The fine-tuning process gradually improves the model's global ability to generate better and more diverse digital imagery. The fine-tuned model has a lower FID score (248.748256), suggesting a higher likelihood of attaining higher image distribution similarity to the targeted dataset. Sparingly, the results of the improved model denoted a lower FID score (212.52) when contrasted with the base model (251.22), pointing out that the generated images from the fine-tuned model were more intimate to the target distribution in the synthetic dataset.

Declaration Associated with this Thesis

Some of the works presented in this thesis have been published or accepted as listed below.

1- S. F. Abdulghanni and A. A. Abdulmunem, "Image Generation Conditioned on Text Using Deep Learning Models: Survey," 2023 Al-Sadiq International Conference on Communication and Information Technology (AICCIT), Al-Muthana, Iraq, 2023, pp. 171-175, doi: 10.1109/AICCIT57614.2023.10218041.

2- Sara Faez Abdulghanni and Ashwan A. Abdulmunem, " An Improved Image Generation Conditioned on Text Using Stable Diffusion Model" Journal of Al-Qadisiyah University for Computer Science and Mathematics, 2024, ISSN:2521-3504(online), ISSN: 2074-0204(print), DOI: 10.29304/jqcm .

Table of Contents

Dedication………………………………………………………………………...……...iii
Acknowledgement…………………………………………………………….……........iv

Abstract…………………………………………………………………………….…..v

Declaration Associated with this Thesis……………………………………………...….vi
Table of Contents…………………………………………………..…………..…..vii
List of Tables………………………………………………………….………….…VII
List of Figures…………………………………………………………………...…..x

List of Abbreviations……………………………………………………………….…xi

CHAPTER ONE: INTRODUCTION………………………………………….....XII

1.1 Overview ………………………………………………………………….…….1

1.2 Problem Statement…………………………………………………………........2

1.3 Research Objectives…………………………………………………….….….3

1.4 Research Questions…………………………………………………….….….3

1.5 Challenges……………………………………………………………………….3

1.6 Related Work……………………………………………………………………….4

1.6.1 Generative adversarial networks(GAN)…………………………….….……...5

1.6.2 Variational Autoencoder……………….………………………….……………...9

1.6.3 Diffusion Model………………………….………………………….……….13

CHAPTER TWO: THEORETICAL BACKGROUD……………….………….……19

2.1 Overview……………………………………………………………………….20

2.2 Introduction………………………………………………………………....….20

2.3 Text To Images…………………………………………………………….……22

2.4 Diffusion Models Types…………………………………………………….…..23

2.4.1 Dall E2 Model…………………………………………………………….…..26

2.4.2 Imagen Model………………………………………………………………...27

2.4.3 Stable Diffusion Model…………………………………………………….27

2.4.3.1 U-Net Architecture……………………………………..………...29

2.4.3.2 Varitional Autoencoder………………………………………......32

2.4.3.3 Clip Model……….…………………………………………………34

2.5 Evaluation Metrics……………………………………………………………...35

2.5.1 Inception- V3 Network…………………………………………………..…..37

2.5.2 Model Assessment Tools……………………………………………………..38

CHAPTER THREE: PROPOSED METHOD……………………...…………….40

3.1 Introduction……………………………………………………………..…..…..41

3.2 Design of the proposed method…………………………………..………….…43

3.2.1 Dataset Preparation…………………………………………………….….…47

3.2.2 Training The Proposed Method……………………………………….….…..47

3.2.2.1 Hyperparameters Setting………………………..………..……....48

3.2.2.2 Models Preparation…………………………………………....49

3.2.2.3 Training Loops……………………………………………….…..49

3.2.3 Testing The Proposed Model………………………………………….…...52

3.2.4 Evaluating the results of The Proposed Model……………………….…53

3.3 Summary……………………………………………………….………………54

CHAPTER FOUR: RESULTS AND DISCUSSION……...……………………...55

4.1 Introduction………………………………………………………………......56

4.2 Dataset……………………………………..………………………………….…...56

4.3 Hardware Requirements…………………………..…………………………….....59

4.4 Results and discussion…………………………..……………………………….…...60

4.4.1 Experiment 1: Fine-Tuning Performance Across Epochs………..………......63

4.4.1.1 Key Observations of Experiment 1……………………….…..….65

4.4.2 Experiment 2: Comparison of Fine-Tuned Model with Base Model………...65

4.4.2.1 Key Observations for Experiment 2…………………………..….67

CHAPTER FIVE: CONCLUSION AND FUTURE WORKS……………….….68

5.1 Overview……………………………………………………………………………..69

5.2 Conclusion…………………………………………….………………………….…..69

5.3 Future Works …………………………………………………………………...70

REFRENCES………………………………………………………………….…71

List of Tables

Table 1.1**:** Summary Of Previous Work About Image Generation From Text………....18

Table 4.1 : Text-to-image generation datasets………………………………………..57

Table 4.2: Samples Of Images And The First Description Associated With The Image From Dataset……………………………………………………………..……..58

Table 4.3: Specifications of the Computational Resources Used for Training the Stable Diffusion Model…………………………………………………………………....60

Table 4.4: Results of the proposed model…………….…………………………….....61

Table 4.5: Fine-Tuned Stable Diffusion Model Performance Across Epochs………......63

Table 4.6 Comparison of Fine-Tuned and Base Models……………………………......65

List of Figures

Figure 1.1 : Deep Learning Techniques For Image Generation From Text………………5

Figure 1.2: Three Methods Of Utilizing Generative Adversarial Networks For Image Synthesis……………………………………………………………………………..6

Figure 1.3: Diagram of Autoencoder Structure…...……………………………………..11

Figure 2.1: U-Net architecture diagram……………………………………………….30

Figure 2.2: The architecture of the CLIP model………………………………………....35

Figure 3.1: Overview of the Stable Diffusion Model Pipeline for Generating Images from Textual Descriptions………………………………………………………………..…42

Figure 3.2: Methodology Flowchart for Stable Diffusion Model with Iterative Refinement……………………………………………………………………….…44

List of Abbreviations

Abbreviation	Description
CAD	Computer Aided Design
CV	Computer Vision
DBN	Deep Belief Networks
DM	Diffusion Model
DDPM	The denoising diffusion probabilistic model
DCGAN	Deep Convolutional Generative Adversarial Network
ELBO	Evidence Lower Bound
FDP	Forword Diffusion Process
FDM	Forward Diffusion Model
FID	Fr´echet Inception Distance
GANs	Generative Adversarial Networks
IS	Inception Score
MSE	Mean Squared Error
LDMs	Latent Diffusion Models
PDF	Probability Density Function
RBM	Restricted Boltzmann Machines
RGB	Red, Green and Blue
SGM	Score-Based Generative Models
SBM	Score Based Model
VAE	Variational Autoencoder

CHAPTER ONE

INTRODUCTION

1.1 Overview

Artificial intelligence is a field of science that deals with assisting machines find solutions to complex problems in a more human-like fashion, compared with the natural intelligence of humans, is usually defined as the science and engineering of imitating, extending, and augmenting human intelligence through artificial means and techniques to make intelligent machines. Machine learning is a subfield of AI that studies the ability to improve performance based on experience. Deep learning is a subfield of machine learning that involves the encoding of hypotheses, employing highly complex algebraic circuits that include connection weights that can be adjusted. The ''deep'' in deep learning refers to the many layers involved in these circuits, making the operations a bit complex. Applications of deep learning are found in almost every field; for example in identifying objects in an image, or translating text from one language to another without human intervention, voice recognition, and image creation [1]. Deep artificial neural networks (ANN) have shown exceptional performance in various applications like object identification, speech recognition, and super-resolution. These networks use multiple layers of artificial neurons to extract advanced features from diverse data distributions. Generative modeling, a domain of machine learning, aims to understand the fundamental data distributions responsible for data generation, enabling generalization across diverse datasets and addressing data sparsity. Deep neural networks are crucial in generative modeling, learning high-dimensional distributions from extensive datasets. Before 2014, generative models employed deep learning architectures like Restricted Boltzmann Machines (RBM) and Deep Belief Networks (DBN). While this yielded favorable outcomes in the field, many difficulties emerged throughout the training process [2]. There are greater and higher generative models that enable the conversion of textual descriptions into artwork and other simple graphics. These are coherent translation systems that can generate detailed images from prompts formulated in natural language [3]. The process of generating images from the text entails various tasks that the model has to complete. Another key purpose is found in the information representation that is needed to define and isolate the shapes or forms, color, and position in relation to the pixel. It also has to make concepts of the input text and correctly map objects and attributes to the relevant words and phrases. In addition, it must be capable of creating detailed distributions that are critical in developing images that contain interpenetrating components [4]. The divergence in performance between diffusion models and GANs can be attributed to two main reasons: firstly, the model designs employed in contemporary GAN studies have undergone extensive development and optimization; secondly, GANs can prioritize fidelity over variety, yielding samples of superior quality at the expense of comprehensive distribution coverage. The visual output from diffusion models is extremely good, and they only require a single U-Net design for training, in contrast to the dual-network setup necessary for GANs, which results in a more consistent training experience [5].

1.2 Problem statement

The stable diffusion model used in this study for image generation task from texts , deals with models that are responsible for many operations such as tokenizing the texts and for adding noise, and denoising the images, so it needs high requirements for resources and takes a long training time. The main problem to solve in this study is to get high-quality images while using a low level of resources and reducing the training time to get results that are significantly better than the original model based on the evaluation metrics.

1.3 Research objectives

1. To investigate the existing research problems related to text-to-image generations and the techniques used to resolve these issues.
2. To utilize hyperparameters that play a crucial role in the training process, influencing the convergence rate, and accuracy of the proposed Stable Diffusion Model.
3. To evaluate the effectiveness of the proposed model using Inception Score and Frechet Inception Distance evaluation metrics.

1.4 Research Questions

To prove the research hypothesis, the following questions are addressed:

1. What are the limitations of the existing solutions regarding text-to-image generation systems?
2. How could the proposed Stable Diffusion Model (SDM) be improved?
3. How can the performance of the proposed SDM be evaluated?

1.5 Challenges

In this study, A Stable Diffusion Model SDM designed to generate high-quality images of flowers from textual descriptions. The proposed model aims to address key challenges in traditional text-to-image models, such as high computational costs and extensive training times while maintaining or improving image quality.

Although the model can create sensible images, they might not be very realistic and hence the applicability in realistic image distribution scenarios might be harmed. because of memory issues, the batch size was set to 2, which may have affected the convergence speed and the model’s capability of generalizing from the training data. The number of training epochs was limited to 12; though the model did not get enough epochs to converge and achieve the optimum results. The training process was carried out on an NVIDIA T4 GPU with a high RAM configuration; however, such a configuration has restrictions regarding the maximal model size and batch sizes. The use of IS and FID as the evaluation criteria can be considered conventional, and though effective, it does not address the qualitative nature of the generated images as much as it should.

1.6 Related work

This section provides an overview of previous studies on the generation of images from text, including notable developments in deep learning techniques like Google's Deep Dream in 2015 and Generative Adversarial Networks (GANs), which were announced in 2014. The attempts to teach machines image generation from texts can be traced to the early times of deep generative models when Mansimov et al. added text information to DRAW[4]. On the other hand, since its January 2021 launch, OpenAI's CLIP has greatly advanced text-to-image generation [3]. Initially, the Deep Convolutional Generative Adversarial Network (DCGAN) model was introduced as a method for achieving advantageous results in the text-to-image synthesis sector [6]. Several approaches have been developed to handle the difficulty of creating images from text and will be examined:

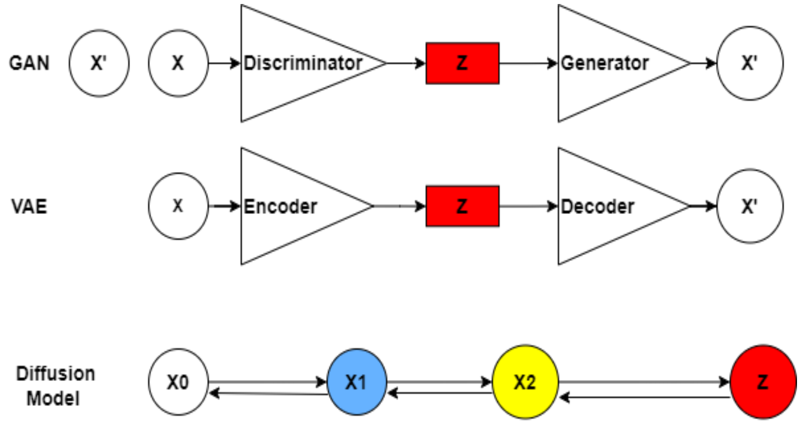


Figure (1.1) : Deep Learning Techniques For Image Generation From Text[7]

1.6.1 Generative adversarial networks(GAN)

The capacity of Generative Adversarial Networks (GANs) to grasp intricate data distributions across many dimensions has garnered significant attention from the research community. From their inception by Ian Goodfellow in 2016 up until March 2022, they’ve seen increasing application and interest. The primary notion underlying GANs can be compared to that of "art forgery," in which artworks are made and misleadingly attributed to other, often more recognized, artists. GANs entail training two neural networks concurrently: the generator G(Z) makes false data, while the discriminator D(Y) analyzes its authenticity, effectively distinguishing genuine artworks from imitations. Through input Y, such as a image, the discriminator assigns a grade near to 1 for "real" or close to 0 for "fake." Meanwhile, G(Z) derives its outputs from random noise Z, hoping to trick D into accepting them as genuine. Training D(Y) entails maximizing its score for genuine images from the original dataset while reducing it for created images. As a result, G and D are always competing, giving rise to the term adversarial training. We alternate the training of G (Generator) and D (Discriminator), maximizing their objectives using loss functions and gradient descent. The generator improves its ability to produce counterfeits, while the discriminator improves its capacity to identify them. Typically, the discriminator uses a typical convolutional neural network to determine if a image is real or not. A major innovation is the use of backpropagation across both networks, which allows modifications to the generator's settings to better fool the discriminator [7]. GANs hold an advantage in producing data akin to the original as they do not rely on predefined probability density assumptions [2]. The techniques employed in image creation through GANs highlight both the advantages and limitations of existing methodologies. We categorize the principal strategies used for synthesizing images into three types: direct, hierarchical, and iterative methods, denoting the triad of techniques for generating images using GANs [8].

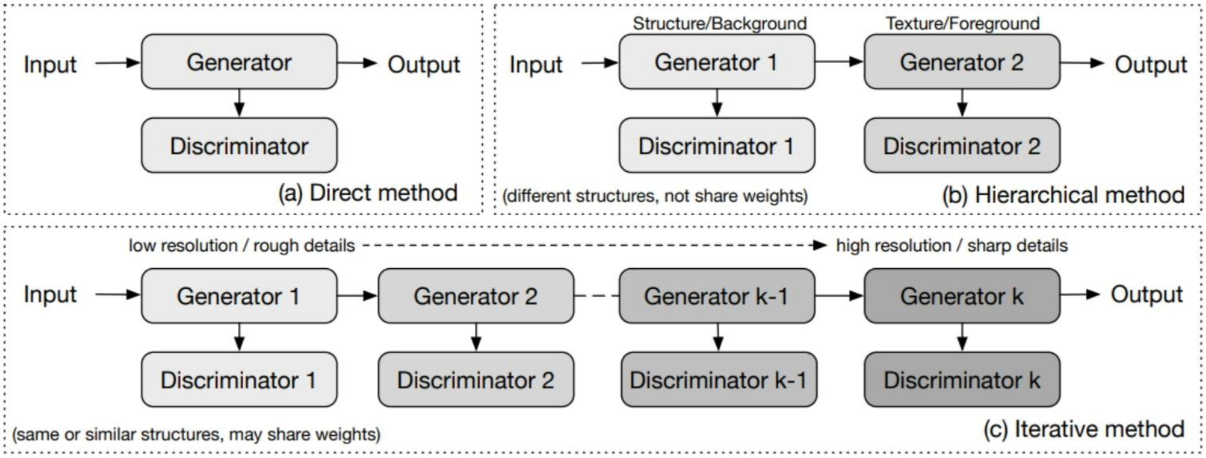


Figure 1.2: Methods Of Utilizing Generative Adversarial Networks For Image Synthesis [8]

The three primary methods for GAN modeling are direct, hierarchical, and iterative. The direct approach involves using a single generator and discriminator, resulting in simpler structures without branching. Early GAN models such as GAN, DCGAN, ImprovedGAN, InfoGAN, f-GAN, and GANINT-CLS fall into this category. This method is relatively straightforward to design and implement compared to hierarchical and iterative methods and usually achieves good results. The algorithms of the Hierarchical Method utilize a pair of generators and discriminators to differentiate images into two categories: "styles & structure" and "foreground & background". The generators have distinct functions and can operate either in parallel or sequentially. According to the SS-GAN, random noise (z-hat) is used to create a surface normal map by a Structure-GAN, and noise (z-tilde) is used as input by a Style-GAN to create an image. The approach used in Iterative Methods differs from Hierarchical Methods in two key ways. Firstly, Instead of utilizing two generators with distinct functions, Iterative Methods use multiple generators with similar or identical structures. These generators produce images by refining details from the previous generator, progressing from coarse to fine. Secondly, weight-sharing is a feature that can be employed among the generators in Iterative Methods, whereas it is typically not allowed in Hierarchical Methods when using the same structures in the generators. An interesting example of this approach is LAPGAN, which employs an iterative process to refine an initial coarse image into a sharper one using a Laplacian pyramid. The method employs multiple generators to produce residual images that are then added to the input image. The only difference in generator structures is the input and output dimensions. However, the lowest-level generator only requires a noise vector as input. StackGAN, functioning as an iterative method, consists of only two layers of generator [9]. As per the method described in [10] the understanding and modeling of a generator's distribution is the main mathematical goal of a GAN. To correctly depict the true distribution, a Probability Density Function (PDF) Pg(x) the understanding and modeling of a generator's distribution is the main mathematical goal of a GAN. To correctly depict the true distribution, a Probability Density Function (PDF) Pg(x). There is research that uses more than one GAN for image generation in addition to a single GAN. The approach in [11] provided a straightforward and effective strategy for creating realistic-looking and high-quality images. The topic under consideration involves the utilization of stacked adversarial generative networks to create text-to-image. This process consists of two distinct stages. Using a random noise vector and a textual description as input, in the first stage, GAN first creates a low-resolution image of an object by defining its basic shape, color, and backdrop layout. In the second stage, the low-resolution image is refined by the GAN using the accompanying textual description to fill in any elements that are lacking. The end product is a high-resolution image that closely mimics a real-life imagegraph. A framework proposed in [12]. helps in generating images from text and offers the user control over the synthetic image using text descriptions. The proposed framework describes a new manageable text-to-image generative adversarial network (ControlGAN) that is capable of creating goodquality images while at the same time permitting the users to change the qualities of certain objects without disturbing the creation of the other content. Using the method mentioned in [13], the author proposed a new technique with the help of a text 10 encoding model that has both RNN and CNN components together with the help of the Generator and Discriminator network. This is a technique that intends to take the textual description of flowers as input and produce a set of images that are different from the ones used in the input description but are semantically similar. T2CI-GAN is presented in two different versions in this study. A standard Generator is used in one version, and a customized Generator is used in the other. The study starts by looking at the basic network topology of the T2CI-GAN models, which are based on the T2I-GAN. To create compressed images from text descriptions, the first model is trained using JPEG-compressed DCT images (in the compressed domain). The second model creates JPEG-compressed DCT representations from text descriptions by training it on RGB images (in the pixel domain). The instability of GANs and their sensitivity to hyper-parameters make the training process challenging. Additionally, the generator experiences mode collapse, which leads it to converge to particular parameter settings and generate a restricted range of samples [14]. Generators can run in parallel, SS-GAN Utilize two networks for the generator and discriminator, these approaches partition an image into two components, namely "styles & structure" and "foreground & background." [8]. Two GANs are used in the proposal: a Structure GAN that uses random noise zˆ to construct a surface normal map, and a Style-GAN that uses noise z、 as well as the generated surface normal map as input to create an image. While the Style-GAN differs slightly from the Structure-GAN, both use the same construction components as DCGAN [15], in Style-Generator, the noise vector and the resulting surface normal map pass through multiple transposed convolutional layers and several convolutional layers, respectively. The final output is a single tensor that passes through the remaining layers of the Style-Generator. To create a single input for the Style-Discriminator, every surface normal map and its matching image are concatenated at the channel dimension. Furthermore, SS-GAN presupposes that reconstructing a decent surface normal map should also be done using a high-quality synthetic image. Using a pixel-wise loss that enforces the reconstructed surface normal to approximate the true one, SS-GAN builds a fully-connected network that converts an image back to its surface normal map based on this assumption. One of the primary drawbacks of SS-GAN is that surface normal map ground-truth necessitates the usage of Kinect [9].

1.6.2 Variational Autoencoder

With the help of GANs, samples that closely resemble the statistical properties of the training data {xi} are produced. By comparison, variational auto-encoders, or VAEs, are generative models with a probabilistic approach that seeks to learn a distribution Pr(x) over the data. Once trained, VAEs can produce new samples from this distribution. However, due to the nature of VAEs, it is not feasible to precisely assess the probability of new examples, denoted as x*. It is important to clarify that the VAE is often discussed as if it represents the model of P r(x), but it is in fact a neural architecture designed to facilitate the learning of the model for P r(x). The final model for P r(x) does not include the "variational" or "autoencoder" components and might be more accurately termed as a nonlinear latent variable model. Variational autoencoders have diverse applications, such as denoising, anomaly detection, and compression. In terms of generation, VAEs construct a probabilistic model, making it straightforward to sample from this model by drawing [16]. I’d like to highlight the following points:

In the process of generating data, we first utilize the previous probability distribution over the latent variable, denoted as P r(z). Subsequently, we pass this outcome through the decoder f[z, ϕ] and introduce noise. Choosing to sample from the aggregated posterior—which is the average posterior over all samples and is a mixture of Gaussians that is more representative of true distribution in latent space—while taking into account the naive spherical Gaussian noise model and the Gaussian models used for both the prior and variational posterior instead of the prior—is an efficient strategy for improving the quality of generation :

$\boldsymbol{q(Z|\theta) = (1/I)\ P\ P}$**iq(z|Xi,**$\boldsymbol{\theta)r}$ **………..(1)**

High-quality sample generation is achievable with modern VAEs, but only with the application of hierarchical priors, specialized network architecture, and regularization strategies. An architecture called VAE makes it easier to learn a nonlinear latent variable model over x. By taking a sample from the latent variable, processing the output through a deep network, and then applying independent Gaussian noise, this model can generate new examples [16]. An autoencoder is a specific type of neural network designed to encode and decode data to produce an output that closely matches the original input. Importantly, it can also serve as a generative model, enabling the decoding of any point in the 2D space as required

[17]. An autoencoder is a neural network comprised of two primary components: an encoder network that condenses high-dimensional input data, such as an image, into a lower-dimensional embedding vector, and a decoder network that reconstructs the original domain from a given embedding vector, for instance, transforming it back into an image. A diagram illustrating the network architecture is provided in fig (1)

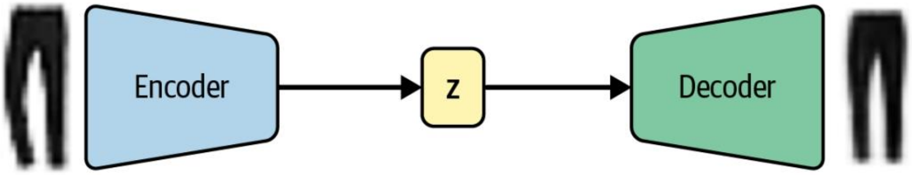


Diagram of Autoencoder Structure [17]

A latent embedding vector is first created by encoding an input image, which is subsequently decoded back into the original pixel space. After an image has gone through the encoder and the decoder, the autoencoder is taught to recreate the image. While it may initially appear peculiar to reconstruct images that are already available, we will soon understand the significance of the embedding space, also known as the latent space. Sampling from this space enables us to generate new images. First, let's clarify what we mean by an embedding. the compression of the original image into a lower-dimensional latent space is called an embedding (z). the basic notion is that we can create new images by selecting any point in the latent space and then running that point through the decoder. because the decoder has mastered the art of transforming latent space points into usable images, this is possible. the encoder's job in an autoencoder is to translate the input image into a latent space embedding vector. using convolutional transpose layers in place of convolutional layers, the decoder functions as an encoder's mirror image.

we must define a model that depicts the passage of an image through the encoder and back out through the decoder to train the encoder and decoder simultaneously. fortunately, keras simplifies this process greatly. the complete autoencoder is defined by the keras model [17].

this model takes an image, processes it via the encoder, and then outputs the result back through the decoder to create a reconstruction of the original image. utilizing a particular probability distribution (the gaussian distribution), vae has the advantage of allowing the model to learn a smooth latent state representation of the input data. vae is an effective model for bayesian inference with latent variables because of its emphasis on variational inference.

the encoder $(z|X)$ uses a kullback-leibler (kl) divergence penalty to encode the data instance x into a latent representation space z to learn the distribution of a hidden variable. using error minimization, the decoder $(x|z)$ then reconstructs z back into the original data space. neural networks with parameters ϕ and θ, respectively, are used to create the encoder and decoder. this process can be described with the following equation:

$\mathbf{log\ P(X) - DKL\ [Q(z|X)||P(z|X)] = E[log\ P(X|z)] - DKL\ [Q(z|X)||P(z)\ldots\ldots(2)}$[2].

This approach is documented in [7], a refinement to the traditional encoder and decoder network of autoencoders involves the incorporation of additional stochastic layers. After the encoder network, the stochastic layer utilizes a gaussian distribution to gather data, while following the decoder network, it employs a bernoulli distribution for this purpose to generate images and figures based on their training distribution. Variational autoencoders (VAEs) permit the setting of complex priors in the latent space, thereby enabling the learning of powerful latent representations [18] sections of the image were gradually produced using a vae and a bidirectional rnn to focus via the captions. It involved the utilization of a variational encoder-decoder (ved) to generate images from textual input. VAE and autoregressive approaches do not perform as well as GANs in terms of generating clearer samples [6].

1.6.3 Diffusion Model

The fundamental idea behind diffusion models is simple. Gaussian noise is added to the input image x0 during each iteration, causing the noise to diffuse. T time steps are spent repeating this process, which finally renders the original image unrecognizable. The goal is to identify a model that can produce a distinct image by reversing the diffusion from a chaotic input. The re-parameterization technique can be used to determine the conditional probabilities (pt|xt), but it is uncertain what the reverse conditional probability (qt|xt) is. A neural network model is trained to estimate these conditional probabilities to address this. Trainability and flexibility are two benefits of diffusion models, however, these goals are at odds with generative models. However, due to their reliance on an extensive Markov chain of diffusion stages, they are computationally demanding. Diffusion models are of great interest, and scientists believe that algorithms that can provide sampling as quickly as GANs will be available soon [7]. To investigate how gender is expressed differently in text-to-image models, scholars in [19]. focused on gender presentation disparities utilizing precise self-presentation variables. Through human annotation, they analyzed the frequency variances of presentation-centric variables (such as "a shirt" and "a dress") and looked at gender indications in the input text, such as "a woman" or "a man." Furthermore, a novel metric named GEP was presented, along with an automated technique for approximating these differences. Therefore, the generation of realistic clinical scenarios, which can be a suitable solution to the lack of medical datasets, should be explored in medical imaging. It is noticed that the diffusion models are significantly better than techniques like GANs or VAEs, where the diffusion model provides better image quality along with better scalability and control. As a result, they have become widely adopted quickly as the best method of achieving high-quality image outputs. Specific emphasis has been made on bigger models like DALL-E 2. The beauty of generative diffusion models is the ability to generate images that have no relation at all with the training set. No other large-scale training efforts have reported problems with overfitting and a group of researchers working in private subject areas have suggested that those using the diffusion models may protect instances with real images by creating fakes [20]. Originally, diffusion models belonged to the family of generative probabilistic models aimed at purposely adding noise to data and then learning to the CD phase and generating new samples. These models have become prevalent when it comes to deep generative models and perform admirably well in macro areas like image generation, videography, and molecular graphics. Recent research on diffusion models has primarily focused on three main approaches: denoising diffusion probabilistic models (DDPMs), score-based generative models (SGMs), and stochastic differential equations (Score SDEs). In [21] study investigated the utilization of LDMs for book illustration, employing the Stable Diffusion model to produce graphics from prompts based on seven classical Brazilian literary texts. The researchers find that demonstrate the efficacy of image formation is greatly affected by the quality and specificity of the suggestions given. The steady diffusion model surpasses traditional methods like as GAN and AttnGAN. Substantial progress in the diversity, realism, and correctness of produced images has been achieved by the fine-tuning of the stable diffusion model, effectively addressing critical challenges in text-to-image conversion [22]. Diffusion models, recognized for their superior image generating and editing skills, have transformed the creation of digital artwork. Nonetheless, their capacity for generating unlawful or detrimental images presents considerable apprehensions. The researchers have developed many image security strategies utilizing undetectable perturbations to inhibit diffusion models from acquiring valuable characteristics from the safeguarded images. This study illustrates that assailants might bypass these safeguards by utilizing semantic and linguistic contrastive alignment alongside visual cues, such as images. Their trials demonstrate that the solution, INSIGHT, surpasses fundamental defenses such as Crop+resize and the leading DM-based method, Impress [23]. This study introduces Tina, a text-to-model neural network diffusion model designed for train-once-for-all customization. Tina has demonstrated exceptional proficiency in creating individualized models from text prompts, exhibiting the capacity to generalize across both in-distribution and out-of-distribution tasks, including zero-shot and few-shot image prompts, natural language prompts, and novel classes. Tina further allows customization across various class quantities. This study investigates the capabilities of text-to-model generative AI and introduces novel applications for neural network diffusion in user customization [24]. The text-to-image generator full-stack web application exemplifies a culmination of innovation at the convergence of deep learning technology and user-centric design. The platform integrates the Stable Diffusion XL foundation model, allowing users to transform verbal descriptions into visually cohesive visuals with exceptional fidelity. The program offers a straightforward interface and advanced back-end technology, facilitating the efficient generation of images from textual prompts for a wide array of users across several disciplines. This study offers several advantages. The tool enables users to convert written concepts into striking visual representations, enhancing communication and expression effectively. Furthermore, the implementation of sophisticated deep learning methodologies guarantees that the produced images demonstrate a significant degree of realism and precision, providing valuable uses in domains such as design, marketing, and content generation. The text-to-image generator full-stack web application exemplifies the revolutionary capacity of artificial intelligence in augmenting human creativity and productivity [25]. The researchers introduce in this study an inaugural systematic safety evaluation concerning the production of hazardous images, namely nasty memes, generated using Text-to-Image algorithms. To quantitatively assess the safety of produced images, they initially developed a safety classifier to identify dangerous images based on the established criteria for harmful content. Subsequently, they use this classifier on four exemplary Text-to-Image algorithms to assess their safety using three detrimental prompt datasets and one innocuous prompt dataset. Their findings indicate that Text-to-Image models exhibit significant rates of producing dangerous images when adversaries deliberately employ damaging stimuli. Moreover, it is feasible to produce inappropriate images even with innocuous suggestions. they comprehensively assess the capability of Text-to-Image algorithms in producing hostile memes. The assessment results indicate that as much as 24% of the created meme variations exhibit traits and attributes akin to real-world hostile meme variants, which can be used for hate campaigns online [26]. This study presents an innovative method that integrates Classifier-Free Guidance (CFG) with Score Identity Distillation (SiD) to effectively distill Stable Diffusion models into efficient one-step generators. The researchers have refined our revolutionary Long and Short CFG methods (LSG) utilizing exclusively synthetic images produced by the one-step generator. This study not only confirms the practical viability of SiD but also sets new standards for one-step diffusion distillation, attaining exceptional zero-shot FID scores on the COCO-2014 validation set. their technique is engineered for improved efficiency while maintaining performance, enabling learning from the teacher model without the necessity of real images or the inclusion of supplementary regression or adversarial losses. They have made their code and condensed models accessible to promote more study [27]. The researchers in this study investigate text-guided image modification with a Hybrid Diffusion Model (HDM) architecture akin to DALLE-2. Their architecture comprises a diffusion prior model that produces CLIP image embeddings based on a text prompt, with a unique Latent Diffusion Model specifically trained to generate images conditioned on CLIP image embeddings. The researchers found that the diffusion prior model may provide text-guided conceptual modifications in the CLIP image embedding space without requiring any fine-tuning or optimization. The researchers integrate this with structure-preserving modifications on the image decoder utilizing established methods like reverse DDIM for text-guided image editing. Their methodology, PRedItOR, needs no further inputs, fine-tuning, optimization, or targets, and demonstrates outcomes that are comparable to or superior to baseline measures, both qualitatively and numerically. The researchers offer an enhanced analysis and comprehension of the diffusion prior model [28]. The denoising diffusion probabilistic model (DDPM) was introduced in 2020 as a significant research effort, leading to a substantial surge in interest within the generative model community ever then. In this article, they provide a comprehensive overview of DDPM, starting with a summary of the key advancements made before the development of DDPM, then they go into the workings of unconditional DDPM, using image synthesis as a specific case study. Additionally, they emphasize the role of guidance in facilitating conditional decision-making, which is essential for understanding text-conditioned decision-making in the context of converting text to images. The rise of DDPM may mostly be attributed to two initial endeavors: score-based generative models (SGM), which were investigated in 2019, and diffusion probabilistic models (DPM), which surfaced as early as 2015 [29].

Table 1.1 : Summary Of Previous Work About Image Generation From Text

Ref.	**Dataset**	**Model**	**Assessment Metric**	**Limitation**
Dhariwal, Nichol(2021).[30]	ImageNet 128×128	ddpm		Access to labeled datasets is limited.
Sharma et al. (2018).[31]	MS COCO	ChatPainter's architecture	IS= 9.74	In many circumstances, the results are unrecognizable. Training the model using conversation data is also quite unstable.
Singh et al.(2018)[32]	oxford flowers-102	CanvasGAN	IS=2.94	discontinuity in higher-dimensional latent mapping because of insufficient data
Ouyang et al. (2018).[33]	-OXFORD-102 cub	Conditional GAN	Euclidean distance and SSI similarity used	No assessment metric exists.
Tao Xu et al. (2017).[34]	CUB *COCO*	AttnGAN	IS= 4.36	Not quite accurate in representing the world's cohesive structures.
Schulze et al. (2022).[35]	CUB *COCO*	CAGAN	(33.89, 32.60) (4.78, 4.96)	fails to create realistic-looking images, although scoring better ISs than the AttnGAN.
Kim et al.(2022).[36]	ImageNet	Diffusion-CLIP	Directional CLIP similarity (Sdir), segmentation-consistency (SC), and face identity similarity (ID) used	Perhaps used to deceive individuals with modified realistic consequences.
Gu et al(2022).[37]	CUB-200 OXFORD-102 MSCOCO CONCEPTUAL CAPTIONS LAION-400M FFHQ256	VQ-Diffusion	FID=(13.86 ,10.32 ,14.10)	Token substitution can have a major impact on the semantics of the port representation. The model does not know which tokens have been substituted, which increases its robustness throughout the denoising process.
Qiao et al. (2019).[38]	Oxford CUB	LeicaGAN	IS= (5.55±0.06, 3.75±0.0)	In the current implementation, the TVE models received independent training from the MPA and CAG models.

CHAPTER TWO

THEORETICAL BACKGROUND

2.1 Overview

In this chapter, a description of different diffusion models is provided. Also, a detailed explanation of the stable diffusion model which is used for the task of flower image generation from texts in this study is presented.

2.2 Introduction

Computer-generated visualizations that resemble real-world scenarios, known as synthetic images, have a wide range of applications in a variety of disciplines, including healthcare, biomedicine, fashion, architecture, geospatial studies, automotive, security, and surveillance. They contribute to the development of innovative solutions, decision-making, and data analysis. Recent advancements in text-to-image generators, including Imagen, Stable Diffusion, DALL-E 2, eDiff-I, and ERNIE-ViLG 2.0, have resulted in substantial advancements and are now extensively employed in domains such as computer graphics, cultural arts, medical, and biological data generation [39]. Content and artistic manifestations have been the subject of deep generative models, such as the Generative Adversarial Network (GAN). Nevertheless, they encounter obstacles in terms of inconsistent outputs and consistent training procedures, which renders them challenging to implement and extend into other fields. As a consequence, likelihood estimation-based models are being implemented to enhance the quality of GAN samples [5]. Due to their capacity to generate detailed and diverse images, denoising diffusion models have become increasingly popular. These models are capable of being effectively implemented across a variety of data formats, including 3D point clouds, videos, and audio. They are employed for the purpose of augmenting resolution, editing, filling in missing sections, and transforming images. Lifelike images have been generated from detailed textual descriptions through diffusion-based text-to-image technologies. The primary function of these models is to generate graphical artwork that is consistent with the text provided. Nevertheless, they are restricted to the identification of art-related terms and do not provide comprehensive descriptors such as color distribution or brushwork nuances. Models that are trained on uncurated text-image pairs frequently demonstrate a bias toward a particular subset of styles, which is indicative of the bias present in the training data [40]. Text-to-image generation is the process of generating images from textual descriptions. Blended diffusion employs natural language instructions to integrate pre-trained DDPM and CLIP models for region-specific editing. A wide range of images can be processed by it. UnCLIP (DALLE-2) employs a dual-stage procedure that commences with a prior model that generates an image embedding that is influenced by text captions [41]. A probabilistic framework that generates a data set by sampling from it is known as a generative model. It is designed to comprehend the fundamental principles of a car's appearance, thereby creating a new image while preserving a realistic appearance. In order to accomplish this, it is necessary to generate a data file that contains a multitude of exemplar images of the vehicle. The image synthesis task is difficult to complete due to the vast array of potential pixel assignments and limited image arrangements. The training dataset contains observations with unique pixel intensities, which can be used to generate new attribute sets that adhere to the original dataset's rules. The resulting images are reorganized and can identify the subject as the same thing, but not a replica of the initial observation. A deterministic model, which carries out a predetermined computation like calculating the average value of each pixel in a dataset, t is not classified as generative since it consistently generates identical results.[17]. The model must have the capacity to estimate the input distribution and then produce fresh, independent observations that closely resemble the original training set [17] [42].

2.3 Text to Image Models

Text-to-Image Models large multimodal models that turn a text cue into an image now dominate cutting-edge image generation. Text-to-image models are extremely valuable because they enable users to quickly edit created images using natural language. In contrast to models like StyleGAN, which are impressive, they do not have a text interface that allows you to explain the desired image to be generated. There are now three text-to-image generating models available for both commercial and personal use: DALL.E 2, Midjourney, and Stable Diffusion. DALL.E 2, developed by OpenAI, is a subscription-based service that may be accessed by a web application and API (Application Programming Interface). Midjourney provides a text-to-image service through its Discord channel, which requires a membership. Both DALL.E 2 and Midjourney provide complimentary credits to anyone who joins their platform for the first investigation [17]. The proliferation of user-friendly open-source generative text-to-image AI has sparked significant public interest in the field. Systems such as Midjourney, DALL-E, Disco Diffusion, and Stable Diffusion utilize generative algorithms and a userfriendly interface. These systems allow users to input natural language prompts and receive a variety of visually generated outputs that represent the concepts mentioned in the text. Recently, there has been a notable increase in the understanding and availability of these systems, which can generate high-quality images in many artistic styles that closely align with text prompts with apparent precision. This naturally raises concerns over the potential impact of text-to-image artificial intelligence on conceptual matters of engineering design. Generative art has had a significant increase in generation in public discussions. Additionally, designers in several sectors have started using text-to-image AI to create magazine covers and storyboards. Nevertheless, the utilization of text-to-image artificial intelligence in the context of engineering design has not been extensively explored. An advantageous use of text-to-image AI systems in conceptual engineering design is as a visual aid to quickly visualize concept ideas and variations during concept generation. This enables the efficient creation of high-quality images that can be utilized to facilitate interpretation and decision-making in concept selection and development. There are several potential benefits to this, such as improved efficiency to take full use of a competitive edge by lowering the time spent on visualizing data and facilitating quick examination of a greater range of options .The generated visuals have the potential to inspire novel concepts that human designers, limited to a narrower design environment, may not have discovered on their own. Furthermore, it can enable those without expertise in design sketching and Computer Aided Design (CAD) to express their thoughts more easily, hence increasing the accessibility of the design process. A closer analysis of the methods incorporated in actual text-to-image systems points to several factors that might lessen the effectiveness of the systems. Some of these challenges include the ability of these tools in engineering design settings as well as the suitability of these tools, which implies that there is room for research.

2.4 Diffusion Models Types

Diffusion models (DM) are gaining popularity due to their stable training and superior quality of samples compared to (GANs). They mitigate GANs' shortcomings like mode collapse, adversarial learning burden, and convergence failure. Diffusion models use Gaussian noise to infuse training data and retrieve original data from corrupted versions. They are suitable for scalability and parallelizability, and their training methodology requires minimal changes to the original data. This enables learning of data distributions that closely resemble original information, resulting in robust realism in produced samples. Diffusion models are a subgroup of probabilistic models that need substantial processing resources to represent unseen data intricacies. Their training procedure necessitates the assessment of models that employ iterative estimates and gradient calculations. The computing expense significantly escalates while processing high-dimensional data such as images and videos [43]. Researchers in this reference [44] reached cutting-edge performance in both the estimation of density and sample quality. DMs primarily serve to elucidate the dissemination of information, behaviors, or phenomena within a population and discern the factors that impact the process of diffusion. The phenomena of diffusion and reverse diffusion. Diffusion processes, also known as forward processes, are a specific form of continuous-time Markov process that exhibit almost continuous sample routes in the field of probability theory and statistics. To be more precise, diffusion processes in digital media include the incremental addition of Gaussian noise to images. When a sample x0 is taken from the true image distribution, the diffusion process introduces Gaussian noise at each step into the samples x1, x2, ..., xT throughout the T steps. Furthermore, q(xt |xt−1) is a Gaussian distribution with xt−1 as the mean, and xt is drawn from this Gaussian distribution. Consequently, we may obtain :

$$\boldsymbol{q(xt \mid xt-1) := \; N\, xt\, ;\, p\; 1-\beta txt-1, \beta tI} \quad \textbf{................(3)}$$

Where ($\boldsymbol{\beta t}$) is a fixed and predetermined constant. To obtain the value of ($\boldsymbol{xt}$) at each step, we can generate a random sample from a conventional Gaussian distribution, which is then scaled by the standard deviation and shifted by the mean value. To streamline the diffusion process from the original image x0, we can express the equation as follows:

$q(xt \mid x0) = N(xt; \bar{\alpha}tx0, (1 - \bar{\alpha}t)I)$…………..**(4)**

where $\alpha t := 1-\beta t$ and $\bar{\alpha} t := \prod t\ s=1$.

To achieve the reverse diffusion process, we need to reverse the direction of the aforementioned process. This means that if we can obtain a sample from q(xt | xt−1), we may reconstruct an authentic original sample from a random Gaussian distribution N (0, I). In other words, we can generate a real image from a distribution that is very disordered and noisy. Nevertheless, given we must determine the data distribution from the entire dataset, we can't forecast q(xt | xt−1). To carry out the reverse diffusion process, it is necessary to first acquire a model εθ that provides an approximation of the conditional probability. The denoising model, denoted as εθ, is trained to minimize the loss function depicted below:

$LDM = E[xt, \varepsilon \sim N(0,1)]\ [(1/2)\ ||k\varepsilon - \varepsilon\theta(xt, t)||\hat{}2]$ ……….**(5)**

Where t is evenly sampled from the T time steps. These methods employ a Forward Diffusion Process (FDP) to incorporate Gaussian noise into the data and then acquire the ability to reverse the process, converting the noise back into data. Multiple techniques exist for establishing diffusion models: Score-Based Models (SBMs) are trained to predict the score, which represents the gradient log density. On the other hand, diffusion Models (DMs) are trained to predict the extra Gaussian noise, which is then subtracted from the noisy data. SBMs and DMs often depend on Gaussian noise, although it is unclear why Gaussian noise is preferred over other forms of noise. Recent research has begun to investigate non-Gaussian noise. In their respective studies,[45] and [46] propose diffusion-based frameworks for sampling from arbitrary distributions by transitioning from dataset 1 to dataset 2. Both papers demonstrate that substituting non-Gaussian distributions for Gaussian noise as the second dataset substantially degrades the quality of the generated data [47]. Diffusion models (DMs), often referred to as generated models, are Markov chains that have been trained via variational inference. They are also known as diffusion probabilistic models. The objective of DM is to introduce noise, such as diffusion, into the data to generate samples [29].

2.4.1 Dalle 2 model

DALL-E 2 is a sophisticated AI system that generates graphics from textual descriptions, published by OpenAI in 2021.

To comprehend the functionality of DALL.E 2, three unique components must be analyzed: the text encoder, the prior, and the decoder.

The text is initially processed by the text encoder to generate a text embedding vector. The vector is subsequently converted by the prior to generate an image embedding vector. Ultimately, information is transmitted through the decoder, in conjunction with the original text, to yield the created image [17].

This advanced technology can generate intricate, high-quality graphics that faithfully represent the written description, regardless of its length or complexity. The AI model was trained on an extensive collection of text-to-image combinations, enabling it to produce visuals with exceptional detail and precision. DALL-E 2's capability to create representations of non-existent items and situations is one of its most remarkable attributes. For instance, it can generate a image of a "banana dog" only from a textual description. This is a substantial advancement in artificial intelligence with vast potential applications across several sectors, including art, design, and advertising [48]. For generating images from text DALL-E 2 utilizes 650 million image-text pairings, whereas Imagen employs 860 million image-text pairs. It necessitates extensive training sessions, demanding significant processing power and resources. Moreover, the datasets and code sources are sometimes inaccessible to the public, hence complicating the replication of these generative models [49].

2.4.2 Imagen Model

The Google Brain team published their own model for the text-to-image generation purpose, just over a month after OpenAI released DALL.E 2 [17]. Imagen is a text-to-image diffusion model that merges the power of transformer language models with precise representation diffusion models to achieve an unparalleled level of imagerealism and a profound level of language understanding in text-to-image synthesis. The primary outcome behind Imagen is that text embeddings from large transformer language models, pre-trained on text-only corpora, are notably successful for text-to-image synthesis. Imagen consists of a frozen T5-XXL encoder that transforms input text into a series of embeddings, accompanied by a 64×64 image diffusion model, and subsequently two super-resolution diffusion models for producing 256×256 and 1024×1024 images [50].

Imagen has several similarities with DALL.E 2, including a text encoder and a diffusion model decoder. A primary distinction between the two models is that the Imagen text encoder is trained only on textual input, whereas the DALL.E 2 text encoder's training incorporates image data via the contrastive CLIP learning goal [17].

2.4.3 Stable Diffusion Model

In August 2022, there was another noteworthy advancement in the realm of artificial intelligence that garnered widespread attention. Stable Diffusion was a product of the collaboration between Stability AI and Computer Vision with the Learning research group at Ludwig Maximilian University of Munich and Runway. Unlike DALL.E 2 and Imagen, this particular model's code and model weights have been openly shared on Hugging Face. This suggests that anybody may engage with the model using their hardware without the requirement of using a specific API, and the model can generate outcomes that are similar to those of DALL-E 2. Stability AI surprised the IT world by promptly making their model open source, diverging from the approach of other companies in the field. Hugging Face, a hub for sharing pretrained models, datasets, and demos of machine learning projects, promotes open source contributions and fosters a collaborative environment for AI enthusiasts [17],[51]. . Stable Diffusion comes with a safety filter that aims to prevent generating explicit images. Unfortunately, the filter is obfuscated and poorly documented [52]. This technique operates within the latent space, which is essentially a compressed version of the image, much like a smaller, condensed file. This family of models is known as latent diffusion models. To create the latent space, an autoencoder is employed, acting as a simplified version of the variational autoencoder mentioned earlier. The autoencoder's encoder compresses the image into lower-dimensional data, similar to zipping a file, whereas the decoder decompresses the latent data back into an image, akin to unzipping a file. One of the most remarkable features of stable diffusion is its ability to generate images from text prompts in a highly impressive manner. The diffusion model is adapted to accept conditioning inputs, comparable to modifying a recipe based on a special request. Text inputs are transformed into embeddings (vectors) using a language model, reminiscent of the process employed by CLIP [51]. Stable Diffusion utilizes latent diffusion as its generative model, setting it apart from other text-to-image models. In December 2021, Rombach et al. released a publication titled "High-Resolution Image Synthesis with Latent Diffusion Models," in which they proposed a novel concept known as latent diffusion models (LDMs). The main goal of the research is to integrate the diffusion model into an autoencoder so that the diffusion process can function on an image's latent space representation as opposed to the actual image [17].

The architecture of this model is distinct in its capability of combining different components. In the segments below the fundamental components U-Net, Variational Autoencoder (VAE), and Clip will be discussed.

2.4.3.1 U-Net Architecture

In several tasks involving image recognition, deep convolutional networks have outperformed the state-of-the-art. Although convolutional networks have been around for a while [7]. The scale of the networks under consideration and the training sets that were accessible to them restricted their success. The breakthrough by Krizhevsky et al.[6] was caused by the supervised training of a large network using the ImageNet dataset, which contains one million training images, and has eight layers and millions of parameters. Larger and deeper networks have since been trained [11]. Convolutional networks are usually applied to classification problems, where the result is an image with a single class label. A U-Net is a type of convolutional neural network that was created for segmenting biomedical images. It has a special architecture that combines a shrinking network with successive layers of operators that up-sample the output, thus increasing the resolution. This gives it a U-shaped form, which is why it's called a U-Net [17]. A U-Net consists of two halves: the up-sampling half, where representations are extended spatially while the number of channels is decreased, and the down-sampling half, where input images are reduced geographically but enlarged channel-wise. But in the network's down-sampling and up-sampling sections, there are also skip connections across layers with the same spatial structure, in contrast to a VAE [5]. Data flows through a VAE sequentially, one layer at a time, from input to output. A U-Net differs in that information can move to later levels and avoid some areas of the network thanks to skip connections. When we want the output to have the same form as the input, a U-Net is quite useful. The U-Net is the most suitable option for the network architecture in the diffusion model example, as it allows the model to predict the noise that will be introduced to an image that has the same structure as the original image.

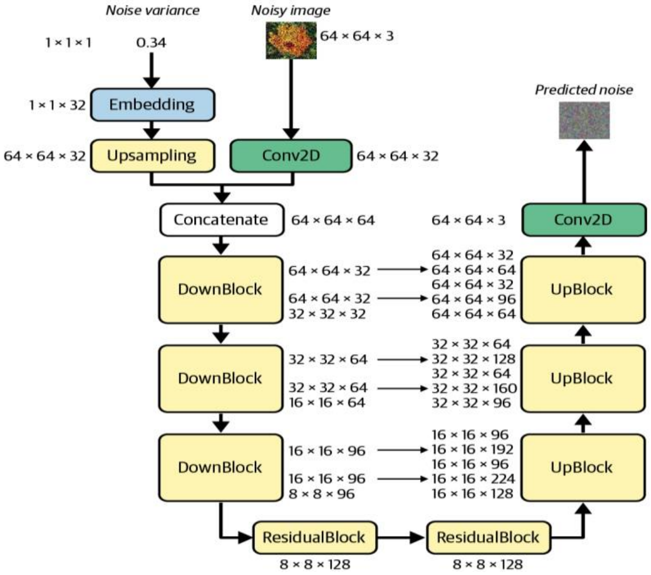


Figure 2.1: U-Net architecture diagram [17]

The U-Net has two inputs: the noisy image and the noise variance (a scalar). The image goes through a Conv2D layer to increase channels. The noise variance is encoded by a sinusoidal embedding and replicated to match the image size. Channels are used to join the inputs. The output of the DownBlock layers is saved in the skips list for use in subsequent skip connections. The tensor passes through layers called DownBlock, which reduces image size and raises channels; ResidualBlock, which maintains channels constant; and UpBlock, which increases image size and decreases channels. The DownBlock layers' output is utilized by the skip connections. The last Conv2D layer reduces the number of channels to three (RGB). The images with high levels of noise and their corresponding noise variances are inputted into the U-Net, a Keras model, which accurately predicts the noise map. The U-Net architecture has four more components: the Residual Block, the UpBlock, the DownBlock, and the sinusoidal embedding, which will be clarified:

A- Sinusoidal Embedding

An article authored by [53] Implemented sinusoidal embedding for the initial time. We will modify the original notion, as presented in the paper "NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis" by [54] Given the specific objectives we have. Transforming a single value (the noise variance) into a unique higher-dimensional vector that can offer a more complex representation for use farther down the network is the aim. The NeRF work expands this idea to include continuous values, whereas the original research used it to translate discrete word positions into vectors.

B- Residual blocks

It is a constituent of both the UpBlock and the DownBlock. A residual block refers to a collection of layers that includes a skip connection, where the input is added to the output. By employing residual blocks, we may construct networks with increased depth that are more resilient to the issues of vanishing gradient and gradient vanishing. The vanishing gradient problem is defined by a minuscule gradient in the deeper layers and a slow learning rate as the network becomes deeper. While the degradation problem may come as a surprise, in reality, it is understood that deeper layers must learn at least the identity mapping, which is a difficult task given the other difficulties that deeper networks encounter, such as the vanishing gradient problem. The answer wasstraightforward and initially put out in the ResNet publication by [55], in 2015. By including a skip link highway around the main weighted layers, the block may circumvent identity mapping and prevent complex 34 weight modifications. This ensures that the magnitude of the gradient and the accuracy of the network are maintained during extended training of the network.

C-Downblocks And UpBlocks

The DownBlock consists of a series of ResidualBlocks, with a block_depth of 2, to enhance the channel count. The next layer that it comes after an AveragePooling2D layer that reduces the size of the image to half. A list is generated that contains the ResidualBlock as the skip connections which are required by the UpBlock levels in the U-Net architecture. An UpBlock is initiated with the UpSampling2D layer which applies the bilinear interpolation to double the size of the image. Every UpBlock in the U-Net has a block_depth of 2 ResidualBlocks applied to reduce the number of channels and a Concatenate layer to join the output of the DownBlocks using skip connections at every stage. Thus, the inclusion of more channels is incorporated through a Residual Block with a specified width to enrich the image's representation by the DownBlock. Each of the enhanced channels is kept as a list called “skips” ready to be used by the UpBlocks. Since the utilization of a direct AveragePooling2D layer will result in a reduction of the number of pixels by 50%. Initially, in the UpBlock, an UpSampling2D layer is employed to expand the dimensions of the image to twice its original size. A Concatenate layer is employed to merge the output of a DownBlock layer with the present output. A ResidualBlock decreases the overall channel count of the image as it traverses the UpBlock [17].

2.4.3.2 Variational Autoencoder

VAEs combine the finest neural networks with Bayesian inference. They are among the most fascinating neural networks and have emerged as one of the most popular ways to unsupervised learning. They're not your typical autoencoders. Autoencoders use further stochastic layers alongside the conventional encoder and decoder networks. After the encoder network, the stochastic layer employs a Gaussian distribution to sample the data, while the decoder network utilizes a Bernoulli distribution for data sampling. VAEs, similar to GANs, can generate visual representations and illustrations based on the specific distribution they were trained on. VAEs enable the specification of intricate priors in the latent space, hence obtaining robust latent representations [7]. A two-step generative method is proposed by the original Variational Autoencoder (VAE): latent variables $z \in R$ h are sampled from a prior distribution $p(z)$ first, then observations x are created from a conditional distribution $p\theta$ $(x|z)$. Formally, $z \sim p(z)$ and $x \sim p\theta(x|z)$ represent the generating process. It is assumed that the prior probability distribution $p(z)$ has a Gaussian distribution, and a neural network is used to simulate the likelihood $p\theta(x|z)$. In the previous studies, the decoder is sometimes referred to as the likelihood model since it converts latent variables into observations. The probability is originally defined using a multivariate Gaussian distribution N $(\mu\theta(z)$, diag$(\sigma$ 2 $\theta(z)))$ for continuous data and a categorical distribution for discrete data. The generative model is trained by finding the decoder parameters θ that maximize the sum of the marginal likelihoods of individual points $p\theta(X)$ = X $x \in X$ log Z $p\theta(x|z)p(z)dz$. Despite the difficulty of these integrals, the introduction of an approximation that represents the posterior distribution $q\varphi(z|x)$ enables the maximization of the related evidence lower. As the estimated posterior, $q\psi(z|x)$, gets closer to the true posterior, the evidence lower bound (ELBO) becomes more precise. The generative model is learned by optimizing the parameters θ of the decoder and φ of the estimated posterior together using stochastic gradient ascent. The approximate posterior is represented by the conditional multivariate Gaussian distribution N $(\mu\varphi(x)$, diag$(\sigma$ 2 φ $(x))$ in the original VAE's encoder. The encoder is forced to choose between two competing goals by the ELBO loss. The system should be able to approximate the previous distribution closely while encoding the data correctly. As a result, the prior and posterior distributions' Gaussian assumptions are frequently contradictory, which limits the capacity to provide performance. A different strategy is to obtain a prior distribution that coincides with the posteriors that were previously obtained. For example, it has been demonstrated by [56],[57] that the performance of VAEs is significantly improved when autoregressive models with normalizing flows [58] as prior distributions are used. Here, we demonstrate how the application of denoising diffusion probabilistic models can improve the efficiency of conventional VAEs [59],[60].

2.4.3.3 Clip Model

Utilizing natural language supervision is a highly efficient method for acquiring knowledge in the field of image representation. Recent research has indicated that augmenting the dataset by incorporating data obtained through web scraping can lead to significant enhancements in the model's performance. In particular, GPT [61] and BERT [62] studies have shown that an effective understanding of natural language relies on a substantial volume of texts and a well-designed self-supervised learning method. On the other hand, computer vision has traditionally relied on strict monitoring, utilizing "gold labels" which are separate class labels. Typically, these annotations are gathered from a large group of people, making it challenging to gather a substantial amount. CLIP utilizes the vast amount of textual material that accompanies imagegraphs on the internet to expand its collection. The latter consists of 400,000 combinations of images and titles, where the caption is designed to express the semantic significance of the image. CLIP utilizes distinct models, namely text and image encoders, to produce embedding for both text and images). The text encoder's architecture is based on a Transformer concept [53], the image encoder is a Vision Transformer. The two encoders are simultaneously trained using a contrastive approach, where the goal is to maximize the cosine similarity between the embedding of caption-image pairs and minimize it for unrelated captions and images (see Fig 2.2). Hence, the acronym, ‘CLIP’, represents Contrastive Language-Image Pre-training. The assumption underlying this technique is that the image and word embedding exist inside a shared multimodal latent space, which means that similar captions and images should be close to each other in terms of cosine similarity [49].

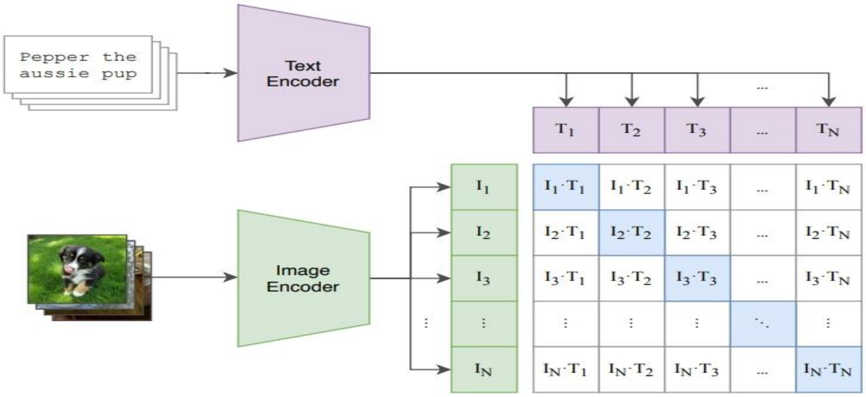

Pepper the aussie pup
Text Encoder
Image Encoder
T₁ T₂ T₃ ... T_N
I₁ I₂ I₃ ⋮ I_N
I₁·T₁ I₁·T₂ I₁·T₃ ... I₁·T_N
I₂·T₁ I₂·T₂ I₂·T₃ ... I₂·T_N
I₃·T₁ I₃·T₂ I₃·T₃ ... I₃·T_N
⋮ ⋮ ⋮ ⋱ ⋮
I_N·T₁ I_N·T₂ I_N·T₃ ... I_N·T_N

Figure 2.2: The architecture of the CLIP model [49]

2.5 Evaluation Metrics

Text-to-image conversion strategies are now being evaluated using both human assessment and quantitative metrics. However, improved measures for statistical and qualitative evaluation of these models are required. The assessment metrics should generate findings that are equivalent to human evaluation, which remains an aim [6]. Despite concerted attempts to establish assessment criteria, the task of generating a diverse and impactful review still poses challenges. For instance, FID does not always align with perceptual quality, and the CLIP score is insufficient in evaluating various limitations of current automated assessment methods. It is imperative to enhance the strength and variety of automated evaluation criteria. Moreover, the limitations in efficiency and aesthetic differences among raters impose constraints on the number of prompts available for human review. Moreover, the majority of benchmarks incorporate a variety of textual inquiries that enable users to evaluate the model from many perspectives. The quality of prompts may be limited, especially when evaluating complex situations, and prompts created by humans may include biases. Although specific datasets have demonstrated favorable outcomes, the current review process is still not optimal. When evaluating a solitary item scenario, it is crucial to assess the visual clarity and definition. When it comes to evaluating the quality of images, Inception Score (IS) and Fréchet Inception Distance (FID) are widely used metrics. They are efficient in assessing visual quality in the majority of situations involving a single item. The primary measures used to assess image quality are Fréchet Inception Distance (FID) and Inception Score (IS) [29]. Assessing generative models may be challenging when employing quantitative methods. The challenge of evaluating model performance solely based on pixel precision is intensified by the potential for a single description to correlate with several images [63]. Additionally, because matching data distributions are not equivalent, existing evaluation measures like FID and IS scores do not entirely fit with the key objectives. A different strategy would be to develop a diversity score, like CLIP, that accounts for a large number of areas. The diversity of the sample and the efficiency of diffusion models in achieving the intended result should both be considered in the best evaluation metric [64]. Assessing generated images is particularly difficult due to their numerous similarities with high-quality imagegraphs, including visual authenticity and variety. An effective text-to-image paradigm, however, does more than simply produce lifelike visuals. The congruence between generated visuals and textual descriptions is a crucial additional feature.

Images generated from textual descriptions should accurately represent the distribution of the data used for training. The Inception score is a valuable tool for assessing and contrasting models. The FID score was introduced by [65].

2.5.1 Inception- V3 Network

Is a well-known model for object detection and feature extraction developed by Google in 2014. This model was first developed by the Google Brain team and has since been used in diverse applications, including object identification and other fields, via the transfer learning process. An extremely sophisticated convolutional neural network, pretrained on ImageNet. ImageNet Large Scale Visual Recognition Challenge (ILSVRC) includes more than one million images for processing. It one of the most accurate models in its area for image classification, after being trained on the ImageNet dataset. There are diverse versions of Inception, such as Inception V1, Inception V2, and Inception V3. The standard input dimensions for this model are 299x299 over three channels. This model exhibits reduced computational efficiency, substituting bigger convolutions with smaller ones, resulting in decreased processing time. The AlexNet model was proposed by Krizhevsky, Sutskever, Hinton (2012), which can detect objects, and substantial progress has been achieved. The object recognition performance of the Inception-v3 model is slightly better. The Inception network is a part of GoogleNet. Inception was a network that consisted of 22 layers and had 5 M parameters. It had a filter size ranging from 1 × 1 to 3 × 3 to 5 × 5 to extract features at different sizes while using maximum pooling. Using 1 × 1 filters is done so the calculation may be performed more proficiently. In the final portion of 2015, Google upgraded the Inception model to the InceptionV3 version, which factors the convolutional layers to minimize the number of parameters. Convolutional filters of size 5 × 5 are changed to two filters of size 3 × 3 to lower the amount of processing required while maintaining the equivalent level of network performance. There are a total of 48 layers in the InceptionV3 model. [7] , [66], [67].

2.5.2 Model Assessment Tools

Two key metrics to assess the performance of the Stable Diffusion model will be employed [68] [65]:

1. **Fréchet Inception Distance (FID)**: This metric measures the similarity between the distribution of generated images and real images, computed using statistics from an Inception-v3 network. Lower FID scores indicate better performance.

FID can be computed as:

$$\mathbf{FID} = \|\boldsymbol{\mu}_{\mathbf{r}} - \boldsymbol{\mu}_{\mathbf{g}}\|^{\mathbf{2}} + \mathbf{Tr}\ \left(\boldsymbol{\Sigma}_{\mathbf{r}} + \boldsymbol{\Sigma}_{\mathbf{g}} - \mathbf{2}\left(\boldsymbol{\Sigma}_{\mathbf{r}}\boldsymbol{\Sigma}_{\mathbf{g}}\right)^{\mathbf{1/2}}\right)\mathbf{\ldots\ldots\ldots(6)}$$

where $\boldsymbol{\mu}_{\mathbf{r}}$ and $\boldsymbol{\Sigma}_{\mathbf{r}}$ = mean and covariance of real images' features;

$\boldsymbol{\mu}_{\mathbf{g}}$ and $\boldsymbol{\Sigma}_{\mathbf{g}}$ = mean and covariance of generated images' features.

2. **Inception Score (IS):** This metric evaluates the quality and diversity of generated images by considering both the realism and diversity of predictions made by an Inception-v3 network. Higher IS scores indicate better performance.

IS can be calculated as:

$$\mathbf{IS} = \mathbf{exp}\ \left(\mathbf{E}_{\mathbf{x \sim p_g}}[\mathbf{D}_{\mathbf{KL}}(\mathbf{p}(\mathbf{y} \mid \mathbf{x}) \parallel \mathbf{p}(\mathbf{y}))]\right)\mathbf{\ldots\ldots(7)}$$

where $\mathbf{p}_{\mathbf{g}}$ = distribution of generated images;

where $\mathbf{D_{KL}}$ is the Kullback-Leibler divergence;

$\mathbf{p}(\mathbf{y} \mid \mathbf{x})$ = conditional label distribution given image x;

$\mathbf{p}(\mathbf{y})$ = marginal distribution over all labels.

CHAPTER THREE

PROPOSED METHOD

3.1 Introduction

This chapter gives a comprehensive description of the actual processes of how the objective was accomplished including design of the proposed model, data pre-processing, model set-up, training strategies, and the assessment tools used. Hence, the relevance of this work is in helping to bring the capabilities of advanced generative models within the reach of more people. Thus, by having reduced computational costs and training time, these models can be made available and usable by many more people, especially those with fewer resources. Based on the given information, one can advance creativity and invention in numerous sectors to employ artificial intelligence in areas not possible earlier. The figure (3.1) below shows the process of generating images from text. The method is specifically dependent on the stable diffusion model. The transformation among the generative models signified significant progress in AI as these machines are capable of interpreting the human language and transforming it into visual forms inclusive of aesthetics in an image form that is deemed appealing to the eye hence paving the way to greater opportunities as regards to creativity and resourcefulness. Clearly, one of the defining milestones in this sphere is known as the Stable Diffusion model. These models provide a solid foundation for generating different variations of the images that are semantically close to the textual descriptions. Though, such models are terrific in terms of performance, even they have issues with certain restrictions such as high computational complexity and training time. Most of these models are computationally intensive and take a lot of time to train which poses a drawback because it is costly. To meet these limitations, the main aim of this study is to provide a new generation model that will enhance the creation of images from the text. The objective is to develop a model that ideally, retains or improves the quality of the generated images while drastically cutting down on the processing power and training time.

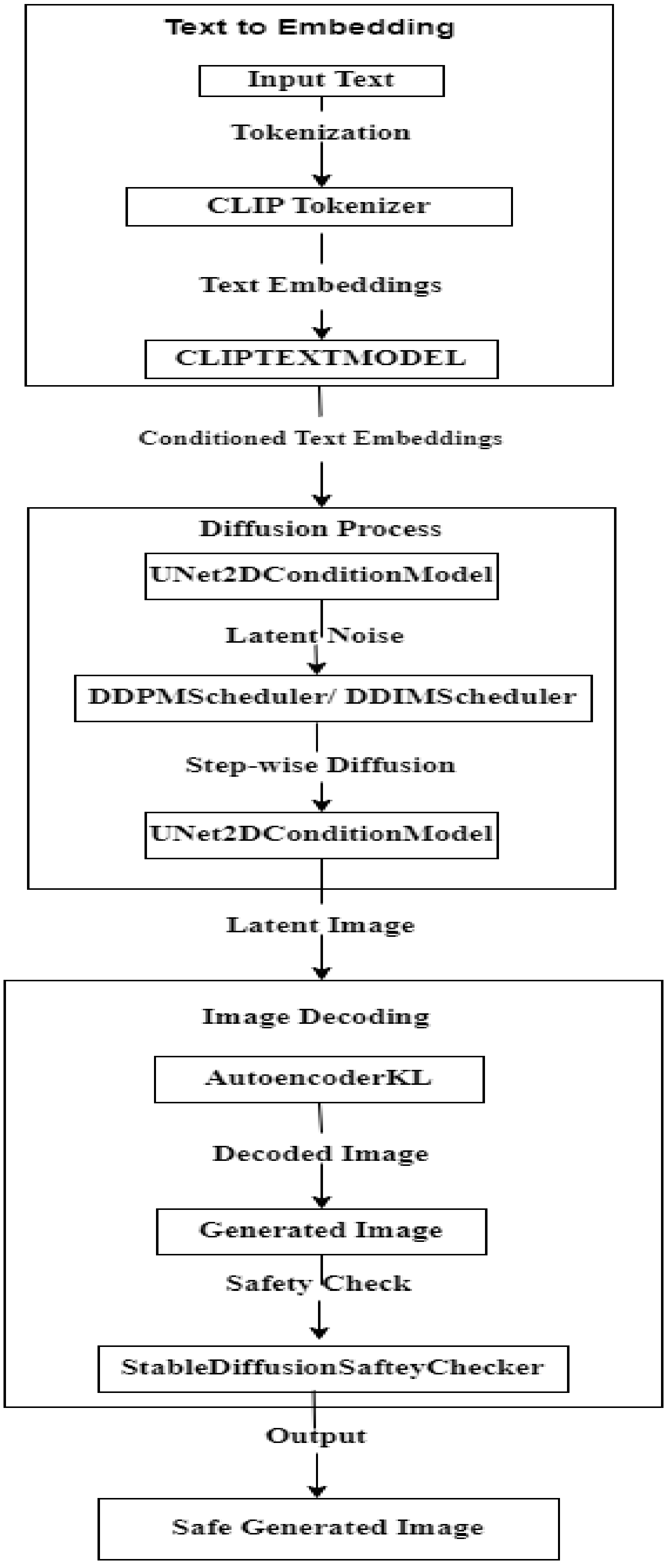


Figure 3.1: Overview of the Stable Diffusion Model Pipeline for Generating Images from Textual Descriptions.

Based on the context of this chapter, the suggested approach can be divided into the following elements: To begin with, the design of the proposed method is given where the difficulties and goals connected with image generation from the text are described. In this segment, the groundwork is laid for the following methodological descriptions by pinpointing the particular concerns the study intends to tackle. After that, the methodology is described, beginning with the models used in the Stable Diffusion pipeline, namely the pre-trained models.

3.2 Design of The Proposed Method

The general workflow of the Stable Diffusion model aimed at flower generation consists of several vital steps, which are described below to match the proposed Stable Diffusion model effectively and generate high-quality images from textual descriptions. The used methodology helps to minimize computational costs and time while training deep neural networks as it relies on pre-trained models. One of the essential features of the implementation of the proposed methodology is the iterative feedback procedure.

This section will describe the steps: data preparation for model construction, model training, and selection, and focuses on the part concerning model improvement based on performance assessment which is executed in the Google Colab Pro Plus environment. The following flowchart Figure (3.2) outlines all the steps required to follow the algorithm starting from a general initialization and preprocessing of the data up to the training loop, and the evaluation of the generated images using FID, ISmetrics:

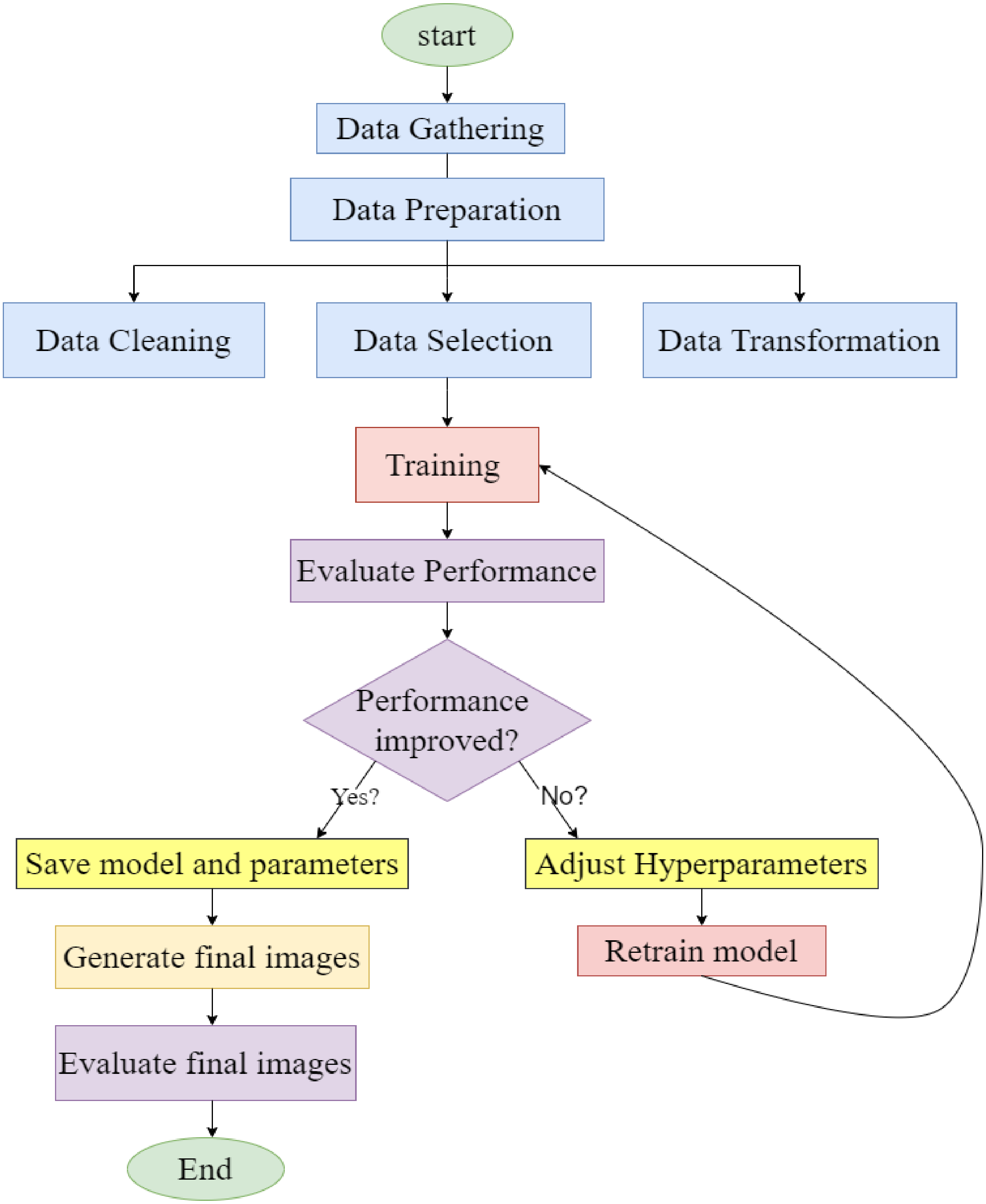


Figure 3.2: Methodology Flowchart for Stable Diffusion Model with Iterative Refinement.

The pseudocode below encapsulates a detailed and cyclical workflow for training and testing the Stable Diffusion model concerning flower image generation. It underlines the necessity of initialization with the pre-trained models, the strict division into the training and test phases, and the evaluation of the results with the help of professional criteria. The aspect of iteration guarantees that the model keeps on acquiring and updating the relevant information hence providing quality images generated from textual descriptions. It plays a key role in constructing the systematic and efficient model for the transformation of textual descriptions to visually good and semantically correct images. The pseudocode has been utilized as a handy guide to have a practical references in implementing the Stable Diffusion model for flower-based generation. Finally, the steps or the evaluation metrics that are used in the proposed model will be explained. The utilized metrics are used to measure the effectiveness of the proposed approach which guarantees the qualitative and semantically meaningful of the synthesized images. Algorithm 3.1 shows the Stable Diffusion Model Training and Evaluation for Flower Generation.

Algorithm 1 Stable Diffusion Model Training and Evaluation for Flower Generation

Require: Pre-trained models (Tokenizer, Text Encoder, VAE, U-Net), Noise Scheduler, Training Data (text prompts, images)

Ensure: Trained Stable Diffusion Model, Generated Images

```
Initialization:
Load pre-trained Tokenizer, Text Encoder, VAE, U-Net
Construct Stable Diffusion Pipeline
Training:
for each record in Training Data (text prompts, images) do
    Tokenize input text prompts
    Encode input images into latent space via VAE
    Add Gaussian noise to latents
    for each iteration in noise scheduler timesteps do
        Predict noise using U-Net
        Compute mean squared error (MSE) loss
        Perform backpropagation and update model parameters
    end for
end for
Testing:
for each record in Testing Data (text prompts) do
    Define text prompt for image synthesis
    Tokenize the prompt
    Initialize latent vectors with random noise
    for each iteration in noise scheduler timesteps do
        Predict noise using U-Net
        Denoise latent vectors progressively
    end for
    Decode the final latents to generate the image using VAE
    Save or display the generated image
end for
Validation:
Compute Inception Score (IS) for the generated images
Compute Frechet Inception Distance (FID) between generated images and real images
Output:
Input: Text prompt (e.g., "A field of sunflowers.”)
Output: Generated image based on the prompt
```

3.2.1 Dataset Preparation

An outline of the steps to be carried out in order to systematically organize the dataset for training .

The first process in the applicability of the presented approach is data gathering. This entails gathering a dataset of text descriptions and flower images and also making preliminary processing of the same. This step takes care of both the textual and visual representation in a manner that will easily help the model understand it. First of all, setting two empty storage lists to store the images and the relevant text paths. The two lists would be the basic data structure for creating the pairs of images with their corresponding descriptive texts. Use the” iterrows()” function to loop through the table to process each image and find its matching text file. A text file will be created for every image. In the training task, only the first description within the text file will be selected from a text file attached to an image in order to simplify the data format. So, to make sure that each image is matched with one text description, it's added along with the text to empty lists. First, a Data Frame will be created after iteration through the dataset to ensure that data is in structured and tabular form. The data frame shall be saved as a CSV file; this CSV file contains images and texts relating to the images. A dataset class, "FlowerDataset," will be created that enables a dataset to be integrated with PyTorch. This class will handle the dataset to be loaded, sample by sample.

3.2.2 Training The Proposed Model

Training first starts with setting up the computational resources and initializing the model on a GPU to fully use of accelerated computing. The training process has several sub-steps that are performed to adjust the parameters of the model progressively. The training loop is the place where the model iteratively learns to synthesize images that are both of high quality and textually relevant.

3.2.2.1 Hyperparameters Setting

To enhance the efficiency of the training process, curial hyperparameters will be chosen. The hyperparameters encompass the number of training epochs, which are set to 12. This indicates that the model will traverse the entire training dataset 12 times, enabling it to progressively refine its weights and discern significant patterns in the data. A limited number of epochs maintains a reasonable training duration, which is essential for iterative experimentation and tuning. The image resolution is determined as 512×512 pixels, which means that the generated images will have sufficient quality for such purposes as graphic design or fine arts. This decision allows catering to the need for quality imagery while at the same time not requiring significant computational power and memory. The number of warmup steps is 25 steps, in deep learning, there are warmup steps that are aimed at increasing the learning rate from zero to some assigned value during the initial training. This aids in making training more stable such that large weight updates would cause unstable training. Next, the learning rate is set to 1e-6 since a small learning rate is preferred to minimize large jumps of values that affect loss in precision with relation to the data. Selection of the learning rate is very important in the training process; a high learning rate could make the training oscillate and be unstable while a small learning rate makes the training slow to converge to the best solution. The batch size is chosen to be 2, which seems to be quite small. The specifications about the batch size have to do with the available calculation potential and the training speed against update accuracy. In this case, it is advisable to set a small batch size as it will ease model stability during training even though the training will be slow.

3.2.2.2 Models Preparation

The components of the stable diffusion model must be loaded to set up the model and ensure that the following models are transferred to the available hardware (GPU). To make the training process fast and the model could generate high-quality images, pre-trained models are used where each of these models has its objective in the pipeline of the model. To enhance the training process and to reduce the memory requirements Mixed Precision is used. the torch.cuda.amp.GradScaler() will be initialized for managing gradient scaling in mixed precision training, maintaining numerical stability while optimizing memory usage and computation speed. All models are put into the training phase using the ".train() method" that enables gradients to update during backpropagation. These pre-trained components are used to ensure that the training becomes fast and does not require a lot of computational power.

3.2.2.3 Training Loops

The training loops are designed to run for multiple epochs, with each epoch doing forward and backward passes of the model on all batches of the dataset using PyTorch. The training process incorporates techniques such as mixed-precision training and gradient scaling for optimizing performance on graphics processing units (GPUs). It also keeps a record of loss values to assess the model performance and optimizes the model parameters based on those losses. The training will converge, or in other words, reach the number of epochs. For every batch, load the input data, comprising images and descriptions, and move these images to the GPU. Inside "torch.cuda.amp.autocast()" context, a forward pass is done, which enables mixed precision; that is, it saves on memory without losing much of the accuracy.

The pre-trained components include a variational autoencoder(VAE) which is responsible for encoding the images and reconstructing the flower's images into the low dimensional space (latent space). This step will reduce the data dimensionality of the image data while maintaining the needful features of the image. Gaussian noise is added to the latent vectors obtained from the VAE during the training, and the model has to learn to remove this noise. This step is very important for the diffusion process which involves decoding the noisy latent vector to get clear images. The latent representation will serve as the input for the denoising U-Net. The other essential model is the CLIP mode, which plays a significant role in bridging the gap between textual descriptions and visual content generation. CLIP, is designed to learn joint representations for images and text by maximizing the cosine similarity between their embeddings. This model is responsible for tokenizing text descriptions by segmenting the text into words or tokens for the model's processing. These tokens are subsequently converted into text embeddings utilizing the pre-trained CLIP text encoder for use in the U-Net model. The U-net model during the process of training will take noisy latent representation images from the variational autoencoder (VAE), the texts embedding from the Clip model, and the timesteps of the diffusion process as inputs. Random noise is generated and added to the latent vectors to simulate noisy inputs. The U-net predicts the noise that needs to be removed from the latent representation, producing a clear latent representation. These inputs allow the U-Net to denoise the latent images progressively, guided by the semantic content provided in the text embeddings. In simpler terms, the CLIP text embeddings perform as a guide, notifying the U-Net how to generate or enhance images to match the given descriptions. The text embeddings direct the U-Net in predicting the proper noise to remove, hence enhancing the latent vectors toward a more distinct image that matches the text description. The U-Net is the essential generative model tasked with eliminating noise from the latent representations over various timesteps. It gradually enhances the noisy latent vectors, directed by the text embeddings from CLIP, to generate images corresponding to the input text descriptions. The Mean Squared Error (MSE) Loss function is used for calculates the difference between the predicted noise and the actual noise that was added to the latent vectors. By decreasing this loss, the model learns to predict the noise more precisely, which directly leads to improvement the quality of generated images. This loss stimulates the U-Net to predict the noise in the latent vectors with greater precision. By minimizing this loss, the model gradually enhances its denoising capability, generating better-quality images as the training progresses. Once the loss is computed, The (AdamW) optimizer is then used to modify these parameters. AdamW is an alternative to the Adam optimizer that unlinks weight decay from the gradient update, allowing for optimized regularization and supporting the avoidance of overfitting. Mixed precision is applied in this step to decrease memory usage and speed up computation. The gradient scaler guarantees that gradients are effectively processed even though operating at lower precision. The optimizer's step is subsequently updating the learning rate scheduler. Mixed precision allows the model to train in a more effective manner by using lower precision during certain operations, while maintaining higher precision (e.g., float32) where necessary. This results in faster computations and minimized memory usage. After processing all batches within an epoch, the average loss is calculated. The model is saved as the best-performing version if the current epoch accomplishes the lowest loss. The entire pipeline, including the VAE, CLIP model, and U-Net, is stored for future analysis. Saving the best-performing model guarantees that the ideal version of the generative model is maintained. This version can then be used to generate images from new text descriptions.

3.2.3 Testing The Proposed Model

During the testing phase, the model processed random text inputs to generate new images. The testing technique entails utilizing training tools and libraries to assess performance indicators and identify any susceptible areas. The results are as follows, compared with the planned objectives to evaluate the efficiency of the model's duties. The resolution and batch size variables employed during training are likewise utilized in testing to ensure the comparability of outcomes between the two processes. The initial step is reading the CSV file that provides the test data, comprising images and text description. The test data comprises 20 rows to ensure the model's performance while avoiding the computational burden of analyzing the complete dataset. Subsequently, it retrieves image paths and duplicates the actual images into a separate directory to facilitate a subsequent comparison between the authentic and generated images. The stable diffusion model pipeline accepts textual input and generates an image corresponding to that textual input. The generated images are archived for further comparison with authentic ones. The Stable Diffusion pipeline will be setup, loaded into the GPU, and then generate and store the images produced. The images will be loaded and preprocessed by resizing them to (299, 299), preparing them for feature extraction with the Inception v3 model. The features extracted from the last pooling layer of the Inception-v3 Network will be delineated since these features are utilized for computing the FID score. The Inception v3 model is a prominent neural network utilized for image recognition, and its activations serve as feature representations for both actual and generated images for calculating the FID score. The Partial Inception Network is employed to extract characteristics from a batch of images. These attributes are subsequently employed to compute the mean and covariance matrix. These characteristics offer a way to quantitatively compare genuine and falsified image graphs.

3.2.4 Evaluating the results of The Proposed Model

The Fréchet Inception Distance (FID) quantifies the resemblance between generated images and authentic ones. The statistical distributions (mean and covariance) of the characteristics from authentic and synthesized images will be calculated. A reduced FID signifies that the produced images more closely resemble the authentic images. The Inception Score (IS) assesses the realism and diversity of generated images. It employs predictions from the Inception v3 model to calculate the KL divergence between class predictions for each image. An elevated IS signifies superior quality and increased diversity. Iterations over several epochs of the generative model's training procedure, incorporating synthetic images produced at each epoch. The FID and IS are computed for the images produced at each epoch, and the results are recorded, enabling the monitoring of enhancements in the quality of generated images during the training process. During this phase, model weights remain unchanged. The weights are fixed to assess the model using the knowledge acquired during the training phase. The testing step follows the training phase, aiming to ascertain the model's efficacy by evaluating it on previously unencountered data.

If the performance metrics show that the model has not evolved better with the help of new, added features or if any parameter related to the performance of the model fails to improve, then the hyper-parameters are modified and the model is retrained. Such an approach of refining the content in cycles guarantees constant enhancement of the results. Parameters are adjusted to improve the training process including; learning rate, size in each batch, and number of iterations. Thus, the new hyper-parameters are used to retrain the model, and it is tested on its performance. The performance results are collected and compared to the baseline and if the desired improvement is not reached, then the loop is run again.

3.3 Summary

This chapter presented a comprehensive methodology for training and evaluating a Stable Diffusion model designed to generate high-quality images of flowers from textual descriptions. The methodology aims to address key challenges in traditional text-to-image models, such as high computational costs and extensive training times while maintaining or improving image quality. The chapter began with an introduction to the significance of generative models in artificial intelligence, highlighting the advancements they have brought to various fields.

This study concentrates on finding an optimal model, which in this case entails alleviating the need for a lot of computations while still producing high-quality images. To reach this goal, the proposed method is based on the use of pre-trained models and presents a structured approach.

The goals were described in detail, focusing on the necessity to reduce computational costs, decrease training time, keep or raise image quality, guarantee the scalability and accessibility of algorithms, and develop a solid evaluation procedure. Every process that was followed in this study was explained in detail right from the way the data was prepared all through to the point of evaluation. One of the important approaches during the implementation of the work was the process of cyclical improvement. Sometimes if the evaluation metrics showed that there has not been an improvement in the performance of the model, new hyperparameters were introduced, and the model was trained again. This cycle was repeated often to make improvements throughout the learning process and other hyper-parameters like learning rate, batch size, and epochs were adjusted.

CHAPTER FOUR

RESULTS AND DISCUSSION

4.1 Introduction

This chapter presents the experimental results and discusion of our study on fine-tuning the Stable Diffusion model for improved image generation from textual descriptions. Building upon the methodology outlined in Chapter 3, we conducted a series of experiments to evaluate the performance of our fine-tuned model compared to the base model.

4.2 Dataset

Every machine learning model is trained and evaluated using data, frequently in the form of static datasets. The features of these datasets have a fundamental impact on the behavior of the model; for example, a model is unlikely to perform well in the real world if its deployment context does not match its training or evaluation datasets, or if these datasets reflect unwelcome societal biases [69].

A watershed moment in the Deep Learning revolution that reshaped Computer Vision (CV) and AI in general occurred with the introduction of the ImageNet dataset. Researchers in the fields of computer vision and image processing used small datasets like CalTech101 (9k photos), PASCAL-VOC (30k images), LabelMe (37k images), and the SUN (131k images) dataset to build image classification models before ImageNet.

With more than 14 million images distributed among 21,841 synsets and 1,034,908 bounding box annotations, ImageNet brought a new concept of scale to the table. For the ImageNet Large Scale Visual Recognition task (ILSVRC), the ImageNet-1k dataset—a subset of 1.2 million photos over 1000 classes—was created from this larger dataset. This dataset is commonly referred to as ILSVRC-2012).

For a text-to-image generation, it is needed to have a dataset containing an image and its description as a text. However, in this section, we will show that this field of study typically involves the use of several categories of datasets. As we need to convert the text into images, the database must contain images along with their captions. Therefore, this section will look into the numerous categories of datasets used in this research field [41].

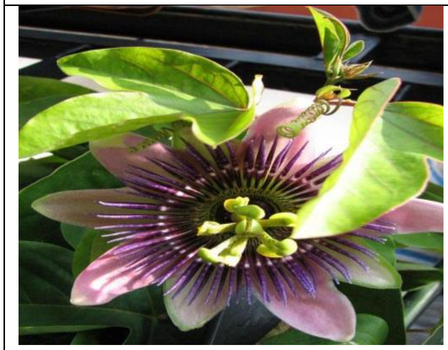
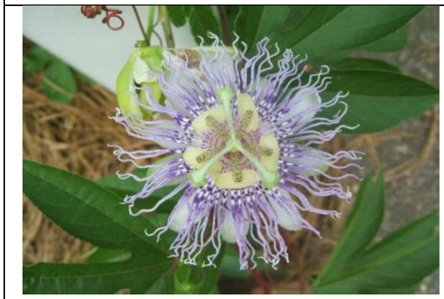
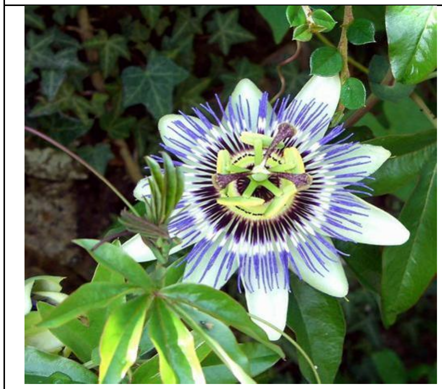
The Caltech-UCSD Birds-200-2011 dataset comprises 11,788 photographs of birds, which have been categorized into 200 distinct groups. Each dataset offers five interpretations for each image.

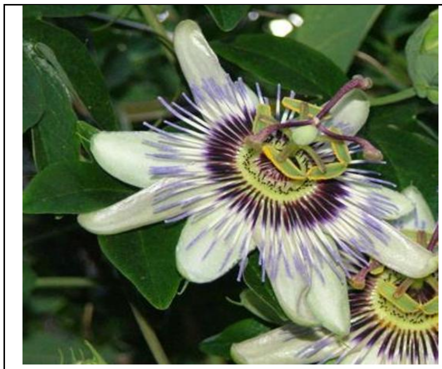
Table 4.1 : Text-to-image generation datasets

Dataset	**Size**	**Language of text**	**Type of Images**
MS COCO	300,000 images	English	Objects in diverse environments
CUB	11788 images	English	Birds
Oxford 102	8189 images	English	Flowers
Flickr30k	over 31,000 images	English -German -Chinese	Daily activities and occurrences

The considered database, Oxford flowers-102, comprises 103 various classes of flowers. Each flower sample is described in detail, many characteristics are described; whether the flower is native, and its texture, the outline of the border formation of petals, the overall formation of the space, and the color pattern [32]. Table (4.2) shows samples from the dataset used in this study.

Table 4.2: Samples Of Images And The First Description Associated With The Image From Dataset [69]

Images	**Description**
	**outer petals are green in color and larger, inner petals are needle-shaped**
	**there are several shapes, sizes, and colors of petals on this complex flower.**
	**the stamen are towering over the stigma which cannot be seen.**

	**this flower is white and purple in color, with petals that are oval shaped.**

4.3 Hardware Requirements

The training of the model utilizes a high-performance computational environment, specifically a Google Colab instance with an NVIDIA T4 Tensor Core GPU. The T4 GPU, based on the Turing architecture, includes 2560 CUDA cores, 320 Tensor cores, and 16 GB of GDDR6 memory, offering peak performance of 8.1 TFLOPS (FP32) and 130 TOPS (INT8). This GPU is optimized for machine learning and AI workloads, providing high throughput for training and inference tasks. The instance also features a high RAM configuration with a capacity exceeding 25 GB, which is crucial for loading large datasets and models, as well as performing intermediate computations during training.

The CPU in this setup is a high-performance virtual CPU with multiple vCPUs (typically 2-4 cores), efficiently handling data preprocessing, augmentation, and other CPU-bound tasks. The fast SSD storage ensures quick access to large datasets and model checkpoints, reducing I/O bottlenecks during training. The software stack includes a Linux-based operating system provided by Google Colab, with PyTorch as the deep learning framework, supported by CUDA for GPU acceleration.

Table 4.3: Specifications of the Computational Resources Used for Training the Stable Diffusion Model

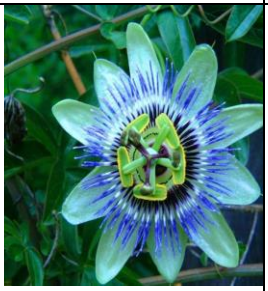
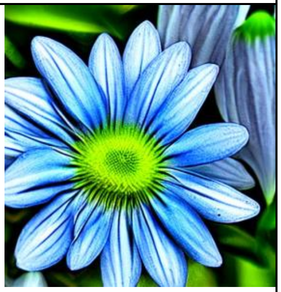
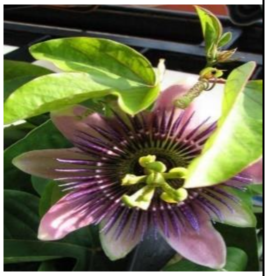
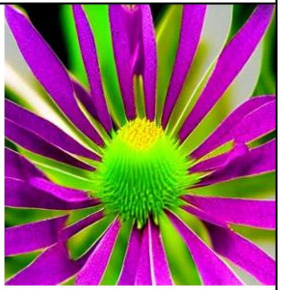
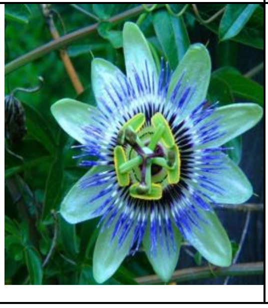
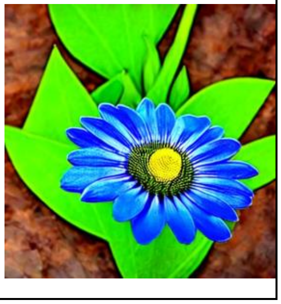
Component	Details
GPU	NVIDIA T4 Tensor Core GPU
Architecture	Turing
CUDA Cores	2560
Tensor Cores	320
Memory	16 GB GDDR6
Peak Performance	8.1 TFLOPS (FP32), 130 TOPS (INT8)
RAM	High RAM configuration (25 GB+)
CPU	High-performance virtual CPU (2-4 vCPUs)
Storage	Fast SSD storage
Operating System	Linux-based environment provided by Google Colab
Deep Learning Framework	PyTorch with CUDA support for GPU acceleration
Additional Libraries	pandas, numpy, PIL, transformers, diffusers

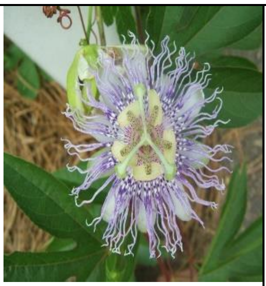
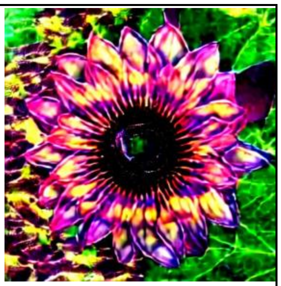
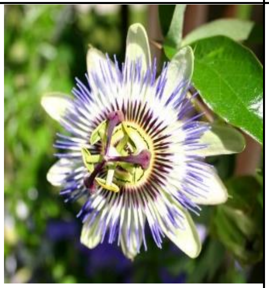
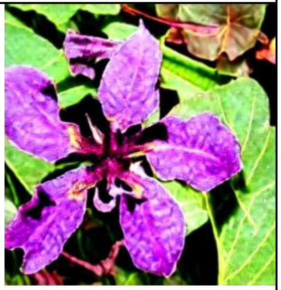
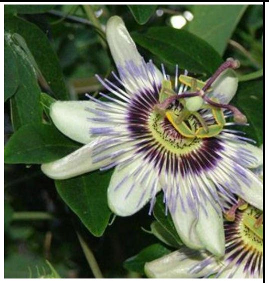
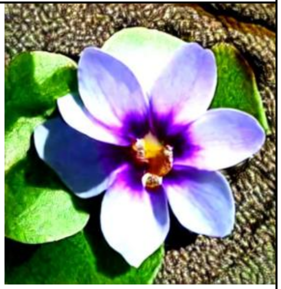
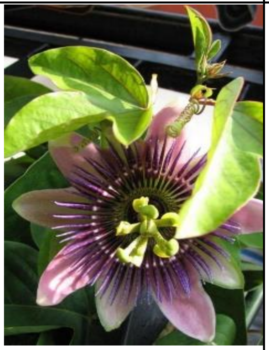
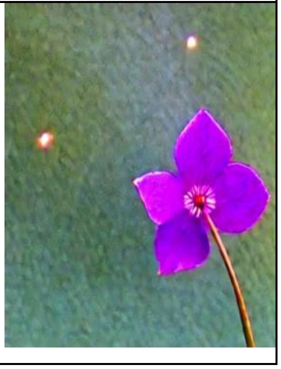
4.4 Results and discussion

The experiments were initialized using a pre-trained Stable Diffusion model from the CompVis/stable-diffusion-v1-4 repository. This served as our base model and starting point for fine-tuning. Additionally, libraries for data handling (pandas, numpy), image processing (PIL), and model-specific modules (transformers, diffusers) are utilized. The training phase involves utilizing the Stable Diffusion model to generate high-quality images based on textual descriptions. This phase employs specific settings and parameters to ensure the accuracy and efficiency of the results. In this experiment, the number of epochs is set to 12, meaning the model will iterate through the entire training dataset 12 times, allowing the model to progressively refine its weights and learn important patterns in the data.

The image resolution is determined as 512×512 pixels, which means that the generated images will have sufficient quality for such purposes as graphic design or fine arts. This decision allows catering to the need for quality imagery while at the same time not requiring significant computational power and memory, the latter of which increases with the resolution. The testing results of the proposed model using the hyper-parameters are explained in the following table:

Table 4.4: Results of the proposed model

Text	**Original image**	**Generated image**
This Flower Is Blue And Green In Color, With Petals That Are Oval Shaped.		
Outer Petals Are Green In Color And Klarger,Inner Petals Are Needle Shaped		
This Flower Is Blue And Green In Color, With Petals That Are Oval Shaped.		

There Are Several Shapes, Sizes, And Colors Of Petals On This Complex Flower.		
Prominent Purple Stigma ,Petals Are White Color		
This Flower Is White And Purple In Color, With Petals That Are Oval Shaped.		
Outer Petals Are Green In Color And Klarger,Inner Petals Are Needle Shaped		

The testing phase succeeds the training phase which seeks to determine that the model works well and tests it on data it has not been trained on. Hypotheses are made during the test, and the quality of images which are generated from new textual inputs is assessed according to specified metrics such as accuracy, clarity, and relevance to the texts.

4.4.1 Experiment 1: Fine-Tuning Performance Across Epochs

The performance of our fine-tuned model across different epochs is evaluated. The results are presented in Table 4.5

Table 4.5: Fine-Tuned Stable Diffusion Model Performance Across Epochs

Model	IS Mean	IS Std	FID
Base Model	1.5960649	0.016032545	248.748256
Epoch 1	1.6064876	0.021265263	252.899013
Epoch 2	1.6140872	0.019562684	252.355621
Epoch 3	1.6145291	0.018562684	252.133787
Epoch 4	1.6163372	0.018729529	251.981384
Epoch 5	1.6181474	0.018896375	251.828981
Epoch 6	1.6199575	0.01906322	251.676578
Epoch 7	1.6217676	0.019230066	251.524175
Epoch 8	1.6235778	0.019396911	251.371772
Epoch 9	1.6244419	0.019563757	251.219369

Table (4.5) presents the performance metrics of a fine-tuned Stable Diffusion model across nine epochs, compared to a base model. The metrics used are Inception Score (IS) Mean and Standard Deviation, and Fréchet Inception Distance (FID). The observed improvement, albeit limited, suggests that the fine-tuning process is exerting a positive influence on the model's capacity to generate realistic and diverse images. An analysis of the Inception Score (IS) Mean reveals a consistent enhancement from the base model (1.5960649) through each epoch, culminating in a maximum of 1.6244419 at Epoch 9. This gradual increase indicates that the model is generating increasingly diverse and high-quality images as training progresses. The IS Standard Deviation exhibits an initial increase from the base model (0.016032545) to Epoch 1 (0.021265263), followed by a general decline in subsequent epochs. This trend suggests that while the model's performance is improving overall, there is also a stabilization in the consistency of the generated images' quality and diversity. Its worthing noting that, the FID scores present a different pattern. The base model starts with the lowest FID score (248.748256), which is generally considered better as lower FID scores indicate greater similarity between the generated images and the real dataset. The FID scores for the fine-tuned model are consistently higher, starting at 252.899013 in Epoch 1 and gradually decreasing to 251.219369 by Epoch 9. This apparent inconsistency between improving IS scores and worsening FID scores could indicate that while the model is generating more diverse and higher quality images (as suggested by the IS), these images may be diverging slightly from the original dataset's distribution (as indicated by the FID). This means that the fine-tuning process allows the model to generate more creative or varied images, but at the cost of strict adherence to the training set's characteristics. The gradual decrease in FID scores across epochs suggests that the model is slowly adjusting back toward the original distribution as training progresses. If this trend continues, it's possible that with more epochs, the FID score might eventually drop below the base model's score, potentially achieving both improved diversity/quality and better alignment with the original dataset.

Overall, these results highlight the complex nature of image generation models and the potential trade-offs between different aspects of performance during the fine-tuning process.

4.4.1.1 Key Observations of Experiment 1:

1. The Inception Score (IS) showed a general trend of improvement across epochs, with the highest mean value of 1.6244419 achieved at epoch 9.
2. The FID scores for the fine-tuned models were consistently higher than the base model, indicating that the fine-tuned models may have diverged slightly from the original distribution of real images.
3. The base model achieved the lowest FID score of 248.748256, suggesting it maintained the closest similarity to real images in terms of overall distribution.

4.4.2 Experiment 2: Comparison of Fine-Tuned Model with Base Model

The performance of our fine-tuned model (labeled as "Stable diffusion finetune") is compared with the base model (labeled as "Stable diffusion V4"). The results are presented in Table 4.6.

Table 4.6 Comparison of Fine-Tuned and Base Models

Model	IS Mean	IS Std	FID
Stable diffusion V4	1.61	0.02	251.22
Stable diffusion finetune	1.60	0.04	212.52

Table (4.6) presents a comparison between the base Stable Diffusion V4 model and the fine-tuned version, evaluating their performance using three key metrics: Inception Score (IS) Mean, IS Standard Deviation, and Fréchet Inception Distance (FID). This comparison provides valuable insights into the effects of the fine-tuning process on the model's performance. There's a notable increase in the IS Standard Deviation from 0.02 in the base model to 0.04 in the fine-tuned model. This doubling of the standard deviation suggests that the fine-tuned model produces a wider range of IS scores. This could indicate that the fine-tuned model is generating a more diverse set of images, with some potentially being of higher quality than the base model's outputs, while others might be of lower quality. This increased variability could be a result of the model adapting to specific characteristics of the fine-tuning dataset. The most significant change is observed in the FID score. The fine-tuned model achieves a substantially lower FID score of 212.52 compared to the base model's 251.22. This marked improvement (a reduction of about 15.4%) is a strong indicator that the fine-tuned model is generating images that are more similar to the real images in the target dataset. The lower FID score of the fine-tuned model indicates that it's generating images that are more closely aligned with the statistical properties of the real images in the dataset. This could mean that the fine-tuned model is better at capturing specific features, styles, or distributions present in the flower dataset used for fine-tuning. The increased variability in the IS scores (higher standard deviation) coupled with the improved FID score suggests that the fine-tuned model might be exploring a wider range of image possibilities while still maintaining overall better alignment with the target dataset. This could be particularly beneficial if the goal is to generate diverse yet realistic flower images.

It's important to note that while the IS Mean has slightly decreased, the magnitude of this decrease is small compared to the significant improvement in the FID score. This suggests that the trade-off is likely worthwhile, especially if the primary goal is to generate images that closely match the characteristics of the target dataset. The results present an interesting trade-off in the model's performance after fine-tuning. While the IS Mean has slightly decreased and its variability has increased, the substantial improvement in the FID score suggests that the fine-tuning process has been largely successful in adapting the model to the specific characteristics of the target dataset.

4.4.2.1 Key Observations for Experiment 2:

1. The Inception Scores (IS) for both models were comparable, with the base model (Stable diffusion V4) slightly outperforming the fine-tuned model.
2. The FID score of the fine-tuned model got down to 212. 52, which was significantly better as compared to the FID score of the base model (251. 22), testifying that the images produced by the fine-tuned model were closer to the real images in the training set. Nevertheless, the finetuning of the model resulted in a higher standard deviation connected to the quality and the variety of the generated IS, which means that it was more variable.

CHAPTER FIVE

CONCLUSION AND FUTURE WORK

5.1 Overview

The conclusion of the work in this study and the areas for future work are described in this chapter.

5.2 Conclusion

The outcome of the study in this led to the following important conclusions:

1. Fine-tuning Impact: The fine-tuning process clearly showed how it can enhance the model's optimization, and therefore make images that are closer to the target distribution embedded in the synthetic dataset, as given by the lower FID in Experiment
2. Trade-offs in Performance Metrics: Although the FID was further reduced indicating that our fine-tuning had a positive effect we could see that the IS was either at par or lower than the base model. This implies that there is a trade-off between image quality and flexibility/improvement in the mentioned criteria.
3. Epoch-wise Performance: The results found in the epoch comparison of IS scores in Experiment 1 reveal that the fine-tuning process was, on the whole, beneficial in gradually enhancing the model's global ability to produce better and more diverse digital imagery.
4. Hyper-parameter Sensitivity: In this case, the results of the model in epochs indicate that proper tuning of the hyper-parameters is critical in the fine-tuning stage.

From the conducted experiments, it became clear that attempts to fine-tune the Stable Diffusion model can have a positive effect on the generation of images closer to the original image. However, this process involves careful consideration of trade-offs between different performance metrics and requires meticulous hyper-parameter tuning.

It was observed that the fine-tuned model holds the capability of a lower FID score, which gives an impression of a better likelihood of attaining a higher image distribution similarity to the targeted dataset. In such a case, the presented approach could be useful, as it is most beneficial for cases requesting similarity to a certain set of real images.

Eventually, research regarding this topic should aim at fine-tuning the Generator to further Inception Scores or even equal that of the FID while at the same time attaining an FID score lower than this study. Also, it can be stated that the refinements of the hyper-parameters optimization algorithm can contribute to the further enhancement of the model.

5.3 Future Work

As for the next step, further fine-tuning research has to be aimed at enhancing or at least preserving Inception Scores while achieving the FID scores as low as possible. Also, there is a possibility of testing even more complex hyper-parameter optimization methods that would have a positive effect on the overall performance of the model. This model could be used for video generation from text because of the robustness of the diffusion model.

REFERENCES

[1] Stuart Russell and Peter Norvig, *Artificial Intelligence A Modern Approach Fourth Edition*, Fourth. 2021.

[2] Haileleol Tibebu, Aadil Malik, and Varuna De Silva, "Text to Image Synthesis using Stacked Conditional Variational Autoencoders and Conditional Generative Adversarial Networks," *INTELLIGENT COMPUTING: PROCEEDINGS OF THE 2022 COMPUTING CONFERENCE*, pp. 1–17, 2022.

[3] J. Oppenlaender, "The Creativity of Text-to-Image Generation," in *ACM International Conference Proceeding Series*, Association for Computing Machinery, Nov. 2022, pp. 192–202. doi: 10.1145/3569219.3569352.

[4] M. Ding *et al.*, "CogView: Mastering Text-to-Image Generation via Transformers," May 2021.

[5] P. Dhariwal and A. Nichol, "Diffusion Models Beat GANs on Image Synthesis," May 2021.

[6] S. Ajay Bankar, S. Ket, and R. Gandhi, "An Analysis of Text-to-Image Synthesis." Accessed: Sep. 20, 2024.

[7] A. Kapoor, A. Gulli, S. (Software engineer) Pal, and F. Chollet, *Deep learning with TensorFlow and Keras*.

[8] F. S. Zadeh, S. Molani, M. Orouskhani, M. Rezaei, M. Shafiei, and H. Abbasi, "Generative Adversarial Networks for Brain Images Synthesis: A Review."

[9] H. Huang, P. S. Yu, and C. Wang, "An Introduction to Image Synthesis with Generative Adversarial Nets," Mar. 2018.

[10] C. Zhang and Y. Peng, "Stacking VAE and GAN for Context-aware Text-to-Image Generation," in *2018 IEEE Fourth International Conference on Multimedia Big Data (BigMM)*, IEEE, Sep. 2018, pp. 1–5. doi: 10.1109/BigMM.2018.8499439.

[11] H. Zhang *et al.*, "StackGAN: Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks," Dec. 2016.

[12] B. Li, X. Qi, T. Lukasiewicz, and P. H. S. Torr, "Controllable Text-to-Image Generation," Sep. 2019.

[13] T. Xu *et al.*, "AttnGAN: Fine-Grained Text to Image Generation with Attentional Generative Adversarial Networks," Nov. 2017.

[14] IEEE Computer Society. Technical Committee on Multimedia Computing, IEEE Computer Society. Technical Committee on Semantic Computing, and Institute of Electrical and Electronics Engineers, *2018 IEEE Fourth International Conference on Multimedia Big Data (BigMM) : 13-16 Sept. 2018.*

[15] B. Rajesh, N. Dusa, M. Javed, S. R. Dubey, and P. Nagabhushan, “T2CI-GAN: Text to Compressed Image generation using Generative Adversarial Network,” Oct. 2022.

[16] S. J. D. Prince, “Understanding Deep Learning,” 2023.

[17] D. (Business consultant) Foster and K. Friston, “*Generative deep learning : teaching machines to paint, write, compose, and play” May*. 2023.

[18] E. Mansimov, E. Parisotto, J. L. Ba, and R. Salakhutdinov, “Generating Images from Captions with Attention,” Nov. 2015.

[19] Y. Zhang, L. Jiang, G. Turk, and D. Yang, “Auditing Gender Presentation Differences in Text-to-Image Models,” Feb. 2023.

[20] N. Carlini *et al.*, “Extracting Training Data from Diffusion Models,” Jan. 2023.

[21] F. Mahlow, A. F. Zanella, W. A. C. Castañeda, and R. A. Sarzi-Ribeiro, “Illustrating Classic Brazilian Books using a Text-To-Image Diffusion Model,” Aug. 2024.

[22] Ashly Correya and Amrutha N, “Text to Image Conversion using Stable Diffusion,” *Indian Journal of Data Mining (IJDM)*, May 2024, doi: 10.54105/ijdm.A1639.04010524.

[23] S. An *et al.*, *Rethinking the Invisible Protection against Unauthorized Image Usage in Stable Diffusion*. Philadelphia, PA, USA : the Proceedings of the 33rd USENIX Security Symposium. August 14–16, 2024.

[24] Z. Li, L. Gao, and C. Wu, “Text-to-Model: Text-Conditioned Neural Network Diffusion for Train-Once-for-All Personalization,” May 2024.

[25] P. Avhad, P. Barman, and K. Wasnik, “WordCanvas: Text-to-Image Generation,” *International Journal of Scientific Research in Engineering and Management*, 2024, doi: 10.55041/IJSREM32152.

[26] Y. Qu, M. Backes, X. Shen, S. Zannettou, X. He, and Y. Zhang, “Unsafe Diffusion: On the Generation of Unsafe Images and Hateful Memes From Text-To-Image Models,” in *CCS 2023 - Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security*, Association for Computing Machinery, Inc, Nov. 2023, pp. 3403–3417. doi: 10.1145/3576915.3616679.

[27] M. Zhou, Z. Wang, H. Zheng, and H. Huang, “Long and Short Guidance in Score identity Distillation for One-Step Text-to-Image Generation,” Jun. 2024.

[28] H. Ravi, S. Kelkar, M. Harikumar, and A. Kale, “PRedItOR: Text Guided Image Editing with Diffusion Prior,” Feb. 2023.

[29] C. Zhang, C. Zhang, M. Zhang, and I. S. Kweon, “Text-to-image Diffusion Models in Generative AI: A Survey,” Mar. 2023.

[30] P. Dhariwal and A. Nichol, “Diffusion Models Beat GANs on Image Synthesis,” May 2021.

[31] S. Sharma, D. Suhubdy, V. Michalski, S. E. Kahou, and Y. Bengio, “ChatPainter: Improving Text to Image Generation using Dialogue,” Feb. 2018.

[32] A. Singh and S. Agrawal, “CanvasGAN: A simple baseline for text to image generation by incrementally patching a canvas,” Oct. 2018.

[33] X. Ouyang, X. Zhang, D. Ma, and G. Agam, “Generating Image Sequence from Description with LSTM Conditional GAN,” Jun. 2018.

[34] T. Xu *et al.*, “AttnGAN: Fine-Grained Text to Image Generation with Attentional Generative Adversarial Networks,” Nov. 2017.

[35] G. Kim, T. Kwon, and J. C. Ye, “DiffusionCLIP: Text-Guided Diffusion Models for Robust Image Manipulation,” Oct. 2021.

[36] H. Schulze, D. Yaman, and A. Waibel, “CAGAN: Text-To-Image Generation with Combined Attention GANs,” Apr. 2021, doi: 10.1007/978-3-030-92659-5_25.

[37] S. Gu *et al.*, “Vector Quantized Diffusion Model for Text-to-Image Synthesis,” Nov. 2021.

[38] T. Qiao, J. Zhang, D. Xu, and D. Tao, “Learn, Imagine and Create Text-to-Image Generation from Prior Knowledge.” 2019.

[39] Z. Xue *et al.*, “RAPHAEL: Text-to-Image Generation via Large Mixture of Diffusion Paths,” May 2023.

[40] Z. Pan, X. Zhou, and H. Tian, “Arbitrary Style Guidance for Enhanced Diffusion-Based Text-to-Image Generation,” Nov. 2022.

[41] Y. Cao *et al.*, “A Comprehensive Survey of AI-Generated Content (AIGC): A History of Generative AI from GAN to ChatGPT,” Mar. 2023.

[42] Siddiqui and Nazia, “Scholarship at UWindsor Comparative Study of Generative Models for Text-to-Image generation,” 2023.[43] A. Ulhaq, N. Akhtar, and G. Pogrebna, “Efficient Diffusion Models for Vision: A Survey,” Oct. 2022.

[44] Y. Liu, Z. Li, M. Backes, Y. Shen, and Y. Zhang, “Watermarking Diffusion Model,” May 2023.

[45] A. Bansal *et al.*, “Cold Diffusion: Inverting Arbitrary Image Transforms Without Noise,” Aug. 2022.

[46] L. B. T. C. Eric Heitz, “ITERATIVE α-(DE)BLENDING: LEARNING A DETER-MINISTIC MAPPING BETWEEN ARBITRARY DENSITIES,” 2023.

[47] A. Jolicoeur-Martineau, K. Fatras, K. Li, and T. Kachman, “Diffusion models with location-scale noise,” Apr. 2023.

[48] Sudershan Manasvi Malhar, “DALL. E 2,” *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, pp. 48–56, Sep. 2023, doi: 10.32628/cseit239052.

[49] R. Zbinden, “Implementing and Experimenting with Diffusion Models for Text-to-Image Generation,” Sep. 2022.

[50] C. Saharia *et al.*, “Photorealistic Text-to-Image Diffusion Models with Deep Language Understanding,” May 2022.

[51] Martin Musiol, “Praise for Generative AI,” 2024.

[52] J. Rando, D. Paleka, D. Lindner, L. Heim, and F. Tramèr, “Red-Teaming the Stable Diffusion Safety Filter,” Oct. 2022.

[53] A. Vaswani *et al.*, “Attention Is All You Need,” Jun. 2017.

[54] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, “NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis,” Mar. 2020.

[55] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition,” Dec. 2015.

[56] A. Habibian, T. van Rozendaal, J. M. Tomczak, and T. S. Cohen, “Video Compression With Rate-Distortion Autoencoders,” Aug. 2019, doi: 10.1109/ICCV.2019.00713.

[57] X. Chen *et al.*, “Variational Lossy Autoencoder,” Nov. 2016.

[58] D. J. Rezende and S. Mohamed, “Variational Inference with Normalizing Flows,” May 2015.

[59] A. Wehenkel and G. Louppe, “Diffusion Priors In Variational Autoencoders,” Jun. 2021.

[60] R. Malik Thaarup, “On Learning Useful Variational Autoencoder Representations: Applications in Audio Modelling and Hearing Loss Treatment,” APA, 2022.

[61] T. B. Brown *et al.*, “Language Models are Few-Shot Learners,” May 2020.

[62] Y. Liu *et al.*, “RoBERTa: A Robustly Optimized BERT Pretraining Approach,” Jul. 2019.

[63] T. M. Dinh, R. Nguyen, and B.-S. Hua, “TISE: Bag of Metrics for Text-to-Image Synthesis Evaluation,” Dec. 2021.

[64] Z. Ji, W. Wang, B. Chen, and X. Han, “Text-to-Image Generation via Semi-Supervised Training,” in *2020 IEEE International Conference on Visual Communications and Image Processing, VCIP 2020*, Institute of Electrical and Electronics Engineers Inc., Dec. 2020, pp. 265–268. doi: 10.1109/VCIP49819.2020.9301888.

[65] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, “GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium,” Jun. 2017.

[66] S. Sharma, K. Guleria, S. Kumar, and S. Tiwari, “Deep Learning based Model for Detection of Vitiligo Skin Disease using Pretrained Inception V3,” *International Journal of Mathematical, Engineering and Management Sciences*, vol. 8, no. 5, pp. 1024–1039, 2023, doi: 10.33889/IJMEMS.2023.8.5.059.

[67] G. Meena, K. K. Mohbey, and S. Kumar, “Sentiment analysis on images using convolutional neural networks based Inception-V3 transfer learning approach,” *International Journal of Information Management Data Insights*, vol. 3, no. 1, Apr. 2023, doi: 10.1016/j.jjimei.2023.100174.

[68] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, “Improved Techniques for Training GANs,” Jun. 2016

[69] M.-E. Nilsback, “An automatic visual Flora-segmentation and classification of flower images,” 2009.

الخلاصة

قد حقق توليد الصور المدعوم بالنص قفزة هائلة نحو أن يصبح ظاهرة سائدة. مع أنظمة تحويل النص إلى صورة، يمكن لأي شخص إنشاء صور رقمية وأعمال فنية و هذا يثير مسألة ما إذا كان توليد النص إلى صورة هو عملاً إبداعيًا. لقد ساهمت الأنظمة التوليدية كثيرًا في تطوير الذكاء الاصطناعي من خلال توليد صور واقعية إلى حد ما من النص. تم استخدام أنظمة توليد الصور بأستخدام النص في أشكال ومجالات مختلفة في النطاق بما في ذلك ، على سبيل المثال لا الحصر ، الأعمال الفنية والتصاميم وأخذ عينات البيانات والترفيه. تم إجراء العديد من الدراسات حول توليد الصور من النص حيث تم اقتراح العديد من تقنيات الذكاء الاصطناعي. ومع ذلك، لا تزال بعض القضايا الحرجة بحاجة إلى الحل، خاصة فيما يتعلق باستهلاك الوقت ووقت التدريب. لذلك، استخدمت الدراسة المقترحة نموذج الانتشار المستقر (SDM) لإجراء تغذية راجعة تكرارية) إذا لم تتحسن مقاييس التقييم وهي درجة البداية (IS) والمسافة الابتدائية فريشيت (FID) يتم ضبط المعلمات الفائقة وتدريب النموذج مرة أخرى .(في هذه الدراسة، يؤدي ضبط نموذج SDM إلى تحسين كبير في توليد الصور التي تشبه الواقع بشكل أكبر. وكذلك، هناك تنازلات بين جودة الصورة ومرونة مقاييس الأداء. تعمل عملية الضبط الدقيق على تحسين القدرة العالمية للنموذج تدريجياً على إنتاج صور رقمية أفضل وأكثر تنوعاً. النموذج الذي تم ضبطه بدقة لديه درجة FID أقل (248.748256)، مما يشير إلى احتمال أكبر لتحقيق تشابه أعلى في توزيع الصور مع مجموعة البيانات المستهدفة. بشكل متقطع، أظهرت نتائج النموذج المحسن درجة FIDأقل (212.52) عند مقارنتها بالنموذج الأساسي (251.22)، مما يشير إلى أن الصور المولدة من النموذج المعدل كانت أقرب إلى التوزيع المستهدف في مجموعة البيانات الاصطناعية .

جامعة كربلاء
كلية علوم الحاسوب وتكنولوجيا المعلومات
قسم علوم الحاسوب

توليد صور المشروطة بالنص باستخدام نماذج الانتشار

رسالة ماجستير
مقدمة الى مجلس كلية علوم الحاسوب وتكنولوجيا المعلومات جامعة كربلاء وهي
جزء من متطلبات نيل درجة الماجستير في علوم الحاسوب

كتبت بوساطة
ساره فائز عبدالغني

بأشراف
أ.م.د اشوان انور عبدالمنعم

1445 هـ

2024 م