Proceedings of the 4th International Conference on Signal Processing and Machine Learning
DOI: 10.54254/2755-2721/45/20241038

AI-based text-to-image synthesis: A review

Zili Wang

University of California, Santa Cruz

zwang400@ucsc.edu

Abstract. The traditional methods of art generation, such as texture synthesis and texture mapping, have been instrumental in crafting digital art for decades. They are used as artistic tools to design and map textures onto 3D models, thereby generating 2D images or animations. However, they can only generate simple, repetitive images. Thanks to the rapid development of deep learning and artificial intelligence, today's text-to-image synthesis (T2IS) models can generate high-quality, realistic images matching the textual description given by the users. This review paper aims to present a comprehensive exploration of groundbreaking AI-based T2IS models in history. We start with an in-depth analysis of the fundamental concepts that underpin T2IS models, followed by an introduction to the primary, or vanilla, models that have served as the foundation for the fields' development. Then, we delve into the examination of several groundbreaking AI-based T2IS applications, from GAN-based to Diffusion-based models, demonstrating their ability to produce high-quality, contextually accurate images from textual descriptions, along with their strengths and weaknesses. In the end, we will discuss the current challenges and potential future directions in the realm of T2IS.

Keywords: Text-to-Image Synthesis, Generative Adversarial Networks, Diffusion Models, Variational Autoencoders, Diffusion Models.

1. Introduction

The technology of Artificial Intelligence Generated Content (AIGC) has grown rapidly and gained huge popularity and social attention due to its astonishing ability to content generation in various forms, such as text, pictures, audio, and video [1]. Many AI content generation products, such as ChatGPT [2], are acknowledged to produce high-quality content at a fast pace. AI-based text-to-image synthesis (T2IS) technologies, as part of AIGC technology, also gained huge popularity due to the dramatic progress that it has achieved. Users can obtain photo-realistic images simply by providing the AI-generated image model with textual descriptions of the image.

The quality of AI-generated art has also been acknowledged by professionals. The first auctioned piece of AI-generated art, the Portrait "Edmond de Belamy", was created with Generative Adversarial Network (GAN) [3]. It was sold for $432,500 at Christie's auction. In 2022, an AI-produced artwork called "Théâtre D'opéra Spatial" by Jason M. Allen (Figure 1) won the art award at the Colorado State Fair's annual art competition. This artwork is created using the AI-based T2IS product Midjourney. Another artwork also generated by Midjourney, "The Electrician" by Boris Eldagsen, clinched the 2023 Sony World Photography Awards, organized by the World Photography Organization. The artist later declined this award because he maintained that the competition couldn't deal with AI-generated artwork.

DOI: 10.54254/2755-2721/45/20241038

On the other hand, the contest organizer admitted their unawareness about the degree to which AI was employed in the creation of the artwork.

Figure 1. Images generated by Midjourney won the award. Left Théâtre D'opéra Spatial". Right "The Electrician".

As shown in Figure 1, the artwork gives the impression of a vintage photograph, depicting two women, with one squatting behind the other. Additionally, there's a hand reaching out towards the woman in the forefront.

This magic of image synthesis is made possible through deep neural networks [4]. In the early stages, image synthesis was not based on deep learning-based techniques. The traditional methods, such as texture synthesis [5] and texture mapping [6], were unable to deliver complex images. Later, the deep learning-based image synthesis models, such as GANs [3], Variational Autoencoders (VAEs) [7], and diffusion generative models [8], were established and offer more precise control over the image generation procedure and the capability to create high-resolution images.

T2IS is a deep-learning-based image synthesis conditioned on text descriptions, which has been a focal point of ongoing research in the field of computer vision. Given a text prompt from users, the models can synthesize high-quality and realistic images like the above; at the same time, the generated images align with the text prompt well. This paper aims to review and discuss the recent findings and techniques related to AI-based T2IS. The papers mentioned are as follows. In Section II, the related concepts of T2IS models are discussed. In Section III, the primary/vanilla models that are used in T2IS models are introduced. Section IV presents several promising T2IS models built upon those primary models. Section V discusses the limitations and future direction of T2IS models. Finally, this paper is concluded in Section VI.

2. Related Concepts

2.1. Artificial Intelligence Generated Content

Artificial Intelligence Generated Content (AIGC) is a method of content generation: a trained model, learning to understand and replicate the statistical distribution of training data [9], is built to automatically generate content with high production efficiency, quantity, and quality. Today, there are three modes of content generation: Professionally-Generated Content (PGC) mode, User-Generated-Content (UGC) mode, and AIGC mode [10]. In PGC mode, content is created by artificial force. The generated content produced by professional groups guarantees high quality; however, reaching the quantity standards requires lots of time and work. In UGC mode, the content is generated by the users through platforms. It provides data availability and simplicity of content generation but loses the quality due to users' large range of skill levels. AIGC mode overcomes the challenges of the first two.

DOI: 10.54254/2755-2721/45/20241038

2.2. Artificial Intelligence Model

AI models refer to a computer program that is designed to perform tasks that are related to human intelligence, such as recognizing patterns and making predictions. A model is built with a machine learning algorithm and a large dataset. In the training process, the dataset is used by the model to train, validate, and test itself, aiming to improve its performance by updating the internal parameters of the model. The algorithm determines how the model is trained.

Different AI models, trained with different algorithms and datasets, are built to solve different types of jobs. Some models' training relies on algorithms of natural language processing and a large dataset of text to understand human languages, such as Recurrent Neural Networks or Transformer models. The models discussed in this paper are basically trained to comprehend human prompts and synthesize images given with textual data.

2.3. Evaluation Methods of T2IS

Typically, there are two primary standards for evaluating T2IS models: image quality and text-image alignment accuracy. Fréchet Inception Distance (FID) [11] is a common metric used to assess image quality quantitatively. It measures the Fréchet distance between images in the real world and the ones synthesized by models. The image synthesized with high fidelity has a smaller FID. To evaluate the text-image alignment accuracy, inception score (IS) [12], CLIP score [13], or R-precision [14] are widely applied. There are also evaluation benchmarks with human rates involved, such as DrawBench [15], UniBench [16], and PartiPropts [17].

3. Primary Models for Image Synthesis

Currently, most of the T2IS models are built upon five primary models: Transformer [18], CLIP [19], Generative Adversarial networks (GANs) [3], Variational Autoencoders (VAEs) [7], and Diffusion [20]. This section briefly discusses their architecture and some noteworthy methodologies and techniques employed in these models.

3.1. Transformer

Today, transformer architecture [18], which was originally proposed by Vaswani et al. (2017) for NLP tasks, is a crucial component in numerous generative models [21]. It is composed of encoder-decoder stacks, each containing N identical layers, as shown in Figure 2. The transformer model also abandons the use of recurrence and convolution layers.

DOI: 10.54254/2755-2721/45/20241038

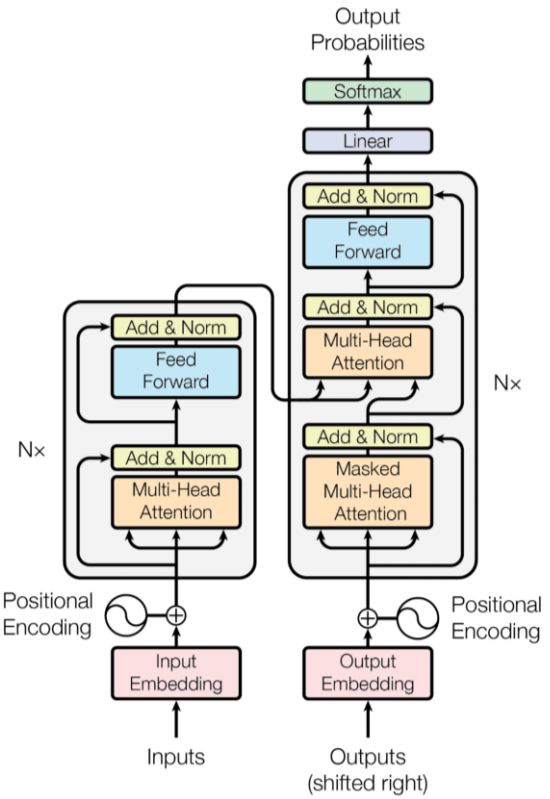

Figure 2. The encoder-decoder structure of the Transformer architecture [18].

3.1.1. Positional Encoding. Instead of using traditional recurrence to capture the relative words' position information, the transformer utilizes positional encoding to inject the positional information into the input embedding. The transformer model first turns each word of the input sequence into a dimensional embedding vector. Then, the positional encoding vectors, which share the same dimensionality as the input embeddings, are generated by using sine and cosine functions with varying frequencies. Finally, the positional encoding vectors are added to the original input embeddings. By doing so, the input embeddings have the positional information to the system.

3.1.2. Encoder-Decoder. The encoder receives the input embeddings, which have undergone positional encoding, and produces hidden representations. These hidden representations are then used by the decoder to generate the output. Both the encoder and decoder layers are built using a multi-head self-attention mechanism, coupled with a fully connected feed-forward network.

3.1.3. Self-attention Mechanism. The self-attention mechanism serves as the central element of transformer models. Its role is to learn how to assign diverse weights to tokens depending on their respective relevance [21]. In the proposed transformer [18], scaled dot-product attention is used. The attention mechanism, shown in Equation 1 below, takes three vectors: Q, K, and V. These vectors are generated by multiplying the input with weight matrices W_q, W_k, and W_v. These weight matrices are updated in the training process for optimization.

Vaswani et al. (2017) [18] found it beneficial to apply multi-head attention, which is the extension of the self-attention mechanism, shown in Equation 2: linear activations are applied h times to the vectors Q, K, and V; each time, different learned linear representations are used. Each of the h projections produces the outputs in the same manner as the self-attention mechanism parallel. These outputs are then concatenated and projected to generate the final result.

DOI: 10.54254/2755-2721/45/20241038

Equation 1:

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$

Equation 2:

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, ..., \text{head}_\text{h})W^O$$
$$\text{where head}_\text{i} = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)$$

3.2. CLIP

3.2.1. Significance. CLIP, which stands for Contrastive Language-Image Pre-training, released by OpenAI in 2021 [19], is the core component of many state-of-the-art T2IS models, such as DALL-E [22] and Stable Diffusion [23]. It is a joint language-vision model that efficiently learns visual concepts from natural language supervision [24]. As a result, given a list of text descriptions and an image, it can predict the most relevant text description with that image. The ability to associate natural language with visual information is the main reason that it is applied widely in many T2IS models.

3.2.2. Methods. The goal is to learn transferable visual models from natural language supervision [19]. The training involves two stages: contrastive pre-training and zero-shot classification. In the first stage, a text encoder and image encoder are trained jointly with a large dataset of 400 million text-image pairs collected from the internet. As a result, the model can predict which one out of 32,768 randomly selected text descriptions is best associated with one given image. Once the pre-training is complete, natural language servers as a means to reference the visual concepts that have been learned, facilitating the model's zero-shot adaptation to subsequent tasks.

3.3. GAN

GANs, first proposed by Goodfellow et al. in 2014 [3], typically have two trained networks: the discriminator and the generator, training and competing against each other at the same time. The discriminator is trained to distinguish the synthetic images from the authentic ones, while the generator is trained to generate more realistic images and try to fool the discriminator. One way to perceive the generator network is as a mapping tool that takes data from a representation space, known as the latent space, and converts it into data space, with our focus primarily being on images; the discriminator network can be considered as a function that maps image data to a probabilistic, reflecting the likelihood of the image originating from the actual data distribution rather than the synthetic data from the generator network.

The generator cannot directly access the real data sample but learns via interaction with the discriminator. The discriminator can access both the synthetic and the authentic data samples. It receives error signals based on understanding whether the image originated from the real collection or was generated. This same error signal, transmitted through the discriminator, can also train the generator, thus improving its ability to create higher-quality images [9].

3.4. VAE

3.4.1. Traditional autoencoder. An autoencoder is a particular kind of neural network engineered primarily for converting input data into a compressed and meaningful representation in the encoder part of the network and then decoding it back into a form that closely resembles the original input as much

DOI: 10.54254/2755-2721/45/20241038

as possible in the decoder part [25]. Traditional autoencoders could be better at generating new image data due to the simplicity of latent space.

3.4.2. Architecture. Variational Autoencoders (VAE), following the Variational Bayes Inference proposed by Kingma et al. in 2013 [7], present a probabilistic approach to defining an observation in the latent space. As a result, instead of creating an encoder that produces a singular value for each attribute of the latent state, the encoder is constructed to define a probability distribution for each latent attribute.

The encoder is supposed to generate the latent variables *z* that follow the optimized probability distribution to help the decoder learn efficient and meaningful latent representations of the input data. The encoder network turns the input samples x into parameters (mean and variance) of a probability distribution for each latent variable. Then, a random sample from a standard normal distribution (mean is 0 and standard deviation 1) is generated. The latent variable *z* for each dimension is generated using the following formula: $z = \mu + \sigma * \varepsilon$, where ε is a noise variable $\sim N(0, 1)$. Finally, the decoder network maps these latent variables back to the input data that resembles the original input [7].

The parameters of the model are trained by two objective functions. The first one is Reconstruction Loss. This loss measures how well the decoder can reconstruct the original input data from the latent representations generated by the encoder. A lower reconstruction loss means the VAE is better at reproducing the original data, suggesting that the latent representations capture the important features of the data. The second one is called KL Divergence. This part of the loss function measures the difference between the learned latent distribution (output of the encoder) and a prior distribution, typically a standard normal distribution. This loss encourages the encoder to generate latent variables that follow the chosen prior distribution [7].

3.5. Diffusion

The main idea of the diffusion model is that it learns to invert a procedure that progressively deteriorates the structure of the training data [20]. The process involves two main phases. The first phase involves several steps where low-level noise is incrementally added to each input image, with the noise's scale differing at every step. This procedure gradually degrades the training data until it transforms into pure Gaussian noise. The second phase reverses this "forward diffusion" process. This phase is carried out in a similar step-by-step manner but in reverse order. It sequentially removes the noise, eventually recreating the original image. During inference, images are produced by gradually reconstructing them from random white noise. A neural network, typically built on a U-Net architecture, estimates the noise subtracted at each step, ensuring the dimensions are preserved throughout the process.

4. Pioneering AI-Based Text-to-Image Synthesis Models

4.1. GAN-Based

4.1.1. Stacked Architecture: StackGAN and StackGAN++. Though GANs proposed by Goodfellow et al. [3] achieved good results, the training procedure is typically unstable and greatly influenced by the selection of hyper-parameters; in addition, it has difficulty in generating high-resolution (e.g., 256 x 256) images [26]. StackGAN, by Zhang et al. [26], is proposed to solve the challenges by decomposing the problem into simpler sub-problems through a sketch-refinement process.

StackGAN contains two-stage generative adversarial networks. In the Stage-I, the GAN outlines the basic shape and colors of the object based on the provided text description, generating low-resolution images. In the second stage, GAN takes results and text descriptions from the Stage-I as inputs and generates high-resolution images with photo-realistic details. As a result, the Stage-II GAN has the capability to correct errors in the Stage-I outcomes and introduce convincing details through the refining procedure.

DOI: 10.54254/2755-2721/45/20241038

Han and his colleagues also introduced a Conditioning Augmentation technique. This approach promotes smoothness in the latent conditioning space, enhancing the variety of the generated images and stabilizing the training of the conditional GAN.

StackGAN++ [27] improves upon the original StackGAN model by incorporating a multi-stage generation process. Rather than just having two stages, StackGAN++ can have multiple stages, each refining the image at an increasing resolution. This allows the model to generate more fine-grained details in the images.

4.1.2. AttnGAN. The early GAN-based T2IS, such as StackGAN and Conditional GAN [28], encodes the textual prompt into a single vector, which was then used as the conditional element; however, this approach has limitations in generating images with complex content because it misses fine-grained word level information for image synthesis. AttnGAN, proposed by Xu et al. [29], solved this problem.

This model employs an innovative attentional generative network that creates intricate details at various image sub-regions by focusing on the relevant words in the natural language description. Additionally, a sophisticated attention-based multimodal similarity model is implemented to compute a detailed image-text matching loss, which assists in the effective training of the generator.

The AttnGAN that's being proposed markedly exceeds prior benchmarks, improving the highest known inception score by 14.14% on the CUB dataset and by an impressive 170.25% on the more demanding COCO dataset.

4.1.3. DF-GAN. Both stacked architecture [26, 27] and AttnGAN [29] made impressive results, but their architecture has drawbacks. The stacked architecture brings entanglements between generators of different image scales. The AttnGAN, due to its architecture, has a weakness in maintaining semantic consistency and making full use of the textual data [30].

DF-GAN [30] is proposed to address the limitations of the existing GAN-based models. It has a simplified backbone that uses only one pair of generators and discriminators to synthesize high-quality images directly, making the training process more efficient. It uses a new regularization technique called Matching-Aware zero-centered Gradient Penalty to promote the generator to synthesize more realistic and text-image semantic consistent images without introducing extra networks, reducing the training complexity and computational cost. The system also employs an innovative component known as the Deep Text-Image Fusion Block, designed to skillfully harness the semantic aspects of text descriptions. This module deeply integrates text and image features during the generation phase, leading to enhanced image quality and greater semantic coherence.

4.1.4. VQGAN+CLIP. CLIP [19], as discussed in the previous section, can make a positive impact as it is combined with generative models, such as GANs. When deployed as a discriminator element within a generative deep learning framework, CLIP can direct the generator component to create digital images that optimally correspond to a provided text prompt.

VQGAN-CLIP [13], which was a popular GAN+CLIP-based T2IS model in 2021. In this model, VQGAN (Vector Quantized Generative Adversarial) [31], an effective and expressive model that combines convolutional neural networks with transformer architectures, serves as a generator producing high-quality images. It interacts with CLIP, which guides the generator to produce images that match text prompts well.

4.2. Transformer-Based

DALL-E, or Craiyon, proposed by Ramesh et al. in 2021 [22], is a zero-shot text-to-image generator that autoregressive models the text and image tokens as a single stream of 1280 tokens, where 256 of them are text tokens, and 1024 of them are image tokens. It generates images from a simple text prompt from the users and outputs only one that best matches the users' requirements.

The training of DALL-E has two stages. In the first stage, a discrete variational autoencoder (dVAE) is trained to compress each 256 x 256 RGB image into a smaller-sized 32 x 32 grid of image tokens. In

DOI: 10.54254/2755-2721/45/20241038

the second stage, a BPE text encoder is utilized to generate 256 text tokens with the 32 x 32 = 1024 image tokens, and then a Transformer is trained to autoregressive models these text and image tokens. In addition, CLIP [19] is utilized for the re-ranking purpose during the inference stage. As a result, the model receives a textual description, predicts the image tokens, and then decodes them into an image during inference. The result follows a zero-shot fashion, which means that it can synthesize images that weren't encountered during the training phase.

Ramesh et al. [22] compared DALL-E with three prior approaches, which are AttnGAN [29], DM-GAN [32], and DF-GAN [30], in terms of Inception Score [12] and Fréchet Inception Distance [11]. The results show that with sufficient data and scale, the proposed approach is competitive with previous domain-specific models. Specifically, the proposed approach achieves state-of-the-art results on the CUB and Oxford-102 datasets and competitive results on the COCO dataset.

Even though DALL-E achieves good results compared to the prior works, it is only good at generating cartoonish images. It falls short in precision while producing images that closely resemble real-life photography.

4.3. Diffusion-Based

Recently, it is widely believed that diffusion-based models outperform GANs-based models, according to the paper [33] released in 2021. In recent years, more and more state-of-the-art diffusion-based T2IS models have been launched by multiple companies and gained popularity.

4.3.1. GLIDE. GLIDE, released by OpenAI in 2021 [34], is classifier-free guidance diffusion-based T2IS model. This model is trained on the same dataset as DALL-E [22], which is a CLIP guidance-based model and generates better-quality images. [34] shows that classifier-free guidance results in an improved trade-off between Precision/Recall and the Inception Score/Fréchet Inception Distance; GLIDE samples, compared to those generated by DALL-E [22], are more photo-realistic and have better conformity with the captions; in addition, the model gets competitive zero-shot FID score on the MS-COCO dataset.

In addition, GLIDE is capable of image inpainting: the user can edit a specified region on an image with a text description.

4.3.2. DALL-E 2. One year later, when DALL-E [22] was released in January 2021, DALL-E 2 [35], also released by OpenAI, came out, generating a more realistic and correct image with greater resolution. According to the survey, more users prefer the image quality produced by DALL-E 2 than DALL-E.

DALL-E 2 [35] leverages the robust representations of images learned from the CLIP text encoder, the contrastive models. A two-stage model is proposed: a prior model that transforms the text embedding into a CLIP embedding and a decoder that generates an image conditioned on the image embedding.

4.3.3. Imagen Imagen [15], the first T2IS model released by Google, consists of a text encoder that converts text into a sequence of embeddings. These embeddings are then processed by a set of conditional diffusion models, which gradually transform them into images of increasing resolution.

Saharia et al. [15] found that augmenting the size of the language model in Imagen greatly enhances both the quality of the samples and the alignment between image and text, far more than enlarging the size of the image diffusion model does. Instead of CLIP embeddings, a pre-trained NLP encoder T5-XXL is used.

Saharia et al. also compare Imagen with other T2IS models, including latent diffusion models [23], Glide [34], DALL-E 2 [35], and VQ-GAN+CLIP [13]. The results show that human raters prefer the quality and image-text alignment of Imagen using the DrawBench evaluation benchmark [15].

4.3.4. Stable Diffusion. Although the image synthesis results of diffusion models are impressive, the optimization process requires hundreds of GPU days, and inference is expansive. The reason is that the image formation process is sequential, and these models operate directly in pixel space. Latent Diffusion

DOI: 10.54254/2755-2721/45/20241038

models [23] are proposed to facilitate training of Diffusion Models on restricted computational resources while maintaining their quality and adaptability.

4.3.5. Latent Diffusion Model. One of the advantages of the Latent Diffusion model [23] is its computational efficiency. Instead of operating in pixel space, the Latent Diffusion model works within a compressed image generation, enabling faster generations and formation of different modalities due to the smaller data size.

Similarly to traditional Diffusion Models, the Latent Diffusion Models also have a forward process and reverse process, as shown in Figure 3. In the forward process, the initial image X is encoded into a latent space, represented as z, with an encoder. Subsequently, Gaussian noise is added T times to this compact image representation z, generating zT in the diffusion process. In the reverse process, zT is fed into a U-Net, where it is predicted back to its original latent space, z. This z is then passed to the Decoder D. The Decoder transforms the latent space image z to pixel space image.

Latent Diffusion Models are also an arbitrary conditioning mechanism based on a cross-attention mechanism, which is beneficial for learning attention-based models across diverse input modalities. As shown in Figure 3, the conditioning inputs, such as text or images, are transformed with τθ and mapped to U-Net layers with the cross-attention layer.

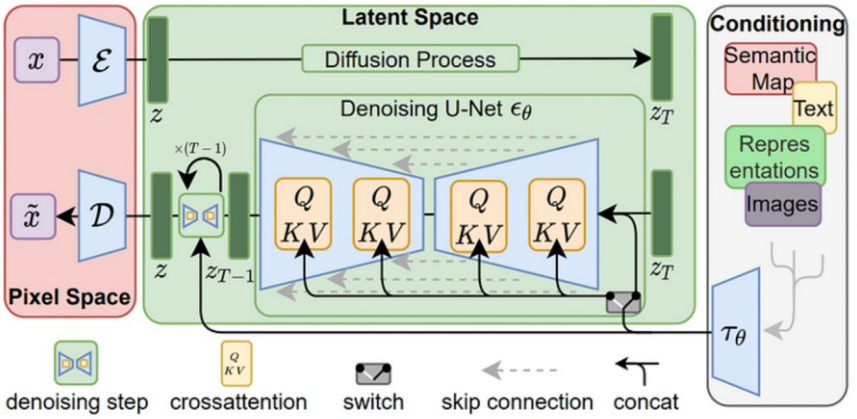

Figure 3. The structure of Latent Diffusion Models [30].

4.3.6. DreamStudio. Developed by Stability AI, DreamStudio employs Stable Diffusion [23], a latent T2I diffusion model. This model is conditioned on the (non-pooled) text embeddings generated by a CLIP ViT-L/14 text encoder, and is used to create images from provided phrases or sentences. Compared with DALL-E 2, It offers competitive performance and fast processing speeds [21].

5. Challenges and Future Directions

5.1. Factuality

Even though today's T2IS model is able to generate realistic content, it is still not guaranteed to produce reliable content in terms of factuality. Even powerful image synthesis models, like stable diffusion [23], cannot draw human hands correctly [36]. They have a propensity to add excessive digits or blend fingers together, resulting in an unsettling, nightmarish appearance. One reason for this might be that the human hands are so small in the source images that models are hard to learn. In addition, 2D image generators struggle to comprehend the hands in 3D form [36] fully.

5.2. Security and privacy risks

Three potential security and privacy attacks on T2IS models are discussed: backdoor attack [37], membership leakage [38], and data extraction. Backdoor attack refers to pre-trained text encoders of

DOI: 10.54254/2755-2721/45/20241038

T2IS models being injected, leading to an image, which may contain biased or sensitive information, being forced to be generated if a keyword exists in the text prompt. With a membership inference attack [38], an image that is used to train the models can be inferred. Three kinds of intuitions, including quality, faithfulness, and reconstruction error, are proposed to design the attack algorithm. The goal of a data extraction attack is to get a sample from a training data set. Some diffusion models have a data replication problem: the models copy content directly from the training set [39]. Carlini et al. [40] proposed a privacy attack method where a generate-and-filter pipeline is utilized to extract over a thousand samples from the models' training set. Carlini et al. [40] reported that state-of-the-art diffusion models, such as stable diffusion [23] and Imagen [15], are more prone to violations of privacy.

5.3. Hardware

Hardware plays an important role in the training of large-scale models, such as T2IS models. Powerful hardware brings several advantages, such as speed and efficiency, parallel processing, and scalability. In today's world, a huge advance has been made in hardware. For instance, the Tensor Processing Units (TPUs) from Google, which are made specifically for large-scale deep learning tasks, accelerate machine learning workloads significantly [41].

To train large-scale models well, well-distributed training frameworks are required. Instead of training on one single processor, multiple processors are used to reduce the stress of huge workloads. Recently, there have been good frameworks, such as PyTorch [42], DeepSpeed [43], and TensorFlow [44], allowing deep learning developers to easily manage huge workloads without knowing the details of the underlying infrastructure.

Cloud computing is another promising field in reducing hardware stress. Instead of training models locally, developers can access powerful computing resources through cloud computing services, such as AWS and Azure. Developers can also access powerful TPUs through Cloud TPU, a web service that provides TPUs as scalable computing resources on Google Cloud.

6. Conclusion

In this paper, we explored various text-to-image synthesis (T2IS) models and their advancements in the field of artificial intelligence. We discussed and analyzed five primary models, including Transformer [18], CLIP [19], Generative Adversarial networks (GANs) [3], Variational Autoencoders (VAEs) [7], and Diffusion [20], along with the pioneering AI-Based T2IS models build upon those primary models. The current T2IS models have demonstrated remarkable capabilities in bridging the gap between natural language and visual data.

Although T2IS has achieved promising results thanks to the invention of the deep-learning algorithm and architectures we discussed, the journey of T2IS is not without its challenges. GAN-based models have shown impressive results in producing high-quality images and photo-realistic images, but they often face challenges in training stability and semantic consistency. Transformer-based models, like DALL-E [22], have revolutionized T2IS by combining text and image tokens into a single stream and creating images based on textual prompts; however, they may struggle to create images that closely resemble real-life photography. Diffusion models, such as GLIDE [34], Imagen [15], and Stable Diffusion [23], offer advantages in terms of computational efficiency and image quality, but they require substantial computational resources for training and inference. Besides, from the issues of models' security and privacy risks to increasing hardware demand for large-scale model training, there still remain many obstacles to overcome. Addressing these issues is critical for advancing and integrating T2IS into our everyday digital experiences. As technology continues to evolve, we eagerly await the next generation of models that will push the boundaries of what is currently possible.

References

[1] Yunjiu L, Wei W, and Zheng Y 2022 Artificial intelligence-generated and human expert-designed vocabulary tests: a comparative study SAGE Open 12 1

[2] "Chatgpt: Optimizing language models for dialogue," Nov. 2022.

DOI: 10.54254/2755-2721/45/20241038

[3] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," 2014.
[4] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," Nature, vol. 521, no. 7553, pp. 436–444, 2015.
[5] A. A. Efros and T. K. Leung, "Texture synthesis by non-parametric sampling," in Proceedings of the seventh IEEE international conference on computer vision, vol. 2, pp. 1033–1038, IEEE, 1999.
[6] P. S. Heckbert, "Survey of texture mapping," IEEE computer graphics and applications, vol. 6, no. 11, pp. 56–67, 1986.
[7] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," arXiv preprint arXiv:1312.6114, 2013.
[8] Y. Song and S. Ermon, "Generative modeling by estimating gradients of the data distribution," Advances in Neural Information Processing Systems, vol. 32, 2019.
[9] A. Creswell, T. White, V. Dumoulin, K. Arulkumaran, B. Sengupta and A. A. Bharath, "Generative Adversarial Networks: An Overview," in IEEE Signal Processing Magazine, vol. 35, no. 1, pp. 53-65, Jan. 2018, doi: 10.1109/MSP.2017.2765202.
[10] AI-Generated Content (AIGC): A Survey
[11] Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., and Hochreiter, S. Gans trained by a two time-scale update rule converge to a local nash equilibrium. arXiv preprint arXiv:1706.08500, 2017.
[12] Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., and Chen, X. Improved techniques for training gans. arXiv preprint arXiv:1606.03498, 2016.
[13] J. Hessel, A. Holtzman, M. Forbes, R. L. Bras, and Y. Choi, "Clipscore: A reference-free evaluation metric for image captioning," arXiv preprint arXiv:2104.08718, 2021.
[14] T. Xu, P. Zhang, Q. Huang, H. Zhang, Z. Gan, X. Huang, and X. He, "Attngan: Fine-grained text to image generation with attentional generative adversarial networks," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 1316–1324.
[15] C. Saharia, W. Chan, S. Saxena, L. Li, J. Whang, E. Denton, S. K. S. Ghasemipour, B. K. Ayan, S. S. Mahdavi, R. G. Lopes, et al., "Photorealistic text-to-image diffusion models with deep language understanding," arXiv preprint arXiv:2205.11487, 2022.
[16] W. Li, X. Xu, X. Xiao, J. Liu, H. Yang, G. Li, Z. Wang, Z. Feng, Q. She, Y. Lyu et al., "Upainting: Unified text-to-image diffusion generation with cross-modal guidance," arXiv preprint arXiv:2210.16031, 2022.
[17] J. Yu, Y. Xu, J. Y. Koh, T. Luong, G. Baid, Z. Wang, V. Vasudevan, A. Ku, Y. Yang, B. K. Ayan et al., "Scaling autoregressive models for content-rich text-to-image generation," arXiv preprint arXiv:2206.10789, 2022.
[18] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, "Attention is all you need," in Advances in Neural Information Processing Systems (I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, eds.), vol. 30, Curran Associates, Inc., 2017.
[19] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning Transferable Visual Models From Natural Language Supervision. In Proceedings of the 38th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 139), Marina Meila and Tong Zhang (Eds.). PMLR, 8748–8763.
[20] F. -A. Croitoru, V. Hondru, R. T. Ionescu and M. Shah, "Diffusion Models in Vision: A Survey," in IEEE Transactions on Pattern Analysis and Machine Intelligence, doi: 10.1109/TPAMI.2023.3261988.
[21] L. Yang et al., Diffusion models: A comprehensive survey of methods and applications, arXiv preprint arXiv:2209.00796 (2022).

DOI: 10.54254/2755-2721/45/20241038

[22] A. Ramesh, M. Pavlov, G. Goh, S. Gray, C. Voss, A. Radford, M. Chen, and I. Sutskever, "Zero-shot text-to-image generation," in International Conference on Machine Learning, pp. 8821–8831, PMLR, 2021.
[23] R. Rombach et al., High-resolution image synthesis with latent diffusion models, Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2022) 10684–10695.
[24] "CLIP: Connecting Text and Images." OpenAI, 5 Jan. 2021, openai.com/research/clip.
[25] Bank, Dor, Noam Koenigstein, and Raja Giryes. "Autoencoders." arXiv preprint arXiv:2003.05991 (2020).
[26] Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitris N Metaxas. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In Proceedings of the IEEE international conference on computer vision, pages 5907–5915, 2017.
[27] H. Zhang, T. Xu, H. Li, S. Zhang, X. Wang, X. Huang, and D. Metaxas, "Stackgan++: Realistic image synthesis with stacked generative adversarial networks," IEEE Trans. Pattern Anal. Mach. Intell., vol. 41, no. 8, pp. 1947–1962, 2018.
[28] M. Mirza and S. Osindero, "Conditional generative adversarial nets," arXiv preprint arXiv:1411.1784, 2014.
[29] T. Xu, P. Zhang, Q. Huang, H. Zhang, Z. Gan, X. Huang, and X. He, "Attngan: Fine-grained text to image generation with attentional generative adversarial networks," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2018, pp. 1316–1324.
[30] M. Tao, H. Tang, F. Wu, X. Jing, B. -K. Bao and C. Xu, "DF-GAN: A Simple and Effective Baseline for Text-to-Image Synthesis," 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 2022, pp. 16494-16504, doi: 10.1109/CVPR52688.2022.01602.
[31] Taming Transformers for High-Resolution Image Synthesis
[32] Zhu, M., Pan, P., Chen, W., and Yang, Y. Dm-gan: Dynamic memory generative adversarial networks for text to-image synthesis. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5802–5810, 2019.
[33] P. Dhariwal, A. Nichol, Diffusion models beat GANs on image synthesis, Advances in Neural Information Processing Systems 34 (2021) 8780-8794.
[34] A. Nichol et al., GLIDE: Towards photorealistic image generation and editing with text-guided diffusion models, arXiv preprint arXiv:2112.10741 (2021).
[35] A. Ramesh et al., Hierarchical text-conditional image generation with CLIP latents, arXiv preprint arXiv:2204.06125 (2022).
[36] "Why AI Image Generators Can't Get Hands Right," March 02, 2023.
[37] L. Struppek, D. Hintersdorf, and K. Kersting, "Rickrolling the artist: Injecting invisible backdoors into text-guided image generation models," arXiv preprint arXiv:2211.02408, 2022.
[38] Y. Wu, N. Yu, Z. Li, M. Backes, and Y. Zhang, "Membership inference attacks against text-to-image generation models," arXiv preprint arXiv:2210.00968, 2022.
[39] G. Somepalli, V. Singla, M. Goldblum, J. Geiping, and T. Goldstein, "Diffusion art or digital forgery? investigating data replication in diffusion models," arXiv preprint arXiv:2212.03860, 2021.
[40] N. Carlini, J. Hayes, M. Nasr, M. Jagielski, V. Sehwag, F. Tramèr, B. Balle, D. Ippolito, and E. Wallace, "Extracting training data from diffusion models," arXiv preprint arXiv:2301.13188, 2023.
[41] "Introduction to Cloud TPU." Google Cloud, 14 June. 2023, cloud.google.com/tpu/docs/intro-to-tpu. Accessed 15 June. 2023.
[42] Paszke, Adam, et al. "Pytorch: An imperative style, high-performance deep learning library." Advances in neural information processing systems 32 (2019).
[43] J. Rasley, S. Rajbhandari, O. Ruwase, and Y. He, "Deepspeed: System optimizations enable training deep learning models with over 100 billion parameters," in Proceedings of the 26th

DOI: 10.54254/2755-2721/45/20241038

ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 3505–3506, 2020.

[44] Abadi, Martín. “TensorFlow: learning functions at scale.” Proceedings of the 21st ACM SIGPLAN International Conference on Functional Programming. 2016.