

RenAlssance: A Survey into AI Text-to-Image Generation in the Era of Large Model

Fengxiang Bie, Yibo Yang, Zhongzhu Zhou, Adam Ghanem, Minjia Zhang, Zhewei Yao, Xiaoxia Wu, Connor Holmes, Pareesa Golnari, David A. Clifton, Yuxiong He, Dacheng Tao, Shuaiwen Leon Song

Abstract—Text-to-image generation (TTI) refers to the usage of models that could process text input and generate high fidelity images based on text descriptions. Text-to-image generation using neural networks could be traced back to the emergence of Generative Adversarial Network (GAN), followed by the autoregressive Transformer. Diffusion models are one prominent type of generative model used for the generation of images through the systematic introduction of noises with repeating steps. As an effect of the impressive results of diffusion models on image synthesis, it has been cemented as the major image decoder used by text-to-image models and brought text-to-image generation to the forefront of machine-learning (ML) research. In the era of large models, scaling up model size and the integration with large language models have further improved the performance of TTI models, resulting the generation result nearly indistinguishable from real-world images, revolutionizing the way we retrieve images. Our explorative study has incentivised us to think that there are further ways of scaling text-to-image models with the combination of innovative model architectures and prediction enhancement techniques. We have divided the work of this survey into five main sections wherein we detail the frameworks of major literature in order to delve into the different types of text-to-image generation methods. Following this we provide a detailed comparison and critique of these methods and offer possible pathways of improvement for future work. In the future work, we argue that TTI development could yield impressive productivity improvements for creation, particularly in the context of the AIGC era, and could be extended to more complex tasks such as video generation and 3D generation.

Index Terms—Text-to-Image, Diffusion Model, DPM, CLIP, Language Model, Vision Transformer, VAE, GAN, Large model

1 INTRODUCTION

IN recent years, AI Generated Content (AIGC) has emerged as a powerful tool for content creation, which has the potential to revolutionize the way we retrieve and consume information. AIGC refers to the use of artificial intelligence to create written, visual, or audio contents for creative, educational, and a plethora of other application cases. With advancements in the field of natural language processing (NLP) and computer vision, AIGC is capable of producing high-quality content that is progressively becoming indistinguishable from the content generated by human writers, designers, and artists. For instance, language generation models like ChatGPT [1, 2] demonstrate a remarkable ability to understand complex language and respond with expertly crafted text, closely aligned with the input query. At the same time, image generation models [3–14] have made remarkable strides in multiple tasks, rendering AI-generated images that rival those created by human designers or real-world photographs.

Building upon this foundation, it becomes evident that applying NLP and computer vision models to AIGC holds great significance. Researchers have explored the synergistic potential of these models, seeking to leverage knowledge from both visual and textual domains, which generate visually coherent and contextually relevant images based on

textual descriptions. With the combination of generative models [15–19] and large language models [20–23], it is now possible to automatically generate high-fidelity images that closely match complex text descriptions. This advancement opens up new possibilities for creative expression, design, and multimedia content creation. The era of large models has facilitated the scaling of model size and training datasets, resulting in further enhanced performance for text-to-image (TTI) generation models. Consequently, the gap between generated images and those produced by top artists and photographers is diminishing. TTI generation has made a profound impact across a wide range of fields and industries, boosting productivity and saving substantial time in art design. As a result, TTI generation has become one of the most hotly debated and researched topics within the realm of AIGC. Nowadays, any individuals or industrial professionals can easily retrieve AI generated images through on-device models or online platforms, such as Stable Diffusion [24], PARTI [5], Dreambooth [25], NovelAi [26], and Midjourney [27].

The field of TTI models has witnessed significant development, evolving from simple image generation capabilities [28, 29] to the impressive capacity to create complex and realistic images [3, 30, 31]. Researchers have explored various deep learning [32] architectures, in order to improve image quality and sampling efficiency. During the early stages of TTI generation, generative adversarial network (GAN) [16] served as the primary backbone generative model [29, 33–38], which generates images efficiently with a generator-discriminator structure [16]. Another novel approach for TTI generation is the Autoregressive model [5, 31, 39], which

- F. Bie, Z. Zhou, A. Gahem, D. Tao, S L. Song are with the University of Sydney, Australia.
- Y. Yang is with King Abdullah University of Science and Technology.
- M. Zhang, Z. Yao, X. Wu, C. Holmes, P. Golnari, Y. He and S L. Song are with Microsoft, US.
- D.A. Clifton is with the University of Oxford, UK.

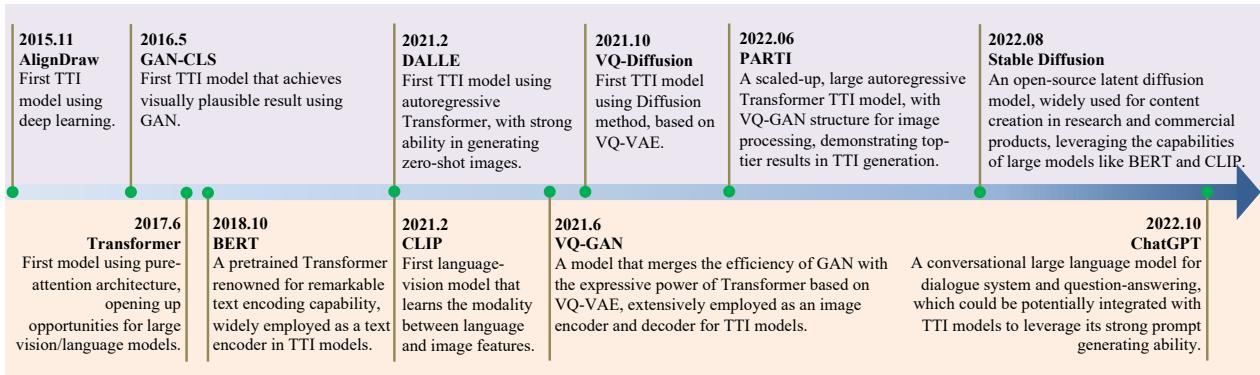


Fig. 1. The milestones of text-to-image (TTI) models and large models. The upper part in light purple shows the key TTI models with high impacts, and the lower part in light yellow shows the progress of large models that provide the stimuli for the development of TTI models.

employs the Transformer [20] architecture to predict image tokens sequentially based on text tokens. More recently, there is a clear trend that Diffusion model has gained popularity as a prominent method [3, 4, 30, 40]. This approach denoises images by modeling them from Gaussian distributions [41] using a Markovian-chain process [17, 42]. The ongoing innovations in architecture improvements and augmentation techniques have substantially enhanced Diffusion TTI models, enabling them to generate higher fidelity and more diverse images.

Each major type of TTI model has its strengths and weaknesses. GAN models are efficient in scale and are able to generate images in a negligible amount of time. However, they may suffer from fidelity loss and lack intricate image details compared to autoregressive Transformer and Diffusion models [42]. Autoregressive Transformer has the ability to generate images highly realistic and comparable to the performance of the Diffusion method. The drawback is that they usually consist of more than 5B [5, 39, 43] parameters, which is much larger in model size and makes it hard to deploy on a wide range of local devices. Autoregressive Transformer predicts image tokens in a non-parallel way, which impacts the efficiency of the inference process. Despite the fact that diffusion TTI models have a reasonable model size and also achieve top results in TTI generation, it requires 50-200 sampling steps which significantly prolongs total time consumption during inference. While early GAN models may have lagged behind state-of-the-art autoregressive and diffusion models, recent advancements, combined with the benefits of large language models, have significantly narrowed the gap [44-46]. This progress has fostered improved performance in GAN models, making them more competitive in comparison to new autoregressive and diffusion models. Given the unique advantages of each model architecture, such as model size, time efficiency, and their demonstrated capability to generate high-fidelity and diverse images, researchers continue to explore and push the boundaries of all three types of TTI models. The accomplishments of various large language models have also significantly catalyzed the advancement of TTI models. These models serve as the bedrock for robust text encoders and exemplify the evolution of more sophisticated model architectures, as illustrated in Fig 1.

As all types of TTI models have experienced considerable improvements, it becomes evident that no single model type holds an absolute advantage over others. Therefore, a comprehensive survey that encompasses recent innovations and advancements in all TTI models becomes necessary to facilitate a clear understanding of their development and enable researchers to conveniently compare different models. To our best knowledge, previous surveys have focused on text-to-image generation only on a certain type of generative models, such as surveys only for GAN TTI generation [47-50], and surveys focusing on diffusion models [51, 52], we are the first survey that gives a holistic overview connecting all types of TTI models, and consider their development under the influence of large models. By addressing this knowledge gap and aligning our content with the current trends in deep learning, our survey serves as a crucial resource in the landscape of TTI research. In summary, we recognize the key contributions of our survey as follows:

- 1) We give a detailed introduction to the vital components of TTI models, including different generative models, language models, and vision models.
- 2) Our survey encompasses multiple types of text-to-image generative models, not only those developed in the early stages, but also modern models that have been influenced by the advancements of large models. This inclusive perspective allows us to capture the full spectrum of TTI model development.
- 3) We compare the performance of different types of TTI models with both visual results and statistical results. We further discuss pros and cons of each model based on the comparison results.
- 4) We outlook the potential development of TTI models, foreseeing their applications to industries and different research aspects.

2 BACKGROUND

2.1 AI Image generation

AI image generation involves the use of neural networks to generate new images that resemble a set of input images or even have the ability to manufacture creative pieces of work. Image generation is always a heated topic under the computer vision area. Popularization of the convolutional neural

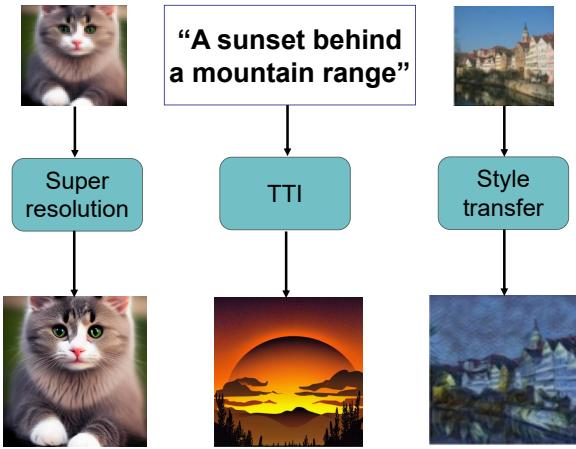


Fig. 2. Examples of common tasks for image generation, including super-resolution (left), text-to-image generation (middle) and style change (right). Super-resolution enhances the image by adding more details and generates the output with high fidelity. Style transfer transfers the image to another domain with mutual information. Text-to-image generates high quality images that are well aligned with user input.

network [53, 54] and Transformer models [20] has pushed the success in many works such as image-segmentation [55–58], super-resolution [59–62], style change [11–14], object detection [63–65] and text-to-image generation. Fig 2 shows examples for popular image generation tasks. Super-resolution model generates images from low resolution to a higher resolution space. Unlike traditional methods such as bi-linear interpolation that results in a blurred image, super-resolution, through the use of an additional neural network, adds new details to the image and results in a high fidelity for the output image. Style transfer [7] using deep learning could transfer input images to another domain but keeps most of the semantic meaning. Text-to-image generation takes a text sentence and generates a variety of images that are all well-aligned with the text description. Text-to-image generation has close connections to daily lives and has the potential to be applied to a wide range of research and industries. Therefore, it has become one of the most discussed topics in image generation. However, text-to-image generation is hard to implement, given that it involves the complex connection of human language and visual representation, which are two separate fields in computer science research. Multiple tools and models are required to support each process of TTI generation pipeline. For example, language models are necessary for processing text inputs as conditional information to guide image generation. Vision models may also be required for encoding image data. Generative models also work as essential components used for image generation, such as GAN and U-net. VAE models could be used to map image information to a latent space.

2.2 Generative models

2.2.1 VAE: Variational Autoencoders

Variational autoencoders (VAE) [15, 66] are a class of generative models comprising of an encoder-decoder [67] structure that is able to reconstruct input data. The encoder maps each

data element x_i to a latent space z that follows a Gaussian distribution through the generation of a mean and variance within the latent space:

$$\log q_\phi(z^{(i)}|x^{(i)}) = \log \mathcal{N}(z^{(i)}; \mu^{(i)}, \sigma^{2(i)} I). \quad (1)$$

The variance of the Gaussian distribution is critical in providing the generative quality as noises are incrementally introduced to the latent space. The encoder learns to predict the variance and mean of the distribution, however, this process here is not differentiable as we are sampling from a distribution we are parameterizing. Thus, VAE makes use of a re-parameterization trick which separates the process of sampling and the prediction of the mean and variance:

$$z_i = \mu_i + \epsilon \times \sigma_i, \quad (2)$$

where ϵ is the noise randomly sampled from the normal Gaussian distribution. The decoder's general aim is to generate the output data to be as close as the given input data. This process is done by maximising the probability of generating input data given the latent space variable $P(x^i|z)$. VAE training is based on maximizing the tractable lower bound L_b of $P(X)$:

$$L_b = -D_{KL}(q(z|x)p(z)) + \frac{1}{L} \sum_{i=1}^L (\log P(x^i|z)). \quad (3)$$

Since the encoder aims to regulate the latent space in the normal Gaussian distribution $P(Z|X) = \mathcal{N}(0, I)$, the loss function could be rewritten as:

$$\begin{aligned} \mathcal{L}(\theta, x^i) = & -\frac{1}{2} \sum_{i=1}^d (\mu_{(i)}^2 + \sigma_{(i)}^2 - \log \sigma_{(i)}^2 - 1) \\ & + \frac{1}{L} \sum_{i=1}^L (\log P(x^i|z)). \end{aligned} \quad (4)$$

Compared to autoencoders, VAE applies a probabilistic model to deep neural networks. VAE's main purpose is to generate a data distribution rather than giving an output with fixed numbers as done by autoencoders. Thus, sampling from the distribution increases the data diversity as well as the capability of noise-handling.

There are some limitations for VAE models. For image generation, directly sampling from a Gaussian distribution always results in a blurred image. Besides this, information loss occurs when projecting the data to a lower dimension. Therefore, VAE has relatively less performance for image synthesis compared to the state-of-the-art generative models such as GAN and Diffusion models. However, the latent space structure works as a useful feature presentation in text-to-image tasks, and it can be used for significant computation reduction combined with diffusion models [3, 68].

2.2.2 GAN: Generative adversarial network

GAN [16] is a generative model that samples the image without the computation of the density function from the training data. GAN consists of a *generator* (G) and a *discriminator* (D). The discriminator maximises the accuracy to distinguish the input images from the synthesised images

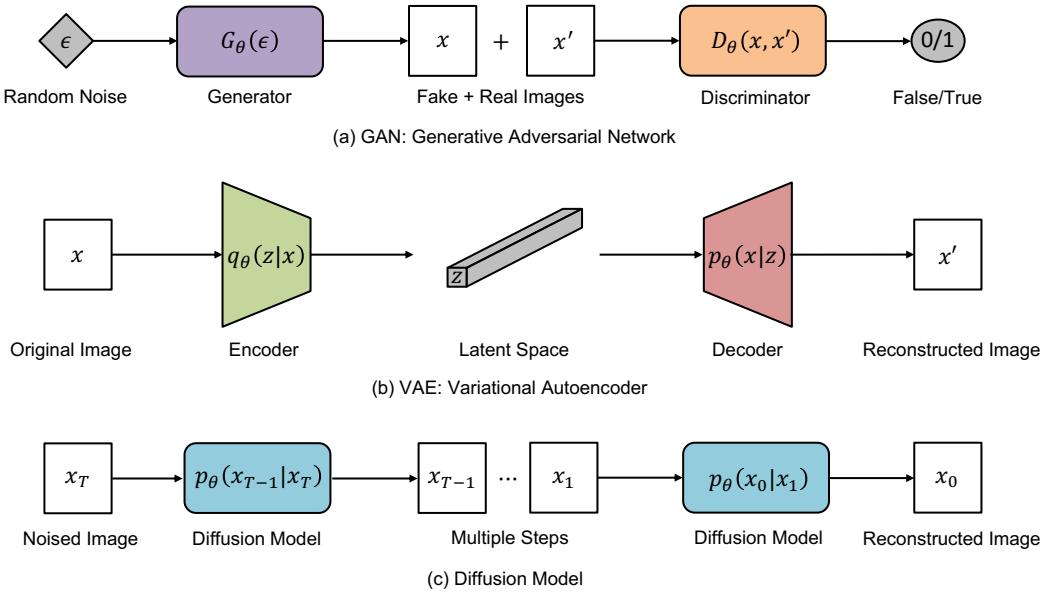


Fig. 3. A sketch comparison of generative model architectures. (a) GAN model generates image using the generator that is trained by adversarial training. (b) VAE model reconstructs image by maximizing variational lower bound from the latent representation. (c) Diffusion model reconstructs image by denoising the Gaussian-noised image in an iterative process.

with a binary output. The generator learns the image distribution from the training samples and tries to minimize the possibility to be classified as fake by the discriminator.

GAN training process is shown in Fig 3, which is based on the adversarial process with the minimax algorithm. This is achieved by either fixing the generator and taking gradient ascent on the discriminator:

$$\nabla_{\theta_d} \frac{1}{n} \sum_{i=1}^n [\log D_{\theta_d}(x^i) + \log(1 - D_{\theta_d}(G_{\theta_g}(z^i)))] , \quad (5)$$

or fixing the discriminator and taking gradient descent on the generator:

$$\nabla_{\theta_g} \frac{1}{n} \sum_{i=1}^n \log(1 - D_{\theta_d}(G_{\theta_g}(z^i))), \quad (6)$$

for a minibatch of n samples of image data x_1, x_2, \dots, x_n and noise data z_1, z_2, \dots, z_n .

Compared to VAE, GAN generates images with better fidelity. However, GAN could not extract an explicit representation of the data density $P(x)$, and training a GAN model is trickier considering the slower converging trend. Moreover, there is also a need to balance the performance of discriminator and generator to avoid mode collapse that ends up generating the same sample from noises [69]. GAN model has achieved impressive results for image generation [7, 34, 36], whose success also drives the development of text-to-image generation. Besides this, GAN model could also be applied to other complicated generation and perception tasks such as image super-resolution [60, 70–72], and object detection [73, 74].

2.2.3 Diffusion models

Diffusion model [17, 42] is another type of generative model that could generate images with repeating steps, in a form

of latent variables to model the probability distribution [75]: $P(x_0) = \int dx(1\dots T)p(x(0\dots T))$, illustrated in Fig 3. Generally, a diffusion model involves adding noise to perturb the data during training and removing the noises with learned parameters during inference. Algorithm 1 and 2 summarize the general steps to train a diffusion model and apply a diffusion model to sample data, respectively. Using diffusion models for TTI not only ensures a high fidelity of the output image but also provides varieties of the image styles. Therefore, it has become one of the top choices for TTI generation. Apart from TTI generation, diffusion model also achieves impressive results in image segmentation [55, 61, 76–78], super-resolution [79, 80], natural language processing [81–84], or even in the field of reinforcement learning [85, 86]. Recent researches have focused on improving diffusion model structures to increase the quality of the synthesised image or decrease the total time for the inference. There are two types of popular diffusion models, denoising diffusion probabilistic models (DDPM) and score-based diffusion models.

DDPM [17] is a fundamental type of diffusion model which uses discrete timesteps. DDPM consists of a forward and a reverse process, both of which follow the Markov chain process. The forward process gradually adds noise to the original image until the data becomes a normal Gaussian noise. The noise added in each timestep is sampled from the normal Gaussian distribution and is scaled by a scheduler β_t and $\alpha_t = 1 - \beta_t$. This is normally done with an arbitrary T timesteps:

$$q(x_{1:T}) = \prod_{t=1}^T q(x_t|x_{t-1}), \quad q_\theta(x_t|x_{t-1}) = N(x_t, \sqrt{\alpha_t}, x_{t-1}, \beta_t I). \quad (7)$$

Because of the addition property of the Gaussian distribu-

Algorithm 1 Diffusion model training

```

1: for every training iteration do
2:   Sample  $t$  from discrete timestep [17] [42], i.e.,
    $1, 2, \dots, T$ , or from continuous timestep [87] [88] i.e.,
    $t \sim [0, 1]$ .
3:   Sample random noise from  $\epsilon \sim \mathcal{N}(0, 1)$ .
4:   Calculate  $x_t$  based on DDPM forward Eq. (8) or SDE
   forward Eq. (12).
5:   Update the model with noise prediction  $\epsilon(x_t, t)$  or
   score function  $s(x_t, t)$ .
6: end for

```

tion, Eq. (7) could be converted to:

$$q_\theta(x_t|x_0) = \mathcal{N}(x_t, \sqrt{\bar{\alpha}_t}x_0, (1 - \bar{\alpha}_t)I), \quad (8)$$

where $\bar{\alpha}_t = \prod_{s=1}^t \alpha_s$. This ensures that we can directly calculate data at any timesteps with a noise sampled from the normal Gaussian distribution: $x_t(x_0, \epsilon) = \sqrt{\bar{\alpha}_t}x_0 + \epsilon \times \sqrt{(1 - \bar{\alpha}_t)}$. The scheduler design of α ensures that $\bar{\alpha}_n = 0$, which indicates that the data will be perturbed to a normal Gaussian noise $\mathcal{N}(0, I)$ at the last timestep n .

The reverse process is also a Markovian chain that samples the data from a learned Gaussian distribution at each timestep to remove the noise:

$$p_\theta(x_{t-1}|x_t) = \mathcal{N}(x_t - 1, \mu_\theta(x_t, t), \Sigma_\theta(x_t, t)). \quad (9)$$

DDPM takes the variance as a fixed schedule guided by β : $\Sigma_\theta(x_t, t) = \beta_t$ or $\Sigma_\theta(x_t, t) = \frac{1 - \alpha_t}{1 - \alpha_t - 1} \beta_t$ and calculates the mean value with the approximation of x_0 given x_t :

$$\mu_\theta(x_t, t) = \bar{\mu}(x_t, \bar{x}_0) = \frac{1}{\sqrt{\alpha_t}}(x_t - \frac{\beta}{\sqrt{1 - \bar{\alpha}_t}}\epsilon_t(x_t, t)), \quad (10)$$

where Eq. (10) is derived from the Bayesian theorem with $\epsilon_t(x_t, t)$ predicting the Gaussian noise added in the forward process at the timestep t . Original image x_0 could be approximated by $x_0 = \frac{1}{\bar{\alpha}_t}(x_t - \beta_t\epsilon_t)$ at any timestep.

Training is based on the optimisation of the variational upper bound, which could be reparameterized based on the form of Eq. (10):

$$L = \left\| \epsilon - \epsilon_\theta(\sqrt{\bar{\alpha}_t}x_0 + \epsilon \times \sqrt{(1 - \bar{\alpha}_t)}, t) \right\|^2. \quad (11)$$

The loss function's purpose is to calculate the MSE error between the predicted noise at a certain timestep and the noise that is sampled from the forward process.

DDPM uses a time-aware U-net [19] as the backbone network that takes the noised data x_t and the timestep t as the input. The output is the predicted noise in the same dimension as the input, corresponding to Eq. (10) and (11). This backbone architecture is adopted by a number of upcoming diffusion models [87–89] and diffusion-based TTI generation models [3, 4, 30].

DDPM shows a much more stable training process in comparison to GAN, and keeps high fidelity of the image at the same time. However, sampling noise at the first stage of the diffusion process often performs poorly on approximating x_0 , therefore, it usually needs massive repeating steps to adjust the sampled noise and thus denoise the data towards the correct direction.

Algorithm 2 Diffusion model Inference

```

1: Sample  $x_t$  from normal gaussian distribution  $x_t \sim \mathcal{N}(0, I)$ .
2: Sample discrete timesteps from  $1, 2, \dots, T$  or continuous timestep from  $[0, 1]$ .
3: for  $t$  in Reverse(timesteps) do
4:   Calculate the noise distribution [17] [42] [87]  $\epsilon(x_t, t)$  or score function [88]  $s(x_t, t)$  with the corresponding diffusion model.
5:   Approximate  $x_{t-1}$  or  $x_{t-\Delta t}$  based on the reverse function.
6: end for

```

Score-based diffusion models [88] generalize the DDPM process using differential equations. Along a similar vein, score matching [90] refers to the use of a score function to estimate the gradient of the dense probability function $S(x_t, t) = \nabla_x \log P(x_t)$. In diffusion models, this could be applied using stochastic differential equations (SDE) [88]:

$$dx = f_t(x)dt + g_tdw, \quad (12)$$

for the forward process. There are a few ways to estimate the score through the training of a score-based model, such as denoising score matching [90, 91], sliced score-matching [92], and finite-difference score matching [93]. SDE diffusion model [88] takes the denoising score matching approach via continuous time generalization of the DDPM loss function to train the score model:

$$\theta^* = \operatorname{argmin}_\theta \mathcal{E}_t \lambda(t) \mathcal{E}_{x(0)} \mathcal{E}_{x(t)|x(0)} [\|s_\theta(x(t), t) - \nabla_{x(t)} \log P(x_t|x_0)\|^2]. \quad (13)$$

The reverse process also matches a diffusion process [94], yielding the result of:

$$dx = [f_t(x) - g_t^2 \nabla_x \log p_t(x)] dt + g_t dw. \quad (14)$$

Such reverse process could alternatively be represented by an ordinary differential equation (ODE):

$$dx = \left(f(x, t) - \frac{1}{2}g_t^2 \nabla_x \log p_t(x) \right) dt, \quad (15)$$

with all of them based on the trained neural network predicting dense probability function to denoise the image towards the correct direction.

2.3 Large Language Model

Language models are advanced natural language processing (NLP) systems that are capable of processing and generating human-like languages. Through the use of deep learning, it has allowed for language models to learn directly from raw text data, without the need for extensive feature engineering. The breakthrough of language model comes with the introduction of the Transformer [20] with an attention mechanism that remembers and builds the relationship based on a complete text sequence. This is instrumental for the emergence of large language models that are trained on massive amounts of data to capture the nuances and complexities of human language. With architectural improvements and significant scaling in model

TABLE 1

A comparison of the popular large language models and vision models across literature. As you can see there is a trend for much large parameters, however, as the computational cost grows there is limited compute power to meet such a large demand.

model	Size	Model Type
CLIP [95]	151M	Language-vision contrastive model
BERT [22]	340M	Encoder-only based Transformer model architecture
Megatron-LM[96]	8.3B	Encoder-decoder based large Transformer model
T5 [21]	11B	Encoder-decoder based Transformer model architecture
Flan-T5 [97]	11B	Improved T5 model
OPT [98]	125M-175B	Decoder-only pre-trained transformers with varied model parameters
LLama-1,2 [99, 100]	3B-70B	Large autoregressive model with designed MLP architecture.
GPT-3 [101]	175B	Large autoregressive model with GPT structure.
Megatron-Turing [102]	530B	Large autoregressive model with GPT structure.
GPT-4 [103]	-	-

size, large language models have brought impressive performance in a wide range of language tasks such as machine translation, question-answering, and dialogue system. There are a few variations in the latest LLMs, such as BERT-based encoder-only language models [22, 104–110], encoder-decoder language models [21, 111–117], and GPT-based language models [23, 101, 102, 118–128]. These variations are succinctly presented in Table 1, which enumerates a subset of widely recognized large language models. Notably, there is a discernible trend of exponential growth in model size across these models.

LLMs have emerged as a prominent approach in natural language processing, wherein the size of small language models is scaled up significantly. For instance, GPT-3 [118] substantially increases the parameters of the GPT structure from 117M to a staggering 175B. The training of LLMs heavily relies on parallel strategies and distributed computing, with several frameworks capable of supporting large-scale training, including DeepSpeed [129, 130], Megatron [96], and colossal-ai [131]. The expansion in model size enables LLMs to possess capabilities that are absent in smaller language models [132]. One notable advantage of LLMs is their ability to perform in-context training, which allows the model to process examples before engaging in a specific task. Additionally, LLMs have demonstrated proficiency in solving intricate reasoning problems, such as mathematical induction [133].

Text-to-image models rely on text encoders to convert textual sentences into meaningful representations that can be utilized for image generation. An effective text encoder plays a crucial role in capturing the essential semantic and contextual information contained within the text description. This information is then utilized to guide the generation process of the output image. Recent advancements in LLMs have introduced a diverse array of pre-trained text encoders that can be seamlessly integrated into generative models for image generation tasks [3, 5, 30, 31, 39]. These models have already demonstrated remarkable capabilities in language feature engineering, which align well with the requirements of TTI generation. Leveraging these pre-trained encoders not only saves substantial time during TTI model's training, but also eliminates the need to invest excessive efforts in improving the model's performance for text embedding.

2.4 Large Vision Model

The advancements of language model architecture also spur their application to the research of vision models [18, 134–139] that are able to learn pixel-level information by encoding image patches with a Transformer-based encoder receiving spatial inputs rather than textual tokens. Vision models could be used in varied visual tasks such as image classification [18], object detection [140, 141], and segmentation [142–144]. The work of language models and vision models could be further extended to multi-modal learning [95, 145–156] that finetunes image-text data with the combination of language and vision models to understand the connections among different types of data. Thus, multi-modal learning could generate modalities different from the input data, such as images to captions.

Vision models could also be used as auto-regression models [4, 5] that encode images to a latent space representation for contrastive learning with text embeddings. Moreover, the integration of language and vision models has propelled the adoption of multi-modal diffusion for TTI tasks. These models effectively connect information from text embeddings and image embeddings, facilitating the diffusion process not only from text embeddings to image embeddings but also from image embeddings to ground truth image output.

Among all the language and vision models, CLIP [95] stands out as the preeminent choice, assuming a pivotal role in the construction of TTI generation. Originally conceived as a language-vision model, CLIP was initially tailored for tasks such as image classification, action recognition, and optical character recognition. CLIP [95] consists of a Transformer language encoder [20] and a ResNet or Vision-Transformer image encoder [18, 157]. Both aforementioned models map the input data to the latent space sharing the same embedding dimension. Training is based on the objective of maximizing the dot-product of text embeddings and image embeddings with the correct image-text pair, and minimizing the value of the incorrect ones, as indicated by Algorithm 1. In this way, CLIP shows impressive ability in matching images with any text inputs based on the highest dot-product value. CLIP also provides strong zero-shot learning abilities when transferring the model to downstream tasks. It is able to produce high accuracy classification results in zero-shot image datasets. CLIP has become a cornerstone for the development of TTI generation. A

Algorithm 3 CLIP Model

- 1: **for** one step during training **do**
- 2: Sample a batch of (Image,Text) pairs with size N from the dataset.
- 3: Feed images through the image encoder to get image embeddings and map them to multimodal embeddings I_1, I_2, \dots, I_N
- 4: Feed texts through the text encoder to get text embeddings and map them to multimodal embeddings T_1, T_2, \dots, T_N
- 5: Create the cosine similarity matrix with dimension $N \times N$.
- 6: Optimize the model based on the cross entropy that maximizes similarity scores over N matched image-text pairs and minimizes similarity scores over $N^2 - N$ unmatched pairs.
- 7: **end for**

pretrained CLIP text-encoder provides strong capability for text encoding and zero-shot representation while keeping a reasonably small model size, therefore, it is widely used for Diffusion TTI models [3, 4, 158, 159] and recent GAN TTI models [45, 46, 160]. Besides, CLIP enables multi-modal learning for contrastive text-image representations, which has spurred the development of prior-decoder TTI architecture [4, 158, 161] that generates images without arbitrary need of text embeddings.

3 TEXT-TO-IMAGE GENERATION

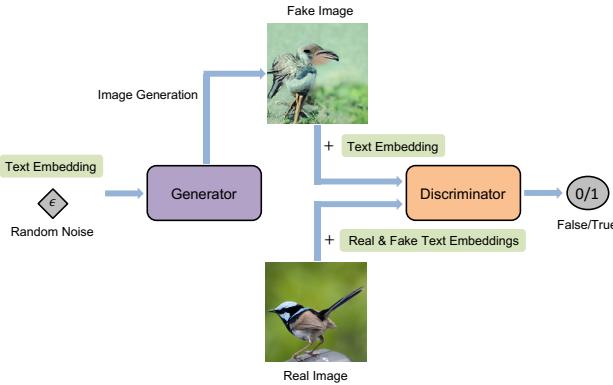


Fig. 4. A general structure of GAN-based text-to-image generation models, where the Generator takes text description as input and tries to generate fake images that ‘fool’ the Discriminator, and the Discriminator learns to discriminate images using fake images with real text, and real images with real and fake texts.

3.1 Prior to Large Model Era

Text-to-image generation could be traced back to the emergence of GAN models [29, 33, 162–164], and the performance is further improved by architecture and algorithm updates in the field [36, 37, 56, 165–170]. The development of GAN-based TTI models has usually involved in generating images with relatively fewer model parameters, and is limited to a domain specific set of dataset [171–173]. Early models have already shown the possibility to generate

images based on the text description. However, there exists a huge gap of image quality and diversity compared to the large TTI models. Due to the limitation of the size of the dataset and the performance of text encoders, GAN-based TTI is usually not capable of generating complicated and detailed images which are more likely to be sought after by user descriptions. Furthermore, GAN models do not have the capability to capture zero-shot text descriptions as well as diffusion TTI models. Despite the drawbacks for early TTI models, consistent research has improved the algorithms and architectures, which not only makes text-to-image generation possible, but enhances the image quality to a competitive level against state-of-the-art TTI models. In this section, we will list examples of GAN-based TTI models and elucidate the features that support text-to-image generation where some of them still impact on the creation of new models in the field.

3.1.1 Image refinement

Applying GAN models for text-to-image generation can be straightforwardly achieved through the incorporation of additional conditional information [174]. This process is typically implemented by taking text embeddings with Gaussian noise as inputs to the generator, images and text embeddings to the discriminator, as depicted in Fig 4. This ensures that the GAN model can possess the generative ability to align its output with the text descriptions, instead of creating random images from Gaussian noise unrelated to the input text. While this has been widely adopted in some prominent work [163], many new techniques are brought up subsequently that aim at enhancing both model performance and the quality of generated images. One common way is the stacking of multiple GAN components for image refinement. For example, StackGAN [162] is composed of two stages of GAN models. The first-stage generator takes the text embeddings from an LSTM encoder [175] and produces the images with 64×64 resolution. These smaller images are then fed into the first-stage discriminator with real image and text embeddings. The second-stage GAN repeats the same process whereas it generates the images with a better fidelity of 256×256 resolution.

StackGAN++ [176] improves the structure of StackGAN with a three-stage tree-like hierarchy. At the start of each stage, there are concat, residual and upsampling layers to pre-process inputs and generate image features for the next generator. Each of the three generators takes the up-sampled image features and produces images with size 64, 128 and 256, respectively. Besides, both StackGan and StackGan++ implement conditional augmentation that adds random Gaussian noise to the text embeddings as new conditional variables. This technique ensures that the model is able to have a stronger noise-proof ability and generate a wider variety of images.

3.1.2 Image-text alignments

Achieving precise alignment between text and image poses a significant challenge in TTI generation, a task that is more complex compared to other image generation applications. AttnGAN [33] proposes an effective way to measure the relevance of image-text pairs. As a similar approach to StackGAN++, AttnGAN [33] takes three refinement stages

with conditional augmentation and pre-processing layers, and samples the image from size 64 to 256. Two new features are introduced for image-text alignment. One is the introduction of the attention model (different from the attention mechanism) and the other one is a pre-trained Deep Attentional Multimodal Similarity Model (DAMSM) which takes in additional word features procured by the AttnGAN. In the DAMSM model, these extra word features contribute to quantifying the similarity between textual descriptions and synthesized images. This is done by calculating the relevance between words and each feature vector extracted from a pre-trained image encoder [173]. During training, AttnGAN extracts these additional word features together with text embeddings from a bi-directional LSTM [175]. Text embeddings are taken as conditional information and concatenated with noise as model input, and the word features are used within both DAMSM and attention models. In the attention model, word features are used to compute the word-context vector for each image feature representing the relevance of an image sub-region to each word, and then word-context vectors are fed into the next stage together with image features.

AttnGAN uses matching score $R(Q, D)$ for the image-text similarity calculated from the DASAM model, which could then derive the DASAM loss function:

$$L_{DASAM} = - \sum_{i=1}^M \log P(D_i|Q_i) - \sum_{i=1}^M \log P(Q_i|D_i) \\ - \sum_{i=1}^M \log P(\hat{D}_i|\hat{Q}_i) - \sum_{i=1}^M \log P(\hat{Q}_i|\hat{D}_i), \quad (16)$$

where $P(D_i|Q_i) = \frac{\exp(\gamma R(Q_i, D_i))}{\sum_{j=1}^m \exp(\gamma R(Q_i, D_j))}$ refers to the probability that image Q_i is matched with its corresponding text descriptions D_i . They are calculated in a level of pixel and word embeddings. $P(\hat{D}_i|\hat{Q}_i)$ refers to the probability considered in the global image and sentence level. A DASAM model absorbs more information to compare the similarity between texts and images. Therefore, adding a DASAM loss function upon training becomes a stronger way to align the images with text descriptions.

Cross-modal contrastive generative adversarial network (XMC-GAN) [165] is another model that considers regional features and attempts to enhance semantic fidelity with text descriptions. Unlike AttnGAN, XMC-GAN introduces an attentional self-modulation layer in the generator that takes use of both word embeddings and sentence embeddings as conditional inputs to generate finer-grained image regions. Besides, the discriminator also extracts features of both regional and global features from synthetic and real images to compare the mutual information with text and sentence embeddings. In addition to the general loss function of GAN model, XMC-GAN introduces contrastive losses that could maximize the lower bound of mutual information of: sentence embeddings and both real and fake images, word embeddings and both real and fake images' regions, and also the mutual information of real and fake images.

3.1.3 Image diversity enhancement

Image diversity is always a vital evaluation criterion of TTI models as text-image translation is not rigorously a

one-to-one mapping relationship. For example, one image could be represented by a different form of text expressions and the same semantic text descriptions could in theory represent different results in the output space. Early TTI models have the ability to generate images that align with text descriptions but discriminate images in diverse forms. Several research has targeted on and solve such problems [37, 177–179]. AC-GAN [177] modifies the general GAN structure to produce label probability distribution inside the generator together with the True/Fake predictions. Such improvements have allowed synthetic images to have an overall diversity coming from all labelled classes. This architecture is also adopted for text-to-image synthesis by TAC-GAN [178] with additional text inputs to both generator and discriminator. The discriminator components keep the layer to predict the probability distribution of class labels receiving fake images and text. This results in a better image diversity compared to the earlier GAN-based TTI generation models using only text embeddings such as GAN-CLS [29] and StackGAN [162].

3.2 Large Model with GAN Method

Recent advancements in large generative models have brought many opportunities to improve the performance of GAN-based TTI models, often accompanied by an escalation in model size, as depicted in Figure 5. Notably, one primary avenue of advancement lies in the adoption of large language models. The evolution of these robust linguistic architectures has imbued GAN models with the capacity to deploy more potent text encoders. This shift signifies a departure from conventional LSTM-based text encoders, frequently employed within previous GAN frameworks. For example, there has been researches that integrate GAN models with the pretrained CLIP model as a powerful tool for text-encoding, visual feature extraction and image-text alignment [44, 45, 160, 180, 181]. CLIPGLASS [160] uses a genetic algorithm to ensure that images generated from Generator has the maximized CLIP score with their corresponding text prompts. Similarly, FuseDream [181] aims at maximizing an augmented CLIP score that is more robust against adversarial attacks. Both LAFITE [44] and GALIP [45] uses CLIP model to adjust loss function for generator or discriminator to ensure GAN feature space is aligned with the pretrained CLIP.

Furthermore, augmenting model size and expanding training datasets have emerged as additional avenues for performance enhancement. GigaGAN [46], for instance, achieved this by scaling up model parameters by approximately 20-fold compared to its foundational counterpart [182]. Its training process leveraged expansive datasets such as LAION-2B [183] and COYO-700M [184]. Collectively, these enhancements usher in noteworthy progress for GAN TTI models, rendering their generative outcomes on par with the latest state-of-the-art autoregressive Transformer and diffusion-based TTI models.

3.3 Large Model with Autoregressive Method

Transformer models have been applied broadly in both natural language processing and computer vision, and have

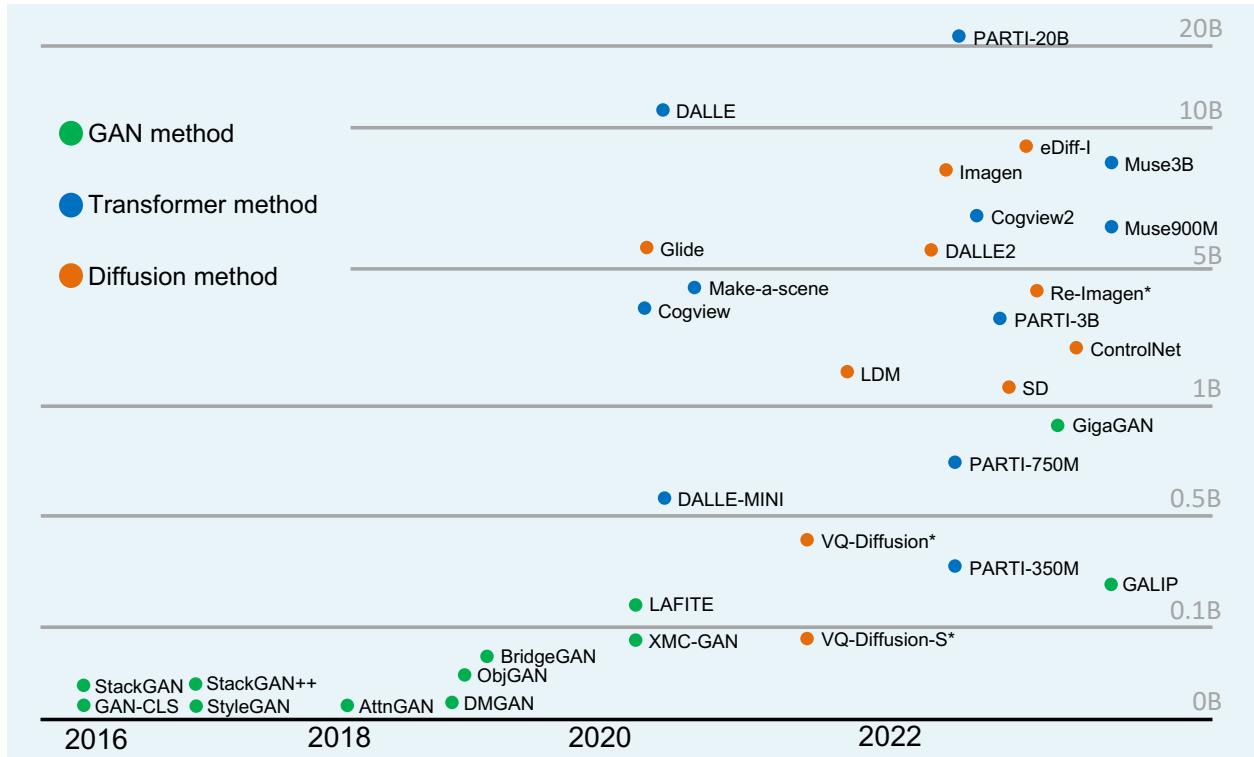


Fig. 5. Timeline of TTI model development, where green dots are GAN TTI models, blue dots are autoregressive Transformers and orange dots are Diffusion TTI models. Models are separated by their parameter, which are in general counted for all their components. Models with asterisk are calculated without the involvement of their text encoders.

shown an impressive result in text-to-image generation that gains the performance as close as the SOTA diffusion models. Autoregressive TTI model refers to the use of an autoregressive Transformer to predict image tokens from text. This is always employed with another encoder-decoder model that could encode an image to tokens and decode back to the image, and this structure plays a vital role in training the autoregressive Transformer. In recent years, with the development of Transformer-based models such as Vision Transformers and GPT, applying autoregression models to TTI produces impressive results. As shown in Fig 5, large autoregressive Transformer has become one of the dominant models in recent years.

3.3.1 DALLE

DALLE [39] is one of the latest text-to-image models that achieves impressive results with large models and datasets. The first stage involves training a discrete variational-autoencoder (dVAE) [185] that maps an image into 32×32 tokens where each grid has 8192 possible values. This is able to compress the 256×256 image data into a much smaller dimension with only slight information loss. In the next stage, text is encoded with byte pair encoding (BPE) [186] to tokens of size 256, and concatenated with image tokens. The new tokens are then fed into a decoder-only autoregressive Transformer to compute text-image joint distribution. This Transformer consists of 64 self-attention layers with causal mask for text attention and either row, column or convolution mask for spatial image attention.

Training of the overall procedure is based on maximizing the evidence lower bound [15, 66] of the text-image distribution:

$$\log p_{\theta, \psi}(x, y) \geq \mathbb{E}_z (\log p_\theta(x|y, z) - \beta D_{KL}(q_\phi(y, z|x) || p_\psi(y, z))). \quad (17)$$

The attainment of this goal unfolds in a two-stage process. Primarily, the dVAE undergoes training to optimize the parameters θ and ϕ . Subsequently, once these parameters are fixed, the autoregressive Transformer is engaged in training to maximize the aforementioned lower bound with respect to parameter ψ . To optimise the probability distribution of discrete parameters, they use the gumble-softmax function with an annealing factor $\tau = \frac{1}{16}$ to calculate a relaxed ELB when training the dVAE. When it comes to the training of the autoregressive Transformer, the image tokens will be sampled from the dVAE directly via an `argmax` operation without any gumble noises.

There are a few generic models of DALLE. DALLE-mini [187] has only 4B parameters which are 27 times less than the original model. It has similar structures as DALLE whereas using BART models as the text encoder and autoregressive image token decoder. Besides, DALLE-mini replaces dVAE with a VQGAN [188] for image encoding and decoding. Training of the BART decoder is based on the cross-entropy loss between the image tokens generated by BART decoder and VQGAN encoder. CLIP model is applied for choosing the best one among the images sampled from

the VQGAN decoder. In the later stage, there is a successful development of DALLE-mega which is an enlarged version of DALLE-mini.

3.3.2 Cogview

Cogview [189] improves upon the work of DALLE with a sophisticated set of features to address DALLE drawbacks. Cogview resembles the structure of DALLE, however, the backbone is an undirectional Transformer(GPT) and the image reconstruction model is a discrete autoencoder. Besides, compared to DALLE, Cogview develops a more efficient and general approach to stabilize the training process by regularizing the values of the LayerNorm and Attention operations of the Transformer.

Apart from the above innovation, Cogview [189] finetunes the model to adapt to more text-image tasks such as super-resolution, image captioning, and style changing, and achieves impressive results. The finetuning here allows the TTI model to extend its understanding of texts and images to a wide range of similar tasks to show a good ability of generalisation in the computer vision field. This philosophy is widely accepted by future text-to-image models.

3.3.3 PARTI

Google PARTI [5] is also a recent TTI model that can synthesize images with high fidelity, varied styles and compositions. Compared to its previous Imagen model, PARTI performs better in controlling the generation direction towards the target image with longer and more detailed text description. In addition to this, it achieves consistent image improvements by scaling up its Transformer models. PARTI has greatly boosted the image quality with the zero-shot FID score of 7.23 and finetuned FID score of 3.22 on MS-COCO. This has again proved the feasibility of using autoregressive models to achieve top performance in TTI generation.

PARTI also works in a very similar way to DALLE. It uses a vision Transformer [18] based model (ViT-VQGAN) as image tokenizer (encoder) and detokenizer (decoder). Alongside the encoder and decoder, there is also an autoregressive Transformer to encode text prompts and transfer them to image tokens. However, this Transformer is different from DALLE in that PARTI trains its own text-encoder in order to achieve a stronger textual representation. The first training stage involves training of the ViT-VQGAN [190] tokenizers to encode an image of size 256×256 to discrete visual tokens with a fixed length of 1024, and the detokenizer's purpose is to reconstruct the image tokens back to the image. PARTI adopts the ViT-VQGAN structure [190] and starts with a smaller configuration. Later on the tokenizer will be frozen and the detokenizer will be replaced with a larger sized and fine-tuned version for better visual acuity. In addition to this, a super-resolution model with 30M parameters is stacked onto the ViT-VQGAN model to convert the 256×256 image outputs to the size of 1024×1024 for a better visual illustration.

After the completion of training the tokenizer, the image tokens are then used to train the autoregressive Transformer. PARTI takes text-token inputs with a max length of 128 from a sentence-piece model with a vocabulary size of 16,000 sampled from the training data's corpus. This model implements the masked sparse attention mechanism for both

encoder and decoder of the autoregressive model. Training is performed on the models with parameters ranging from $350M$ to $20B$, which is done by scaling up the MLP dimensions with an expansion ratio of 4 and also doubling the attention heads whenever the MLP dimensions are doubled. This scaling of parameters within training has helped to explore TTI results on a wide range of model sizes.

3.3.4 Further Improvements to Autoregressive Architecture

Despite the notable achievements of recent autoregressive Transformers, their image token generation still relies on hundreds of iterative steps. To overcome this limitation, a potential enhancement involves exploring the utilization of non-autoregressive Transformers to achieve increased decoding parallelism. An example of such an approach is showcased in the work of MUSE [31]. MUSE [31] employs masked transformers that apply partial masking to image tokens during training. This methodology predicts the marginal distribution of masked tokens while leveraging unmasked text embeddings as conditional inputs. This training process encompasses both their base text-to-image Transformer and the super-resolution Transformer.

During inference, the model follows an iterative parallel decoding approach to predict all masked image tokens at each step. To improve image quality, only tokens with the highest prediction confidence are unmasked and used as inputs for subsequent steps. In contrast to autoregressive Transformers, which necessitate at least 256 steps, MUSE requires only 24 decoding steps for the base Transformer and 8 steps for the super-resolution Transformer. This design not only alleviates the time efficiency concerns of autoregressive Transformers through enhanced parallelism but also yields impressive text-to-image outcomes. For the purpose of comparative analysis in subsequent sections, MUSE is grouped with autoregressive Transformers.

3.4 Large Model with Diffusion Method

Diffusion models for TTI generation involve combining an extra conditional element to the U-net. This is usually achieved by encoding text descriptions with a strong text encoder, such as CLIP, Google T5 [21, 95], etc. Besides, [42] introduces classifier-guided diffusion process, which guides the diffusion process towards the direction of the class label as shown in the following equation:

$$\hat{\mu}(x_t|y) = \mu_\theta(x_t|y) + s * \Sigma_\theta(x_t|y) \nabla_{x_t} \log p_\phi(y|x_t), \quad (18)$$

where the mean value of predicted image distribution is further perturbed by the gradient of log probability of a target class predicted by a classifier, s is called guidance scale.

Classifier guided diffusion models have been further improved by [191] that introduces classifier-free guidance, in which a diffusion model is jointly trained with and without conditional information whilst changing the noise based on a scale factor. Showcased in Eq. (19), the inference could rely solely on the diffusion process without the need for classifier models:

$$\hat{\epsilon}(x_t|y) = \epsilon_\theta(x_t|\emptyset) + s * (\epsilon_\theta(x_t|y) - \epsilon_\theta(x_t|\emptyset)). \quad (19)$$

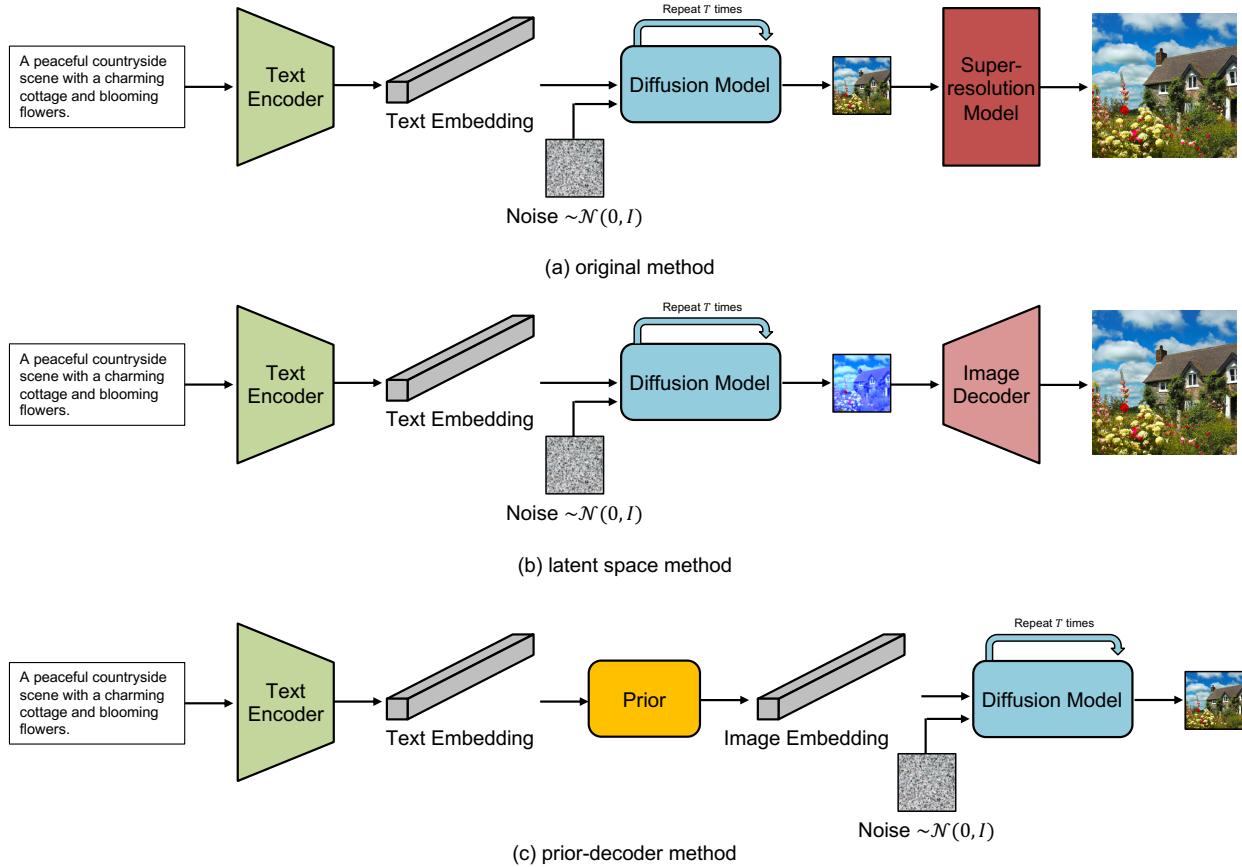


Fig. 6. Diffusion TTI model architectures, with original diffusion structure shown in (a), latent space structure shown in (b) and prior-decoder structure shown in (c), note that in prior-decoder structure, the output image has relatively small scale, therefore superresolution model could be also applied to this structure.

Together with the above methods, diffusion models pose impressive results in sampling images based on the text description. Figure 6 illustrated the most common diffusion TTI architectures. In this section we will explain in detail about different TTI generation models that take use of the diffusion method.

3.4.1 Early TTI applications with diffusion

GLIDE [40] is one of the first applications of diffusion models for TTI generation. It applies a fairly simple approach by generating a random Gaussian noise and inserting it into the diffusion model together with a CLIP-encoded text variable. Both classifier-guided and classifier-free diffusion are implemented during the training process for comparison. GLIDE also adopts the CLIP model [95] to the guided diffusion function. The guidance function is provided for calculating the similarity between the noise image and text caption in a certain timestamp. The process of the guidance function can be detailed by:

$$\hat{\mu}(x_t|c) = \mu_{\theta(x_t|c)} + s * \Sigma_{\theta}(x_t|c) \nabla_{x_t} (f(x_t) \cdot g(c)), \quad (20)$$

where $f(x_t) \cdot g(c)$ is the dot product of the CLIP image embeddings and text embeddings. The diffusion process will then perturb the mean of a timestamp with the gradient of the CLIP guided function. Although this is useful

for enhancing the performance, it increases the training requirement and decreases the image variation. Therefore, classifier-free guidance is also implemented to produce a better performance in inference tasks.

Imagen [30] is a major work developed on increasingly larger language models, and is regarded as one of the leading works for image fidelity. Their key hypothesis, other than providing improvements to the U-Net architecture, is in regards to the benefits of scaling language models, instead of directly scaling the size of the U-Net. Imagen investigated this through a T5-XXL [21] encoder which is tasked with mapping the input tokens into a sequence of embeddings. This then gets scaled up using two super resolution layers [30] from a 64×64 image to a 1024×1024 image. These super resolution layers are modified to be aware of the noise added to the diffusion models through a technique known as noise-level conditioning [30] to improve the sampling quality. This noise level conditioning is dependent on a cascaded diffusion model structure which is sourced from [192] and works by selecting augmentation values randomly during training and conditioning the diffusion model on this level. Moreover, there are a number of techniques they have to employ from relevant literature in order to tune their diffusion models. They adopt methods from such as

classifier free guidance [191], which is a technique that uses gradients from pre-trained models to improve the sample quality of the images. Yet, a diffusion model produced with classifier free guidance [191] can suffer from image fidelity issues due to the high guidance weights, and thus they have to make use of thresholding techniques to limit the effect of them.

Their work also introduces many small improvements to the U-Net architecture targeting the inference time, convergence speed and memory efficiency through their efficient U-Net structure. This involves integrating the work from [79, 88] to scale the U-Net skip connections by a constant value to accelerate convergence. In order to reduce the memory cost, Imagen prioritises the lower model parameters with lower resolutions as they contain more channels and thus allow them to increase the capacity of the model. As an attempt to improve inference speed, they reverse the order of down-sampling and up-sampling operations within their blocks such that the down-sampling occurs before the convolutions and the up-sampling occurs after the convolutions.

Their final text-to-image diffusion model consists of 2B parameters, and 600M/400M parameters for the two super resolution models, respectively. Using such a large model, they surprisingly did not find any over-fitting issues, and concluded that further training could be beneficial. They also pointed out that the text encoder size, rather than the U-Net’s size, is more paramount for image-text alignment and fidelity.

3.4.2 Prior-decoder TTI

With the impact of CLIP model that connects text and image, there has been research on a multi-model diffusion process that separates the generation of text embeddings and image embeddings. DALLE-2 [4] extends the work of GLIDE [40] and introduces prior component into their diffusion TTI generation model.

DALLE-2 takes CLIP [95] as the text encoder which encodes a text caption into latent space. The prior is an autoregressive Transformer or diffusion model that transfers the CLIP text latent space into the CLIP image latent space. When training the autoregressive prior, Principal Component Analysis (PCA) [193, 194] is applied for dimensionality reduction to induce a more efficient training process. When training the diffusion prior, instead of using the noise prediction model [17], DALLE-2 calculates mean square error (MSE) directly between the noise embedding, z_i , and the target embedding, z_0 . Both of these embeddings end up producing similar results. The decoder is also a diffusion network that converts the CLIP latent space into an image. DALLE-2 applies a similar strategy as mentioned in denoising diffusion probabilistic models. In the decoder diffusion model, CLIP embeddings and text captions are concatenated into the diffusion model as the guidance. To enable classifier-free training, they randomly drop the text caption roughly half of the time and set the CLIP embeddings to 0-10 percent of the total training time. In the prior diffusion model, they adopt a method by generating two image embedding samples from the diffusion model and choose the one that has the highest dot product with the CLIP text embedding. Unlike the traditional diffusion model

that predicts the noise distribution, DALLE-2 predicts the un-noised image directly within an arbitrary timestamp where the loss function is based on the mean squared error (MSE) with the true image distribution.

Despite the misclassification that occurs in the CLIP model, such as the notorious examples of classifying an apple to an iPod, this shortcoming never really materialised in DALLE-2. Although DALLE-2 [4] requires an extra prior model, it separates the training processes between text conversion and image conversion. In this way, the decoder does not take the text embeddings as a compulsory input to enable text-free training.

Based on the similar structure, Shift Diffusion [158] extends the work of DALLE-2 [4] and improves upon the sample efficiency of the prior model. Instead of sampling from a randomly normal gaussian space, Corgi [158] explores the possible distribution of the image input. Sampling the noise from a random space in the new distribution shortens the distance between the noise in the start timestep T , and the target output. Therefore, there are fewer timestamps that are needed compared to the DALLE-2 prior [4]. Noise sampling from the new distribution is based on clustering on the training data or simply train a new model using a neural network to approximate the data distribution. Selection of the noise of a image-text pair is based on the following calculation using CLIP:

$$c_y = \operatorname{argmax}_{1 \leq i \leq k} \operatorname{Sim}(\mu_i, \operatorname{CLIP}(text)), \quad (21)$$

where μ_i denotes the mean values of $noise_i$ from the noise collection, c_y is the noise index of selected Gaussian whose μ_i has the highest cosine similarity with the target image-text pair.

3.4.3 MOE diffusion

Refinement of noise prediction within diffusion TTI models stands to gain further refinement through the application of the mixture-of-expert structure [195], thereby offering an avenue for effectively denoising images in different stages of the diffusion process. This proposition aligns with the pioneering work of ERNIE-ViLG 2.0 [196] and Ediff [197], both of which underscore the different roles that diffusion model focuses during distinct timesteps.

In the initial stages of the diffusion process, wherein the input resembles random noise to a considerable extent, the diffusion model’s objective is to synthesize image intricacies in relation to this noise. This synthesis relies on the interplay of attention mechanisms with text embeddings. Conversely, as the input converges towards the target image, the diffusion model’s orientation shifts. It effectively “ignores” the textual guidance and pivots towards the enhancement of image quality itself.

In light of this bifurcation, a novel strategy emerges. The discrete stages of the diffusion process are inherently distinct, prompting the exploration of a multi-network approach. This approach envisions the training of multiple diffusion networks, each with a distinct expertise calibrated to denoise inputs within a particular stage. This differentiation leverages the expertise of each network to optimize denoising effectiveness and bolster the overall performance of the TTI model.

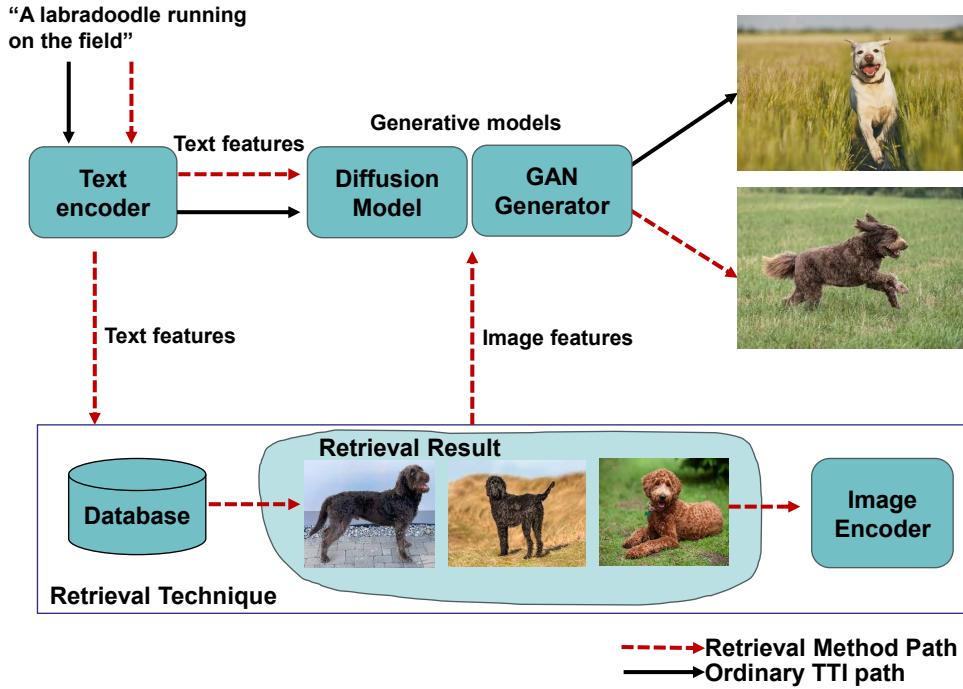


Fig. 7. A comparison over TTI model pipeline with and without retrieval technique, where the retrieval technique provides extra visual features for guidance, such techniques can be used for any generative models, examples of GAN and Diffusion method are shown in the figure.

3.4.4 Retrieval augmented diffusion models

Diffusion models have seen significant advances through combining their unique U-Net architectures with existing techniques [30, 198]. One such prominent example is the usage of a retrieval method [199–203] that sets up an external explicit memory dedicated for the TTI models. Retrieval models [204–207] have achieved state-of-the-art results in the past by using this external memory to circumvent adding an obnoxious amount of parameters. There are multiple approaches on how to achieve this. Popular ideas include changing the formulation of a standard parametric model into a semi-parametric one conditioned with the external memory or database [199]. Other approaches do not use an external retrieval memory but instead use part of the training data to avoid significant memory overhead. However, these latter approaches risk limited generalization capabilities due to their reliance on restricted datasets.

The retrieval technique has been applied to diffusion TTI models through a number of papers [199, 201, 203], with a common approach demonstrated in Fig 7. Retrieval-diffusion [199] achieves this through augmenting the model with the latent CLIP [95] space of a fixed set of images. The retrieval model is semi-parametric, encoding the non-trainable weights, ϵ_k , as a lookup. The dataset for the retrieval model, defined as D , is where the images are sampled from. The goal of this lookup is to obtain the k -nearest neighbour subsets of the dataset based on the input query. This is extremely similar to the work by [200]. The trainable parameters, w , take the results of the query, which are comprised of image patterns, in order to generate the

scenes. They could technically provide the results of this query to aid the model by sampling the images directly to update the trainable parameters. However, this would obviously involve significant unknowns and be strained by the dimensionality of the images. Thus, they make use of conditioning in order to reduce the dimensionality by encoding them in a lower dimensional space. This was achieved by using a fixed CLIP image encoder to convert all these examples onto a lower dimensional manifold $p_w(x|\cdot)$. The sampling strategy [199] can be expressed as:

$$p_{w,D_{\epsilon_k}}(x) = p_w(x|\{y|y \in M_D^{(k)}\}). \quad (22)$$

The database is thus used to approximate a distribution, $p(x)$, which they will sample queries from to build the k -nearest neighbours, $x \sim p(x)$. Since these are encoded in the latent space of CLIP, they use cosine similarity within this image feature space as the metric to search for nearest neighbours, $\epsilon_k^{Train}(x, D)$. When training is completed, the database is replaced with a specific dataset of a certain style. Hence, through the use of retrieval, they are able to apply quite flexible ad-hoc stylisation to their models merely by switching out different stylised datasets. Re-Imagen [203] is another updated version for Imagen that also takes the strategy of retrieval diffusion and achieves remarkable FID gain in the most used COCO dataset. The same retrieval technique could be also applied to GAN models [208, 209] and has achieved considerable results with external memory matching semantic relationship between texts and images.

3.4.5 Latent space diffusion TTI

Compressing images into low-dimensional latent space representations effectively reduces computational complexity while retaining generation capability in diffusion models. For example, latent diffusion model (LDM) [3] works by mapping the input data to a latent space 4-16 times smaller than the pixel space for the diffusion process. This is performed with the motivation that image details are derived from a very low-level representation of the inputs rather than a richer one informed by the position of a pixel in the context of the image. This method keeps the perceptually relevant details of the image and is able to efficiently sample during the reverse process. Therefore, it could greatly reduce the computation time for both training and inference while ensuring the quality of the sampled image.

To compress the image into a latent space, LDM adopts an auto-encoder that maps the image data into a lower-dimension representation with equivalent perceptual meaning. Given an image x with size $H \times W \times 3$, the encoder will construct it to the latent variable z with dimension $h \times w \times c$ by a down-sampling factor f where $f = \frac{H}{h} = \frac{W}{w} = 2^m$. The decoder is trained to reconstruct the latent space variable back to the pixel space, in a form that $\hat{x} = D(E(x))$. Perceptual loss and adversarial loss are introduced during training to enforce local realistic and avoid blurred images.

When fixing the auto-encoder, the diffusion model could be trained on the latent space with the loss function rewritten as follows:

$$L = E_{\mathcal{E}(x), \epsilon \sim \mathcal{N}(0, 1), t} [\|\epsilon - \epsilon_\theta(z_t, t)\|]. \quad (23)$$

Besides the above contribution, in order to augment the diffusion process with conditional information, stable-diffusion implements a cross-attention mechanism [20] by projecting all the conditional inputs to a domain-specific latent representation using domain-specific neural models $\tau_\theta(y)$, where y could be any conditional information such as text prompts, semantic map, and images. Attention [20] is then performed in each layer of the U-Net [19] backbone, with:

$$Q = W_Q^i \cdot \phi_i(z_t), \quad K = W_Q^i \cdot \tau_\theta(y), \quad V = W_V^i \cdot \tau_\theta(y),$$

given a U-Net layer i and intermediate result of noised image $\phi_i(z_t)$. The conditioning-processing network is trained jointly with the U-Net, using a slightly modified loss objective as follows:

$$L_{LDM} = \mathcal{E}_{\epsilon(x), y, \epsilon \sim \mathcal{N}(0, 1), t} [\|\epsilon - \epsilon_\theta(z_t, t, \tau_\theta(y))\|]. \quad (24)$$

LDM [3] works in a more efficient space with less imperceptible details and more semantics of the data. Besides, it adopts a neural network to process conditional information rather than simply concatenating it into the U-Net backbone. This processing stage makes the conditioning more domain-specific and performs better in guiding the denoising direction for the diffusion model. All these advantages have helped LDM to produce impressive results. LDM is further improved with enlarged training dataset and more varied pre-trained model selections, known as stable-diffusion [24], which has become one of the most popular choices for image generation.

Vector-quantized diffusion (VQ-diffusion) [68] is another latent space model. Without encoding images with an autoencoder, VQ-diffusion adopts the VQ-VAE [188] structure to reduce the image size into a discrete space to perform the diffusion process. VQ-diffusion introduces two techniques to perturb the latent space data. One is randomly masking the latent space value and the other is replacing one of the value to another randomly discrete value. Besides, VQ-diffusion also implements discrete diffusion algorithm that enables the diffusion model to perform the forward process and discrete process with discrete inputs. With all the contributions, VQ-diffusion is able to generate images efficiently in a much compressed discrete latent space.

In contrast to the majority of recent text-to-image generation methods, there is a significant appeal in creating more models that can be realistically and reliably run on commodity GPU hardware, which is a philosophy that latent models have pushed forth.

3.4.6 Sketch guided control

Another remarkable technique to generate images that meet users' expectation is the introduction of additional pre-defined input parameters, which enable users to steer the diffusion process with greater precision toward their envisioned image outcomes. By providing expected image specifications such as segmentation maps, object locations, and edge maps, users can now exert precise control over the generated images [159, 161, 219]. Spatext [161] extracts image segments during training process through a pre-trained segmentation model, and their corresponding CLIP image embeddings are computed. These image embeddings are then strategically stacked to mirror the shape and location of their respective segments within the original image, thus forming a comprehensive spatio-textual representation. This representation is then introduced as conditional input to the U-Net. This innovative training methodology empowers the diffusion model to generate images that closely adhere to the characteristics of the spatio map. During the inference process, users provide image sketches accompanied by local prompts that describe segmented areas. These prompts are processed through a prior model [4] to convert their text embeddings into image embeddings. Subsequently, these embeddings are stacked and mapped onto the spatial space where they align with the sketch's shapes and locations. This approach enables Spatext [161] to generate high-fidelity images, rich in object details that precisely match the user-provided shape, location, and color specifications.

Controlnet [159], meanwhile, extends the current stable-diffusion structure with replicate U-Net layers sharing the same weights with the original U-Net. Controlnet [159] receives extra pre-defined image information such as background colors and edge maps. During training, these additional image information are channeled into the replicate U-Net. Given that there is a convolution layer with zero weights attached to the end of each new U-Net layer, the initial training phase does not compromise the performance of the pre-trained stable diffusion model. However, as iterations progress, the new U-Net learns to leverage the extra image information to enhance predictions from the stable diffusion model. After short iterations of finetuning, the new

TABLE 2

Comparison of different dataset commonly used for computer vision, from small scale image-only dataset to large-scale image-text dataset.

Public Dataset	Size	Class Label	Captions	Description
CIFAR10 [171]	60K	10	X	Small-scale images with 10 categories.
LSUN-Church [210]	120K	X	X	Images with church scene.
LSUN-Bedroom [210]	3M	X	X	Images with bedroom scene.
ImageNet [173]	140M	1000	X	Manual-annotated dataset with 1000 non-overlapped classes.
CC3M [211]	3.1M	X	1:1	Auto-captioned real-world dataset.
CC12M [212]	12M	X	1:1	More diverse image collections based on CC3M.
COCOS [172]	120K	80	1:5	Real-world images with 80 object classes.
LAION400M [213]	400M	X	1:1	Large scale dataset with web-crawled images.
LAION5B [183]	5B	X	1:1	Enlarged Laion dataset with multiple languages.

TABLE 3

List of evaluation metrics and benchmarks for text-to-image models, where FID is the most commonly used metrics that covers image fidelity and diversity. While benchmarks covers thorough comparisons, they necessitate human evaluations, leading to increased time and effort investments.

Metric	Fidelity	Diversity	Text alignment	Human Evaluation	Language
IS[214]	✓	✓	X	X	-
FID[215]	✓	✓	X	X	-
R-precision[33]	X	X	✓	X	-
CLIP-Score[216]	X	X	✓	X	English
UniBench [217]	✓	X	✓	✓	Chinese, English
PaintSkills [218]	✓	X	✓	✓	English
DrawBench [30]	✓	X	✓	✓	English
PartiPrompts [5]	✓	X	✓	✓	English

U-Net is able to help guide the diffusion process towards the sketch and domain pre-defined by users.

4 COMPARISON OF APPROACHES

4.1 Experimental Setups and Selection of Dataset

The comparative options for training generative models are presented in Table 2. CIFAR10 [171], ImageNet [173], and LSUN [210] are labelled datasets that categorize images into various scenes and objects. Widely embraced, they form the bedrock for training fundamental generative models in scenarios of unconditional or class-conditional sampling[17, 42, 87]. Additionally, these datasets serve as valuable resources for training image encoders and super-resolution models[3], integral components of the TTI model framework. However, their applicability is generally not extended to training text-conditioned image generation.

MSCOCO [172] contains approximately 330k captioned images and over 200k have class labels. There are in total 80 object categories and 91 stuff categories and on average 3.5 categories per image. Reflecting diverse real-world scenarios, MSCOCO has evolved into a widely recognized evaluation dataset, assessing model performance across image segmentation [222, 223], classification, and object detection. Besides, MSCOCO [172] offers utility in training small-scale GAN TTI models in preliminary stages, given its compact size and computationally less demanding nature, while still yielding satisfactory results. Although no longer a primary dataset for training large TTI models, MSCOCO's high-quality image-text pairs continue to hold value as a crucial evaluation benchmark. In particular, they serve to rigorously test TTI model performance under zero-shot conditions, offering essential insights into model distinctions and capabilities.

LAION [213] emerges as a large dataset encompassing approximately 400 million image-text pairs in its initial iteration, a number that has since escalated to 5 billion [183] due to a continuous influx of new images. The image-text pairs are retrieved from random web pages with Common Crawl [224] and are filtered by CLIP score to ensure image-text alignment. Beyond the textual descriptions, LAION400M also equips users with CLIP embeddings and KNN indices associated with the image-text pairs. These ancillary resources serve as valuable assets for diverse analytical tasks involving clustering, K-nearest methods, and data subset filtration. This augmented resource environment significantly bolsters the training and inference process of retrieval-augmented TTI models, without the need of excessive time to collect extra data and build external database.

The Conceptual Captions Datasets [211, 212] also constitute substantial resources containing image-text pairs. In pursuit of a harmonious blend of qualities, informativeness, fluency, and learnability in the ensuing captions, a systematic automatic pipeline is employed. This pipeline orchestrates the extraction, filtration, and transformation of potential image/caption pairs, thereby achieving a fine balance in the dataset's attributes.

For the training of large TTI models, an enlarged dataset spanning diverse domains is imperative. This broadens the models' generalizability, enabling them to yield real-world images and effectively operating with zero-shot captions. Notably, the images curated from these large image-text datasets, LAION and Conceptual Captions, stand as premier choices for training large TTI models, encapsulating the requisite diversity to ensure robust performance and broad domain coverage.

TABLE 4

A comparison of the text to image methods discussed highlighting their date published, model configuration and evaluation results. For the model type, green dot refers to the GAN model TTI, blue dot refers to the autoregressive TTI and orange dot indicates the Diffusion TTI. For evaluation metrics, IS and FID score are provided under the evaluation of MSCOCO dataset in a zero-shot fashion. The last column provides the specific model size in scale of Million(M) or Billion(B); * : no zero-shot results found, use standard results instead.

Method	Date	Model Type	Data Size	Open Source	IS evaluation	FID evaluation	Model size
AttnGAN [33]	11/2017	●	120K	✗	20.80	35.49 *	13M
StyleGAN [34]	11/2017	●	120K	✗	20.80	35.49 *	-
Obj-GAN [220]	09/2019	●	120K	✓	24.09	36.52 *	34M
Control-GAN [221]	09/2019	●	120K	✓	23.61	33.10 *	-
DM-GAN [35]	04/2019	●	120K	✓	32.32	27.34 *	21M
XMC-GAN [165]	01/2021	●	120K	✗	30.45	9.33 *	90M
LAFITE [44]	11/2021	●	-	✓	26.02	26.94	150M
Retreival-GAN [208]	08/2022	●	120K	✗	29.33	9.13 *	25M
GigaGAN [46]	01/2023	●	-	✗	-	10.24	650M
GALIP [45]	03/2023	●	3M-12M	✓	-	12.54	240M
DALLE [39]	02/2021	●	250M	✗	-	27.5	12B
Cogview [189]	06/2021	●	300M	✓	-	27.1	4B
Make-A-Scene	03/2022	●	35M	✗	-	11.84	4B
Cogview2 [43]	05/2022	●	300M	✓	-	24.0	6B
PARTI-350M [5]	06/2022	●	~1000M	✗	-	14.10	350M
PARTI-20B [5]	06/2022	●	~1000M	✗	-	7.23	20B
DALLE-mini [187]	07/2021	●	250M	✗	-	-	~500M
MUSE-3B [31]	03/2023	●	~1000M	✗	-	7.88	7.6B
GLIDE [40]	12/2021	●	250M	✓	-	12.24	5B
VQ-diffusion-F [68]	11/2021	●	>7M	✓	-	13.86 *	370M
DALLE-2 [4]	04/2022	●	250M	✗	-	10.39	5.2B
Imagen [30]	05/2022	●	~860M	✗	-	7.27	7.6B
LDM [3]	08/2022	●	400M	✓	30.29	12.63	1.45B
eDiff-I [197]	11/2022	●	1000M	✗	-	6.95	9B
Shift Diffusion[158]	08/2022	●	900M	✓	-	10.88	-
Re-Imagen[203]	09/2022	●	50M	✗	-	6.88	~8B
ControlNet [159]	03/2023	●	-	✓	-	-	~2.2B

4.2 Evaluation Metrics and Benchmarks

Evaluating the performance of TTI models is a critical part of their training process. Researchers have adopted a number of evaluation metrics and benchmarks as a scale of TTI model performance, as shown in Table 3. Inception Score (IS) [214] is an algorithm that measures the image quality for generative models. It is applied as an evaluation metric for most of the GAN TTI models. Inception score [214] uses a pre-trained Inception model [225] to predict the conditional label or text distribution $P(y|x)$. The inception score, as detailed in Eq. (25), is calculated through the KL divergence of the conditional distribution and the real distribution:

$$\text{IS} = \exp(\mathcal{E}_x(KL(p(y|x)||p(y))). \quad (25)$$

The IS reflects the quality and diversity of the synthesised images. It could be maximized when the following two conditions are satisfied:

- 1) The conditional distribution $P(y|x)$ has low entropy for each generated image, implying that the generated image could be clearly identified to a class label by the pretrained model.
- 2) The marginal distribution of generated samples $\int p(y|x = G(z))$ has a high entropy, so the model is able to generate images with diversity.

The drawback of the IS is that it lacks the comparison between the generated images and real-world images, as it is calculated within the pretrained model [225]. Therefore, IS is only used in the early GAN TTI models, and a few of the updated Diffusion TTI models. The drawback of Inception

Score is resolved by the Féchet Inception Distance (FID) [226] with more consistency in comparison to real-world samples. The FID score calculation is detailed in Eq. (26) [215]:

$$d^2((\mathbf{m}, \mathbf{C}), (\mathbf{m}_w, \mathbf{C}_w)) = ||\mathbf{m} - \mathbf{m}_w||^2 + \text{Tr}(\mathbf{C} + \mathbf{C}_w - 2(\mathbf{C}\mathbf{C}_w)^{\frac{1}{2}}), \quad (26)$$

where d refers to the Féchet Distance [227]. The values of $\mathbf{m}, \mathbf{C}, \mathbf{m}_w, \mathbf{C}_w$ are the mean and covariance matrices of the evaluation dataset and generated samples in the last pooling layer of the pre-trained Inception model [225], respectively [214]. Besides, FID score is calculated in every mini-batch [215] during the training process. FID score [215] absorbs the similarity between synthetic images and real-world images by comparing the image distribution from both output samples and evaluation samples, this ensures the generated images to have high qualities and diversities as the evaluation images. It has become the standard metric to assess model performance for a wide range of GAN, autoregressive Transformer and Diffusion-based TTI models under MSCOCO dataset [172], as an alternative to the IS.

Despite the ability to involve the real world samples, FID is biased from the generator given the finite number of training samples [228]. The same reason also causes the IS score to be log negatively biased [228]. This problem could be alleviated using FID_∞ and IS_∞ [228]. Besides, FID could produce inconsistent results from human judgements [229] with unmatched data-sets between the Inception model and training model.

TABLE 5

Comparison of the visual result from different TTI models, where we select 4 models and generate 5 images respectively for each model. Text prompts are randomly selected.

Model	A peaceful countryside scene with a charming cottage and blooming flowers.	An adorable kitten playing with a colorful ball of yarn	A surreal dreamlike landscape with floating islands and rainbow-colored waterfalls	An underwater world teeming with colorful coral reefs and exotic fish.	A profile photo for a smart, engaging digital assistant.
GLIDE					
DALLE-2					
SD					
GALIP					

Beside the well-established IS and FID metrics, there exists a spectrum of other evaluation measures tailored to specific aspects of TTI model performance. The CLIP score, for instance, hones in on image-text alignment, leveraging a pretrained CLIP model for computation. An alternative approach, R-precision [33], quantifies visual-semantic affinity by ranking retrieval outcomes between extracted image and text features.

Furthermore, a comprehensive suite of human-evaluation benchmarks aids in TTI model comparison, focusing on aspects like image fidelity and text-image alignment. For example, Unibench [217] selects 200 text prompts in Chinese and English, with 100 simple-scene prompts and 100 complex-scene prompts, to collect raters' preference by the comparison of two model outputs. Imagen [30] introduces DrawBench, featuring 200 prompts across 11 categories, serving as a testing ground for diverse capabilities such as color rendering and object recognition. PartiPrompts [5] elevates the prompt count to 1600, encompassing distinct categories and facets that diverge from DrawBench's scope.

4.3 Comparison of TTI models

Table 4 details the model sizes and best evaluation results for a plethora of TTI models. For quantitative analyses, we provide IS and FID results tested on MSCOCO dataset in a zero-shot fashion. While these benchmarks cannot definitively adjudicate the performance of models in the absence of additional human-evaluation benchmarks, they do offer valuable insights into the performance of these models across a broad range of real-world scenarios.

Earlier GAN TTI models exhibit small model parameter counts less than 100M threshold. This limited model scale impedes the model performance and its applicability in large dataset training for zero-shot TTI generation [46]. These limitations have caused poor evaluation results with high FID and low IS even not in a zero-shot environment. Despite their substantial disparity with contemporary TTI models, recent work applying novel model architectures and scaling up model sizes have engendered a staggering boost in GAN TTI performance. These advancements have propelled GAN TTI models to rival with some of the

latest Diffusion and Autoregressive TTI models. Notable examples are GALIP [45] and GigaGAN [46], which exhibit impressive FID scores of 12.54 and 10.24, respectively, while maintaining a compact model size. Moreover, GAN TTI models have a natural computational advantage, generating images in a singular step, whereas both diffusion and autoregressive Transformers require iterative generation process. This combination of model size and computational efficiency position GAN TTI models as promising contenders for future TTI applications, underscoring their untapped potential.

Autoregressive Transformers typically encompass a higher number of model parameters, driven by the necessity for large language models and multiple attention layers. Recent progressions in architectural designs and the scale up of model sizes have markedly enhanced their performance, culminating in some of the most exceptional results within the domain of TTI generation. Their large training datasets enable further bolsters to the zero-shot capabilities. Notably examples are PARTI-20B [5] and MUSE [31], whose FID scores are only marginally higher than state-of-the-art Diffusion models.

Diffusion model, on the other hand, have demonstrated their capacity to achieve commendable synthetic outcomes even with moderate model parameters, as evident in the case of LDM [3]. Diffusion models have undergone substantial advancements in recent years, with significant techniques introduced to enhance their performance. Two noteworthy examples are eDiff-I [197] and Re-Imagen [203], which yield FID scores of 6.95 and 6.88, respectively, leveraging the advantages of MOE and retrieval techniques. These outcomes surpass the performance of all GAN-based TTI models and slightly better than state-of-the-art autoregressive Transformers. However, introduction of these techniques has inevitably increased the memory requirement and computation time, consequently leading to larger model sizes.

In addition to this, Table 5 gives a succinct visual comparison among distinct TTI models. Their effectiveness is assessed through the analysis of five randomly generated text prompts, drawn from ChatGPT-3 [1]. Each prompt encompasses a distinct zero-shot scenario. All models generate outputs that achieve an overall satisfactory quality. Nonetheless, upon closer examination, specific nuances in their performance become evident. For instance, GLIDE [40], representing an early diffusion model, exhibits a minor degree of information loss that occasionally results in misalignment with the provided text prompts. This discrepancy is especially obvious in its last column, where the text interpretation is severely misaligned. On the other hand, DALLE-2 [4], Stable-Diffusion [24], and GALIP [45] showcase exceptional image detailing and vivid color rendition. Despite these achievements, some images within their outputs display blurred objects or unrealistic scenes. It is noteworthy that stable-diffusion [3] and GALIP [45] encounter challenges when synthesizing images featuring faces, as exemplified by the text prompt "A peaceful countryside scene with a charming cottage and blooming flowers."

5 FUTURE OUTLOOK

5.1 Limitations and Potential Improvement of Current TTI Models

While various types of TTI models have showcased their respective strengths and limitations, there remains significant potential for the development of models that mitigate these shortcomings. The goal is to create resource-efficient and computationally expedient models capable of generating high-fidelity images. Such advancements would extend the applicability of these models to a broader range of local computing hardware and even mobile devices.

Currently, a trade-off between sampling efficiency and image quality persists across TTI models. State-of-the-art TTI generation models typically require around 50-200 iterative steps to synthesize high-quality images. Autoregressive Transformers, while achieving impressive results, demand substantial memory resources and employ an iterative generation process to predict image tokens. GAN-based TTI models, although promising, still exhibit a minor performance gap when compared to the leading autoregressive Transformers and Diffusion models.

Given these ongoing limitations, there remains a need for continued research efforts aimed at addressing the shortcomings of all types of TTI models. One avenue worth exploring is the investigation of model architectures and algorithms that facilitate TTI generation with minimal steps, such as GAN models [45, 46], non-autoregressive Transformers [31], and Diffusion models enhanced by recent distillation techniques [230, 231] and advanced ODE solvers [232]. By advancing these models to overcome their existing constraints, researchers can contribute to the realization of more versatile and efficient text-to-image generation techniques.

5.2 Beyond TTI: Embrace the Era of AIGC

5.2.1 TTI applications under the influence of AIGC

The development of AI-generated content (AIGC) has brought significant influence in many social and research aspects. New AIGC models have made it possible for people to access a vast amount of information on various topics, which will be beneficial to a wide range group of people and occupations. With the help of ChatGPT [2] generated content, people can quickly retrieve information on a wide range of subjects without significant expertise or having to spend time researching and reading through a vast amount of material. For writers, ChatGPT can also help to generate new ideas for stories, characters, and plotlines. Using these generative models, it is even possible to suggest titles, opening sentences, and themes for writers to explore with. Provided that more integration of AIGC models to the industry fields is happening, AIGC can be projected to see endless potential to boost the efficiency and productivity for people with their works.

Following the trend of AIGC, text-to-image models also expect the possibility to support varied generation tasks to apply in a wide range of general or industrial aspects. Some TTI models have enabled multiple generative tasks other than TTI generation. For example, stable-diffusion [3] not only supports TTI generation, but also enables image inpainting and super-resolution. Imagen [30] and GigaGAN

[46] both enable style transfer given image inputs. Such features potentially have a significant impact on both creativity and productivity on the fields of art creation and fashion design, with artists and designers seeking digital art and illustrations more easily and quickly than ever before. It posits the potential of providing these creators with the ability of creating art in different styles, including realistic, abstract, and surrealistic. This will also allow them to explore new possibilities and push the boundaries of what is possible. With further development of TTI models, we can also expect more plugin and software support such as Midjourney as well as integration to a broad range of tasks that suffice human needs, leading to a refined ecosystem with close connections to our daily life.

5.2.2 Extend the task of TTI

TTI techniques, with their generative models capable of generating new contents based on user inputs, are closely associated with other generative tasks that employ similar model architectures and algorithms. There has been development for video generation [233–244] with the involvement of diffusion techniques, which produces a sequence of image frames. Instead of using the original U-Net structure, video-diffusion model [233] implements 3D-UNet structure with each block representing 4D tensor to process fixed-length time-framed images. Current TTI techniques have also been successfully applied in diffusion video generation, such as classifier-free guidance that enables text-to-video [238], and latent diffusion [245–250] for sampling efficiency. New techniques have also been brought up to ensure the consistency among time frames, such as additional guidance information [233, 251]. This extension of TTI to text-to-video generation holds great potential for enhancing the field of digital art creation. However, it is important to note that text-to-video generation requires more computation time compared to TTI generation, as a video frame typically comprises numerous image frames to achieve smooth animation.

In addition to their application in the field of text-to-video, TTI models can also be extended to facilitate 3D generation using either Generative Adversarial Networks (GANs) [252–255] or Diffusion methods [256–264]. The aim of 3D generation through diffusion techniques is to approximate the distribution of a 3D object’s shape by utilizing random Gaussian points within 3D space. This approach enables the efficient creation of objects for 3D printing based on user descriptions, even if the desired object is not already present in an existing database. Consequently, the need for time-consuming manual design and drawing processes is eliminated, allowing for a more streamlined and expedited object creation process.

Despite the exciting prospects of TTI-based 3D generation, there are several challenges to overcome. One significant challenge is the complexity of representing three-dimensional shapes accurately [265]. While TTI models have achieved impressive results in generating 2D images, extending their capabilities to 3D objects requires handling additional spatial and structural information. Ensuring the fidelity and coherence of the generated 3D shapes remains an ongoing research area, with active investigations into

novel model architectures and loss functions that can effectively capture the intricacies of 3D geometry.

6 CONCLUSION

In our survey, we delve deeply into the world of TTI generation models, presenting a meticulous review of their diverse types. Our investigation encompasses their origins, highlighting the foundational models, intricate algorithms, and innovative architectures that underlie their remarkable capabilities. We assess their performance, conducting both quantitative and qualitative comparisons that shed light on their respective strengths and weaknesses. Within the context of the ever-evolving era of large models, we endeavor to unravel the intricate ways that TTI models interact with and complement the burgeoning domain of large models. By doing so, we lay the groundwork that has made important contributions to the field of TTI generation, seeking for potential improvements that could unleash the full potential of TTI models and harnessing their power to new heights. Moreover, our survey extends beyond the present, exploring the exciting possibilities that lie ahead for TTI models. We gaze into the horizon of future applications, envisioning novel ways they can contribute to various fields, from creative arts to data augmentation and beyond. Additionally, we examine the potential expansion of TTI models into other domains in the realm of AIGC that could change the way we retrieve information.

ACKNOWLEDGMENT

DAC is funded by an NIHR Research Professorship, an RAErg Research Chair, the NIHR Oxford Biomedical Research Centre, the InnoHK Centre for Cerebrocardiovascular Engineering, the Wellcome Trust under grant 217650/Z/19/Z, and the Oxford Pandemic Sciences Institute.

REFERENCES

- [1] C. Leiter, R. Zhang, Y. Chen, J. Belouadi, D. Larionov, V. Fresen, and S. Eger, “Chatgpt: A meta-analysis after 2.5 months,” *arXiv preprint arXiv:2302.13795*, 2023.
- [2] OpenAI, “Gpt-4 technical report,” 2023.
- [3] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, “High-resolution image synthesis with latent diffusion models,” in *CVPR*, 2022, pp. 10684–10695.
- [4] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen, “Hierarchical text-conditional image generation with clip latents,” *arXiv preprint arXiv:2204.06125*, 2022.
- [5] 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.
- [6] L. Tang, Y. Cai, J. Liu, Z. Hong, M. Gong, M. Fan, J. Han, J. Liu, E. Ding, and J. Wang, “Few-shot font generation by learning fine-grained local styles,” in *CVPR*, 2022, pp. 7895–7904.
- [7] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, “Unpaired image-to-image translation using cycle-consistent adversarial networks,” in *ICCV*, 2017, pp. 2223–2232.
- [8] T. Park, M.-Y. Liu, T.-C. Wang, and J.-Y. Zhu, “Semantic image synthesis with spatially-adaptive normalization,” in *CVPR*, 2019, pp. 2337–2346.
- [9] Y. Zhang, N. Huang, F. Tang, H. Huang, C. Ma, W. Dong, and C. Xu, “Inversion-based style transfer with diffusion models,” in *CVPR*, 2023, pp. 10146–10156.

- [10] N. Tumanyan, M. Geyer, S. Bagor, and T. Dekel, "Plug-and-play diffusion features for text-driven image-to-image translation," in *CVPR*, 2023, pp. 1921–1930.
- [11] C. Saharia, W. Chan, H. Chang, C. Lee, J. Ho, T. Salimans, D. Fleet, and M. Norouzi, "Palette: Image-to-image diffusion models," in *ACM SIGGRAPH 2022 Conference Proceedings*, 2022, pp. 1–10.
- [12] A. Lugmayr, M. Danelljan, A. Romero, F. Yu, R. Timofte, and L. Van Gool, "Repaint: Inpainting using denoising diffusion probabilistic models," in *CVPR*, 2022, pp. 11 461–11 471.
- [13] O. Avrahami, D. Lischinski, and O. Fried, "Blended diffusion for text-driven editing of natural images," in *CVPR*, 2022, pp. 18 208–18 218.
- [14] B. Yang, S. Gu, B. Zhang, T. Zhang, X. Chen, X. Sun, D. Chen, and F. Wen, "Paint by example: Exemplar-based image editing with diffusion models," in *CVPR*, 2023, pp. 18 381–18 391.
- [15] D. J. Rezende, S. Mohamed, and D. Wierstra, "Stochastic back-propagation and approximate inference in deep generative models," in *ICML*. PMLR, 2014, pp. 1278–1286.
- [16] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," *NeurIPS*, vol. 27, 2014.
- [17] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *NeurIPS*, vol. 33, pp. 6840–6851, 2020.
- [18] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," in *ICLR*, 2020.
- [19] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention—MICCAI 2015: 18th International Conference, Munich, Germany, October 5–9, 2015, Proceedings, Part III 18*. Springer, 2015, pp. 234–241.
- [20] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *NeurIPS*, vol. 30, 2017.
- [21] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," *JMLR*, vol. 21, no. 1, pp. 5485–5551, 2020.
- [22] J. D. M.-W. C. Kenton and L. K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of NAACL-HLT*, 2019, pp. 4171–4186.
- [23] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever *et al.*, "Language models are unsupervised multitask learners," *OpenAI blog*, vol. 1, no. 8, p. 9, 2019.
- [24] Stable diffusion online. [Online]. Available: <https://stablediffusionweb.com/>
- [25] N. Ruiz, Y. Li, V. Jampani, Y. Pritch, M. Rubinstein, and K. Aberman, "Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation," in *CVPR*, 2023, pp. 22 500–22 510.
- [26] "Novelai," <https://novelai.net/>.
- [27] "Midjourney," <https://www.midjourney.com/>.
- [28] E. Mansimov, E. Parisotto, J. L. Ba, and R. Salakhutdinov, "Generating images from captions with attention," *arXiv preprint arXiv:1511.02793*, 2015.
- [29] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee, "Generative adversarial text to image synthesis," in *ICML*. PMLR, 2016, pp. 1060–1069.
- [30] C. Saharia, W. Chan, S. Saxena, L. Li, J. Whang, E. L. Denton, K. Ghasemipour, R. Gontijo Lopes, B. Karagol Ayan, T. Salimans *et al.*, "Photorealistic text-to-image diffusion models with deep language understanding," *NeurIPS*, vol. 35, pp. 36 479–36 494, 2022.
- [31] H. Chang, H. Zhang, J. Barber, A. Maschinot, J. Lezama, L. Jiang, M.-H. Yang, K. Murphy, W. T. Freeman, M. Rubinstein *et al.*, "Muse: Text-to-image generation via masked generative transformers," *arXiv preprint arXiv:2301.00704*, 2023.
- [32] "Deep learning," *Nature Cell Biology*, vol. 521, no. 7553, pp. 436–444, May 2015.
- [33] 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 *CVPR*, 2018, pp. 1316–1324.
- [34] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila, "Analyzing and improving the image quality of stylegan," in *CVPR*, 2020, pp. 8110–8119.
- [35] M. Zhu, P. Pan, W. Chen, and Y. Yang, "Dm-gan: Dynamic memory generative adversarial networks for text-to-image synthesis," in *CVPR*, 2019, pp. 5802–5810.
- [36] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," *arXiv preprint arXiv:1511.06434*, 2015.
- [37] T. Qiao, J. Zhang, D. Xu, and D. Tao, "Mirorgan: Learning text-to-image generation by redescription," in *CVPR*, 2019, pp. 1505–1514.
- [38] W. Liao, K. Hu, M. Y. Yang, and B. Rosenhahn, "Text to image generation with semantic-spatial aware gan," in *CVPR*, 2022, pp. 18 187–18 196.
- [39] A. Ramesh, M. Pavlov, G. Goh, S. Gray, C. Voss, A. Radford, M. Chen, and I. Sutskever, "Zero-shot text-to-image generation," in *ICML*. PMLR, 2021, pp. 8821–8831.
- [40] A. Q. Nichol, P. Dhariwal, A. Ramesh, P. Shyam, P. Mishkin, B. McGrew, I. Sutskever, and M. Chen, "GLIDE: Towards photo-realistic image generation and editing with text-guided diffusion models," in *ICML*, vol. 162. PMLR, 2022, pp. 16 784–16 804.
- [41] C. B. Do, "The multivariate gaussian distribution," *Section Notes, Lecture on Machine Learning, CS*, vol. 229, 2008.
- [42] P. Dhariwal and A. Nichol, "Diffusion models beat gans on image synthesis," *NeurIPS*, vol. 34, pp. 8780–8794, 2021.
- [43] M. Ding, W. Zheng, W. Hong, and J. Tang, "Cogview2: Faster and better text-to-image generation via hierarchical transformers," *NeurIPS*, vol. 35, pp. 16 890–16 902, 2022.
- [44] Y. Zhou, R. Zhang, C. Chen, C. Li, C. Tensmeyer, T. Yu, J. Gu, J. Xu, and T. Sun, "Lafite: Towards language-free training for text-to-image generation," *arXiv preprint arXiv:2111.13792*, 2021.
- [45] M. Tao, B.-K. Bao, H. Tang, and C. Xu, "Galip: Generative adversarial clips for text-to-image synthesis," in *CVPR*, 2023, pp. 14 214–14 223.
- [46] M. Kang, J.-Y. Zhu, R. Zhang, J. Park, E. Shechtman, S. Paris, and T. Park, "Scaling up gans for text-to-image synthesis," in *CVPR*, 2023, pp. 10 124–10 134.
- [47] S. Frolov, T. Hinz, F. Raue, J. Hees, and A. Dengel, "Adversarial text-to-image synthesis: A review," *Neural Networks*, vol. 144, pp. 187–209, 2021.
- [48] M. Yeshevski, P. Kayal, and T. Subetha, "A survey on text description to image generation using gan," in *Soft Computing for Security Applications*. Springer Nature Singapore, 2023, pp. 665–675.
- [49] G. H. de Rose and J. P. Papa, "A survey on text generation using generative adversarial networks," *Pattern Recognition*, vol. 119, p. 108098, 2021.
- [50] R. Zhou, C. Jiang, and Q. Xu, "A survey on generative adversarial network-based text-to-image synthesis," *Neurocomputing*, vol. 451, pp. 316–336, 2021.
- [51] F.-A. Croitoru, V. Hondru, R. T. Ionescu, and M. Shah, "Diffusion models in vision: A survey," *TPAMI*, 2023.
- [52] L. Yang, Z. Zhang, Y. Song, S. Hong, R. Xu, Y. Zhao, Y. Shao, W. Zhang, B. Cui, and M.-H. Yang, "Diffusion models: A comprehensive survey of methods and applications," *arXiv preprint arXiv:2209.00796*, 2022.
- [53] K. O'Shea and R. Nash, "An introduction to convolutional neural networks," *arXiv preprint arXiv:1511.08458*, 2015.
- [54] J. Wu, "Introduction to convolutional neural networks," *National Key Lab for Novel Software Technology. Nanjing University. China*, vol. 5, no. 23, p. 495, 2017.
- [55] T. Amit, T. Shaharbany, E. Nachmani, and L. Wolf, "Segdiff: Image segmentation with diffusion probabilistic models," *arXiv preprint arXiv:2112.00390*, 2021.
- [56] M. Cha, Y. L. Gwon, and H. Kung, "Adversarial learning of semantic relevance in text to image synthesis," in *AAAI*, vol. 33, no. 01, 2019, pp. 3272–3279.
- [57] J. Wu, X. Li, S. X. H. Yuan, H. Ding, Y. Yang, X. Li, J. Zhang, Y. Tong, X. Jiang, B. Ghanem *et al.*, "Towards open vocabulary learning: A survey," *arXiv preprint arXiv:2306.15880*, 2023.
- [58] Y. Yang, H. Li, X. Li, Q. Zhao, J. Wu, and Z. Lin, "Sognet: Scene overlap graph network for panoptic segmentation," in *AAAI*, 2019.
- [59] C. Dong, C. C. Loy, K. He, and X. Tang, "Image super-resolution using deep convolutional networks," *TPAMI*, vol. 38, no. 2, pp. 295–307, 2015.
- [60] E. L. Denton, S. Chintala, R. Fergus *et al.*, "Deep generative image models using a laplacian pyramid of adversarial networks," *NeurIPS*, vol. 28, 2015.
- [61] H. Li, Y. Yang, M. Chang, S. Chen, H. Feng, Z. Xu, Q. Li, and Y. Chen, "Srdiff: Single image super-resolution with diffusion

- probabilistic models," *Neurocomputing*, vol. 479, pp. 47–59, 2022.
- [62] Z. Lu, J. Li, H. Liu, C. Huang, L. Zhang, and T. Zeng, "Transformer for single image super-resolution," in *CVPR*, 2022, pp. 457–466.
- [63] X. Li, H. Ding, W. Zhang, H. Yuan, J. Pang, G. Cheng, K. Chen, Z. Liu, and C. C. Loy, "Transformer-based visual segmentation: A survey," *arXiv preprint arXiv:2304.09854*, 2023.
- [64] Q. Zhou, X. Li, L. He, Y. Yang, G. Cheng, Y. Tong, L. Ma, and D. Tao, "Transvod: End-to-end video object detection with spatial-temporal transformers," *TPAMI*, 2022.
- [65] H. Yuan, X. Li, Y. Yang, G. Cheng, J. Zhang, Y. Tong, L. Zhang, and D. Tao, "Polyphonicformer: unified query learning for depth-aware video panoptic segmentation," in *ECCV*, 2022, pp. 582–599.
- [66] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.
- [67] K. Cho, B. van Merriënboer, D. Bahdanau, and Y. Bengio, "On the properties of neural machine translation: Encoder–decoder approaches," in *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*. Association for Computational Linguistics, 2014, p. 103.
- [68] S. Gu, D. Chen, J. Bao, F. Wen, B. Zhang, D. Chen, L. Yuan, and B. Guo, "Vector quantized diffusion model for text-to-image synthesis," in *CVPR*, 2022, pp. 10 696–10 706.
- [69] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training gans," *NeurIPS*, vol. 29, 2016.
- [70] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang *et al.*, "Photo-realistic single image super-resolution using a generative adversarial network," in *CVPR*, 2017, pp. 4681–4690.
- [71] A. Brock, J. Donahue, and K. Simonyan, "Large scale gan training for high fidelity natural image synthesis," in *ICLR*, 2018.
- [72] M.-Y. Liu and O. Tuzel, "Coupled generative adversarial networks," *NeurIPS*, vol. 29, 2016.
- [73] C. D. Prakash and L. J. Karam, "It gan do better: Gan-based detection of objects on images with varying quality," *IEEE Transactions on Image Processing*, vol. 30, pp. 9220–9230, 2021.
- [74] L. Liu, M. Muellly, J. Deng, T. Pfister, and L.-J. Li, "Generative modeling for small-data object detection," in *ICCV*, 2019, pp. 6073–6081.
- [75] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli, "Deep unsupervised learning using nonequilibrium thermodynamics," in *ICML*. PMLR, 2015, pp. 2256–2265.
- [76] D. Baranchuk, A. Voynov, I. Rubachev, V. Khrulkov, and A. Babenko, "Label-efficient semantic segmentation with diffusion models," in *ICLR*, 2021.
- [77] J. Wolleb, R. Sandkühler, F. Bieder, P. Valmaggia, and P. C. Cattin, "Diffusion models for implicit image segmentation ensembles," in *International Conference on Medical Imaging with Deep Learning*. PMLR, 2022, pp. 1336–1348.
- [78] S. Gong, M. Li, J. Feng, Z. Wu, and L. Kong, "Diffuseq: Sequence to sequence text generation with diffusion models," in *ICLR*, 2022.
- [79] C. Saharia, J. Ho, W. Chan, T. Salimans, D. J. Fleet, and M. Norouzi, "Image super-resolution via iterative refinement," *TPAMI*, vol. 45, no. 4, pp. 4713–4726, 2022.
- [80] M. Zhao, F. Bao, C. Li, and J. Zhu, "Egsde: Unpaired image-to-image translation via energy-guided stochastic differential equations," *NeurIPS*, vol. 35, pp. 3609–3623, 2022.
- [81] J. Austin, D. D. Johnson, J. Ho, D. Tarlow, and R. Van Den Berg, "Structured denoising diffusion models in discrete state-spaces," *NeurIPS*, vol. 34, pp. 17 981–17 993, 2021.
- [82] X. Li, J. Thickstun, I. Gulrajani, P. S. Liang, and T. B. Hashimoto, "Diffusion-lm improves controllable text generation," *NeurIPS*, vol. 35, pp. 4328–4343, 2022.
- [83] P. Yu, S. Xie, X. Ma, B. Jia, B. Pang, R. Gao, Y. Zhu, S.-C. Zhu, and Y. Wu, "Latent diffusion energy-based model for interpretable text modeling," in *ICML*, 2022.
- [84] D. Yang, J. Yu, H. Wang, W. Wang, C. Weng, Y. Zou, and D. Yu, "Diffsound: Discrete diffusion model for text-to-sound generation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2023.
- [85] M. Janner, Y. Du, J. Tenenbaum, and S. Levine, "Planning with diffusion for flexible behavior synthesis," in *ICML*. PMLR, 2022, pp. 9902–9915.
- [86] Z. Wang, J. J. Hunt, and M. Zhou, "Diffusion policies as an expressive policy class for offline reinforcement learning," in *ICLR*, 2022.
- [87] J. Song, C. Meng, and S. Ermon, "Denoising diffusion implicit models," in *ICLR*, 2020.
- [88] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole, "Score-based generative modeling through stochastic differential equations," *arXiv preprint arXiv:2011.13456*, 2020.
- [89] F. Bao, C. Li, J. Zhu, and B. Zhang, "Analytic-dpm: an analytic estimate of the optimal reverse variance in diffusion probabilistic models," in *ICLR*, 2021.
- [90] Y. Song and S. Ermon, "Generative modeling by estimating gradients of the data distribution," *NeurIPS*, vol. 32, 2019.
- [91] P. Vincent, "A connection between score matching and denoising autoencoders," *Neural Computation*, vol. 23, no. 7, pp. 1661–1674, 2011.
- [92] Y. Song, S. Garg, J. Shi, and S. Ermon, "Sliced score matching: A scalable approach to density and score estimation," in *Uncertainty in Artificial Intelligence*. PMLR, 2020, pp. 574–584.
- [93] T. Pang, K. Xu, C. Li, Y. Song, S. Ermon, and J. Zhu, "Efficient learning of generative models via finite-difference score matching," *NeurIPS*, vol. 33, pp. 19 175–19 188, 2020.
- [94] B. D. Anderson, "Reverse-time diffusion equation models," *Stochastic Processes and their Applications*, vol. 12, no. 3, pp. 313–326, 1982.
- [95] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, "Learning transferable visual models from natural language supervision," in *ICML*. PMLR, 2021, pp. 8748–8763.
- [96] M. Shoeybi, M. Patwary, R. Puri, P. LeGresley, J. Casper, and B. Catanzaro, "Megatron-lm: Training multi-billion parameter language models using model parallelism," *arXiv preprint arXiv:1909.08053*, 2019.
- [97] H. W. Chung, L. Hou, S. Longpre, B. Zoph, Y. Tay, W. Fedus, E. Li, X. Wang, M. Dehghani, S. Brahma *et al.*, "Scaling instruction-finetuned language models," *arXiv preprint arXiv:2210.11416*, 2022.
- [98] S. Zhang, S. Roller, N. Goyal, M. Artetxe, M. Chen, S. Chen, C. Dewan, M. Diab, X. Li, X. V. Lin *et al.*, "Opt: Open pre-trained transformer language models," *arXiv preprint arXiv:2205.01068*, 2022.
- [99] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar *et al.*, "Llama: Open and efficient foundation language models," *arXiv preprint arXiv:2302.13971*, 2023.
- [100] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaee, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale *et al.*, "Llama 2: Open foundation and fine-tuned chat models," *arXiv preprint arXiv:2307.09288*, 2023.
- [101] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell *et al.*, "Language models are few-shot learners," *NeurIPS*, vol. 33, pp. 1877–1901, 2020.
- [102] S. Smith, M. Patwary, B. Norick, P. LeGresley, S. Rajbhandari, J. Casper, Z. Liu, S. Prabhummoye, G. Zerveas, V. Korthikanti *et al.*, "Using deepspeed and megatron to train megatron-turing nlg 530b, a large-scale generative language model," *arXiv preprint arXiv:2201.11990*, 2022.
- [103] OpenAI, "Gpt-4 technical report," 2023.
- [104] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, "Albert: A lite bert for self-supervised learning of language representations," in *ICLR*, 2020.
- [105] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter," *arXiv preprint arXiv:1910.01108*, 2019.
- [106] L. Dong, N. Yang, W. Wang, F. Wei, X. Liu, Y. Wang, J. Gao, M. Zhou, and H.-W. Hon, "Unified language model pre-training for natural language understanding and generation," *NeurIPS*, vol. 32, 2019.
- [107] K. Clark, M.-T. Luong, Q. V. Le, and C. D. Manning, "Electra: Pre-training text encoders as discriminators rather than generators," in *ICLR*, 2019.
- [108] Z. Zhang, X. Han, Z. Liu, X. Jiang, M. Sun, and Q. Liu, "Ernie: Enhanced language representation with informative entities," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019, pp. 1441–1451.
- [109] S. Wang, Y. Sun, Y. Xiang, Z. Wu, S. Ding, W. Gong, S. Feng, J. Shang, Y. Zhao, C. Pang *et al.*, "Ernie 3.0 titan: Exploring larger-scale knowledge enhanced pre-training for language understand-

- ing and generation,” *arXiv preprint arXiv:2112.12731*, 2021.
- [110] P. He, X. Liu, J. Gao, and W. Chen, “Deberta: Decoding-enhanced bert with disentangled attention,” in *ICLR*, 2021.
- [111] V. Sanh, A. Webson, C. Raffel, S. H. Bach, L. Sutawika, Z. Alyafeai, A. Chaffin, A. Stiegler, T. Le Scao, A. Raja *et al.*, “Multitask prompted training enables zero-shot task generalization,” in *ICLR*, 2022.
- [112] A. Zeng, X. Liu, Z. Du, Z. Wang, H. Lai, M. Ding, Z. Yang, Y. Xu, W. Zheng, X. Xia *et al.*, “Glm-130b: An open bilingual pre-trained model,” in *ICLR*, 2022.
- [113] S. Soltan, S. Ananthkrishnan, J. FitzGerald, R. Gupta, W. Hamza, H. Khan, C. Peris, S. Rawls, A. Rosenbaum, A. Rumshisky *et al.*, “Alexatm 20b: Few-shot learning using a large-scale multilingual seq2seq model,” *arXiv preprint arXiv:2208.01448*, 2022.
- [114] Y. Wang, S. Mishra, P. Alipoormolabashi, Y. Kordi, A. Mirzaei, A. Arunkumar, A. Ashok, A. S. Dhanasekaran, A. Naik, D. Stap *et al.*, “Super-naturalinstructions: Generalization via declarative instructions on 1600+ nlp tasks,” in *EMNLP*, 2022.
- [115] L. Xue, N. Constant, A. Roberts, M. Kale, R. Al-Rfou, A. Sidhant, A. Barua, and C. Raffel, “mt5: A massively multilingual pre-trained text-to-text transformer,” in *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2021, pp. 483–498.
- [116] N. Muennighoff, T. Wang, L. Sutawika, A. Roberts, S. Biderman, T. L. Scao, M. S. Bari, S. Shen, Z.-X. Yong, H. Schoelkopf *et al.*, “Crosslingual generalization through multitask finetuning,” *arXiv preprint arXiv:2211.01786*, 2022.
- [117] B. Zoph, I. Bello, S. Kumar, N. Du, Y. Huang, J. Dean, N. Shazeer, and W. Fedus, “St-moe: Designing stable and transferable sparse expert models,” *arXiv preprint arXiv:2202.08906*, 2022.
- [118] A. Radford, K. Narasimhan, T. Salimans, I. Sutskever *et al.*, “Improving language understanding by generative pre-training,” 2018.
- [119] S. Zhang, S. Roller, N. Goyal, M. Artetxe, M. Chen, S. Chen, C. Dewan, M. Diab, X. Li, X. V. Lin *et al.*, “Opt: Open pre-trained transformer language models,” *arXiv preprint arXiv:2205.01068*, 2022.
- [120] A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, A. Roberts, P. Barham, H. W. Chung, C. Sutton, S. Gehrmann *et al.*, “Palms: Scaling language modeling with pathways,” *arXiv preprint arXiv:2204.02311*, 2022.
- [121] T. L. Scao, A. Fan, C. Akiki, E. Pavlick, S. Ilić, D. Hesslow, R. Castagné, A. S. Luccioni, F. Yvon, M. Gallé *et al.*, “Bloom: A 176b-parameter open-access multilingual language model,” *arXiv preprint arXiv:2211.05100*, 2022.
- [122] R. Thoppilan, D. De Freitas, J. Hall, N. Shazeer, A. Kulshreshtha, H.-T. Cheng, A. Jin, T. Bos, L. Baker, Y. Du *et al.*, “Lamda: Language models for dialog applications,” *arXiv preprint arXiv:2201.08239*, 2022.
- [123] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar *et al.*, “Llama: Open and efficient foundation language models,” *arXiv preprint arXiv:2302.13971*, 2023.
- [124] S. Black, S. Biderman, E. Hallahan, Q. Anthony, L. Gao, L. Golding, H. He, C. Leahy, K. McDonell, J. Phang *et al.*, “Gpt-neox-20b: An open-source autoregressive language model,” in *Proceedings of BigScience Episode# 5-Workshop on Challenges & Perspectives in Creating Large Language Models*, 2022, pp. 95–136.
- [125] S. Wu, O. Irsøy, S. Lu, V. Dabrowski, M. Dredze, S. Gehrmann, P. Kambadur, D. Rosenberg, and G. Mann, “Bloomberggpt: A large language model for finance,” *arXiv preprint arXiv:2303.17564*, 2023.
- [126] J. W. Rae, S. Borgeaud, T. Cai, K. Millican, J. Hoffmann, F. Song, J. Aslanides, S. Henderson, R. Ring, S. Young *et al.*, “Scaling language models: Methods, analysis & insights from training gopher,” *arXiv preprint arXiv:2112.11446*, 2021.
- [127] J. Hoffmann, S. Borgeaud, A. Mensch, E. Buchatskaya, T. Cai, E. Rutherford, D. d. L. Casas, L. A. Hendricks, J. Welbl, A. Clark *et al.*, “Training compute-optimal large language models,” *arXiv preprint arXiv:2203.15556*, 2022.
- [128] R. Nakano, J. Hilton, S. Balaji, J. Wu, L. Ouyang, C. Kim, C. Hesse, S. Jain, V. Kosaraju, W. Saunders *et al.*, “Webgpt: Browser-assisted question-answering with human feedback,” *arXiv preprint arXiv:2112.09332*, 2021.
- [129] S. Rajbhandari, C. Li, Z. Yao, M. Zhang, R. Y. Aminabadi, A. A. Awan, J. Rasley, and Y. He, “Deepspeed-moe: Advancing mixture-of-experts inference and training to power next-generation ai scale,” in *ICML*. PMLR, 2022, pp. 18332–18346.
- [130] S. Rajbhandari, J. Rasley, O. Ruwase, and Y. He, “Zero: Memory optimizations toward training trillion parameter models,” in *SC20: International Conference for High Performance Computing, Networking, Storage and Analysis*. IEEE, 2020, pp. 1–16.
- [131] S. Li, J. Fang, Z. Bian, H. Liu, Y. Liu, H. Huang, B. Wang, and Y. You, “Colossal-ai: A unified deep learning system for large-scale parallel training,” *arXiv preprint arXiv:2110.14883*, 2021.
- [132] W. X. Zhao, K. Zhou, J. Li, T. Tang, X. Wang, Y. Hou, Y. Min, B. Zhang, J. Zhang, Z. Dong *et al.*, “A survey of large language models,” *arXiv preprint arXiv:2303.18223*, 2023.
- [133] S. Welleck, J. Liu, X. Lu, H. Hajishirzi, and Y. Choi, “Naturalprover: Grounded mathematical proof generation with language models,” *Advances in Neural Information Processing Systems*, vol. 35, pp. 4913–4927, 2022.
- [134] Y. Li, K. Zhang, J. Cao, R. Timofte, and L. Van Gool, “Localityvit: Bringing locality to vision transformers,” *arXiv preprint arXiv:2104.05707*, 2021.
- [135] C. Riquelme, J. Puigcerver, B. Mustafa, M. Neumann, R. Jenatton, A. Susano Pinto, D. Keysers, and N. Houlsby, “Scaling vision with sparse mixture of experts,” *NeurIPS*, vol. 34, pp. 8583–8595, 2021.
- [136] H. Bao, W. Wang, L. Dong, Q. Liu, O. K. Mohammed, K. Aggarwal, S. Som, S. Piao, and F. Wei, “Vlmo: Unified vision-language pre-training with mixture-of-modality-experts,” *NeurIPS*, vol. 35, pp. 32897–32912, 2022.
- [137] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou, “Training data-efficient image transformers & distillation through attention,” in *ICML*. PMLR, 2021, pp. 10347–10357.
- [138] L. Yuan, Y. Chen, T. Wang, W. Yu, Y. Shi, Z.-H. Jiang, F. E. Tay, J. Feng, and S. Yan, “Tokens-to-token vit: Training vision transformers from scratch on imagenet,” in *ICCV*, 2021, pp. 558–567.
- [139] W. Wang, E. Xie, X. Li, D.-P. Fan, K. Song, D. Liang, T. Lu, P. Luo, and L. Shao, “Pyramid vision transformer: A versatile backbone for dense prediction without convolutions,” in *ICCV*, 2021, pp. 568–578.
- [140] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, “End-to-end object detection with transformers,” in *ECCV*. Springer, 2020, pp. 213–229.
- [141] Y. Fang, B. Liao, X. Wang, J. Fang, J. Qi, R. Wu, J. Niu, and W. Liu, “You only look at one sequence: Rethinking transformer in vision through object detection,” *NeurIPS*, vol. 34, pp. 26183–26197, 2021.
- [142] A. Kirillov, K. He, R. Girshick, C. Rother, and P. Dollár, “Panoptic segmentation,” in *CVPR*, 2019, pp. 9404–9413.
- [143] L. Ye, M. Rochan, Z. Liu, and Y. Wang, “Cross-modal self-attention network for referring image segmentation,” in *CVPR*, 2019, pp. 10502–10511.
- [144] S. Zheng, J. Lu, H. Zhao, X. Zhu, Z. Luo, Y. Wang, Y. Fu, J. Feng, T. Xiang, P. H. Torr *et al.*, “Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers,” in *CVPR*, 2021, pp. 6881–6890.
- [145] H. Bao, L. Dong, S. Piao, and F. Wei, “Beit: Bert pre-training of image transformers,” in *ICLR*, 2022.
- [146] D. Qi, L. Su, J. Song, E. Cui, T. Bharti, and A. Sacheti, “Imagebert: Cross-modal pre-training with large-scale weak-supervised image-text data,” *arXiv preprint arXiv:2001.07966*, 2020.
- [147] J. Cho, J. Lei, H. Tan, and M. Bansal, “Unifying vision-and-language tasks via text generation,” in *ICML*. PMLR, 2021, pp. 1931–1942.
- [148] Q. Xia, H. Huang, N. Duan, D. Zhang, L. Ji, Z. Sui, E. Cui, T. Bharti, and M. Zhou, “Xgpt: Cross-modal generative pre-training for image captioning,” in *Natural Language Processing and Chinese Computing: 10th CCF International Conference, NLPCC 2021, Qingdao, China, October 13–17, 2021, Proceedings, Part I 10*. Springer, 2021, pp. 786–797.
- [149] Y. Li, F. Liang, L. Zhao, Y. Cui, W. Ouyang, J. Shao, F. Yu, and J. Yan, “Supervision exists everywhere: A data efficient contrastive language-image pre-training paradigm,” in *ICLR*, 2022.
- [150] W. Kim, B. Son, and I. Kim, “Vilt: Vision-and-language transformer without convolution or region supervision,” in *ICML*. PMLR, 2021, pp. 5583–5594.
- [151] Z. Huang, Z. Zeng, B. Liu, D. Fu, and J. Fu, “Pixel-bert: Aligning image pixels with text by deep multi-modal transformers,” *arXiv preprint arXiv:2004.00849*, 2020.

- [152] L. H. Li, M. Yatskar, D. Yin, C.-J. Hsieh, and K.-W. Chang, "Visualbert: A simple and performant baseline for vision and language," *arXiv preprint arXiv:1908.03557*, 2019.
- [153] J. Chen, H. Guo, K. Yi, B. Li, and M. Elhoseiny, "Visualgpt: Data-efficient adaptation of pretrained language models for image captioning," in *CVPR*, 2022, pp. 18\,030–18\,040.
- [154] J. Lin, A. Yang, Y. Zhang, J. Liu, J. Zhou, and H. Yang, "Interbert: Vision-and-language interaction for multi-modal pretraining," *arXiv preprint arXiv:2003.13198*, 2020.
- [155] T. Wang, W. Jiang, Z. Lu, F. Zheng, R. Cheng, C. Yin, and P. Luo, "Vlmixer: Unpaired vision-language pre-training via cross-modal cutmix," in *ICML*. PMLR, 2022, pp. 22\,680–22\,690.
- [156] M. Zhuge, D. Gao, D.-P. Fan, L. Jin, B. Chen, H. Zhou, M. Qiu, and L. Shao, "Kaleido-bert: Vision-language pre-training on fashion domain," in *CVPR*, 2021, pp. 12\,647–12\,657.
- [157] T. He, Z. Zhang, H. Zhang, Z. Zhang, J. Xie, and M. Li, "Bag of tricks for image classification with convolutional neural networks," in *CVPR*, 2019, pp. 558–567.
- [158] Y. Zhou, B. Liu, Y. Zhu, X. Yang, C. Chen, and J. Xu, "Shifted diffusion for text-to-image generation," in *CVPR*, 2023, pp. 10\,157–10\,166.
- [159] L. Zhang and M. Agrawala, "Adding conditional control to text-to-image diffusion models," *arXiv preprint arXiv:2302.05543*, 2023.
- [160] F. A. Galatolo, M. G. Cimino, G. Vaglini *et al.*, "Generating images from caption and vice versa via clip-guided generative latent space search," in *Proceedings of the International Conference on Image Processing and Vision Engineering*, vol. 1, 2021, pp. 166–174.
- [161] O. Avrahami, T. Hayes, O. Gafni, S. Gupta, Y. Taigman, D. Parikh, D. Lischinski, O. Fried, and X. Yin, "Spatext: Spatio-textual representation for controllable image generation," in *CVPR*, 2023, pp. 18\,370–18\,380.
- [162] H. Zhang, T. Xu, H. Li, S. Zhang, X. Wang, X. Huang, and D. N. Metaxas, "Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks," in *ICCV*, 2017, pp. 5907–5915.
- [163] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee, "Generative adversarial text to image synthesis," in *ICML*. PMLR, 2016, pp. 1060–1069.
- [164] S. E. Reed, Z. Akata, S. Mohan, S. Tenka, B. Schiele, and H. Lee, "Learning what and where to draw," *NeurIPS*, vol. 29, 2016.
- [165] H. Zhang, J. Y. Koh, J. Baldridge, H. Lee, and Y. Yang, "Cross-modal contrastive learning for text-to-image generation," in *CVPR*, 2021, pp. 833–842.
- [166] J. Y. Koh, J. Baldridge, H. Lee, and Y. Yang, "Text-to-image generation grounded by fine-grained user attention," in *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, 2021, pp. 237–246.
- [167] Z. Zhang, Y. Xie, and L. Yang, "Photographic text-to-image synthesis with a hierarchically-nested adversarial network," in *CVPR*, 2018, pp. 6199–6208.
- [168] L. Gao, D. Chen, J. Song, X. Xu, D. Zhang, and H. T. Shen, "Perceptual pyramid adversarial networks for text-to-image synthesis," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, 2019, pp. 8312–8319.
- [169] G. Yin, B. Liu, L. Sheng, N. Yu, X. Wang, and J. Shao, "Semantics disentangling for text-to-image generation," in *CVPR*, 2019, pp. 2327–2336.
- [170] Z. Li, Z. Geng, Z. Kang, W. Chen, and Y. Yang, "Eliminating gradient conflict in reference-based line-art colorization," in *ECCV*, 2022.
- [171] A. Krizhevsky, G. Hinton *et al.*, "Learning multiple layers of features from tiny images," 2009.
- [172] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *ECCV*. Springer, 2014, pp. 740–755.
- [173] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *CVPR*, 2009, pp. 248–255.
- [174] J. Gauthier, "Conditional generative adversarial nets for convolutional face generation," *Class project for Stanford CS231N: convolutional neural networks for visual recognition, Winter semester*, vol. 2014, no. 5, p. 2, 2014.
- [175] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE transactions on Signal Processing*, vol. 45, no. 11, pp. 2673–2681, 1997.
- [176] H. Zhang, T. Xu, H. Li, S. Zhang, X. Wang, X. Huang, and D. N. Metaxas, "Stackgan++: Realistic image synthesis with stacked generative adversarial networks," *TPAMI*, vol. 41, no. 8, pp. 1947–1962, 2018.
- [177] A. Odena, C. Olah, and J. Shlens, "Conditional image synthesis with auxiliary classifier gans," in *ICML*. PMLR, 2017, pp. 2642–2651.
- [178] A. Dash, J. C. B. Gamboa, S. Ahmed, M. Liwicki, and M. Z. Afzal, "Tac-gan-text conditioned auxiliary classifier generative adversarial network," *arXiv preprint arXiv:1703.06412*, 2017.
- [179] J. Johnson, A. Gupta, and L. Fei-Fei, "Image generation from scene graphs," in *CVPR*, 2018, pp. 1219–1228.
- [180] R. Gal, O. Patashnik, H. Maron, A. H. Bermano, G. Chechik, and D. Cohen-Or, "Stylegan-nada: Clip-guided domain adaptation of image generators," *ACM Transactions on Graphics*, vol. 41, no. 4, pp. 1–13, 2022.
- [181] X. Liu, C. Gong, L. Wu, S. Zhang, H. Su, and Q. Liu, "Fusedream: Training-free text-to-image generation with improved clip+ gan space optimization," *arXiv preprint arXiv:2112.01573*, 2021.
- [182] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila, "Analyzing and improving the image quality of stylegan," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 8110–8119.
- [183] C. Schuhmann, R. Beaumont, R. Vencu, C. Gordon, R. Wightman, M. Cherti, T. Coombes, A. Katta, C. Mullis, M. Wortsman *et al.*, "Laion-5b: An open large-scale dataset for training next generation image-text models," *NeurIPS*, vol. 35, pp. 25\,278–25\,294, 2022.
- [184] M. Byeon, B. Park, H. Kim, S. Lee, W. Baek, and S. Kim, "Coyo-700m: Image-text pair dataset," <https://github.com/kakaobrain/coyo-dataset>, 2022.
- [185] J. T. Rolfe, "Discrete variational autoencoders," in *ICLR*, 2017.
- [186] R. Sennrich, B. Haddow, and A. Birch, "Neural machine translation of rare words with subword units," in *54th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics (ACL), 2016, pp. 1715–1725.
- [187] B. Dayma, S. Patil, P. Cuenca, K. Saifullah, T. Abraham, P. Le Khac, L. Melas, and R. Ghosh, "Dall-e mini," *HuggingFace.com*. <https://huggingface.co/spaces/dallemini/dalle-mini> (accessed Sep. 29, 2022), 2021.
- [188] P. Esser, R. Rombach, and B. Ommer, "Taming transformers for high-resolution image synthesis," in *CVPR*, 2021, pp. 12\,873–12\,883.
- [189] M. Ding, Z. Yang, W. Hong, W. Zheng, C. Zhou, D. Yin, J. Lin, X. Zou, Z. Shao, H. Yang *et al.*, "Cogview: Mastering text-to-image generation via transformers," *NeurIPS*, vol. 34, pp. 19\,822–19\,835, 2021.
- [190] J. Yu, X. Li, J. Y. Koh, H. Zhang, R. Pang, J. Qin, A. Ku, Y. Xu, J. Baldridge, and Y. Wu, "Vector-quantized image modeling with improved vqgan," in *ICLR*, 2021.
- [191] J. Ho and T. Salimans, "Classifier-free diffusion guidance," *arXiv preprint arXiv:2207.12598*, 2022.
- [192] J. Ho, C. Saharia, W. Chan, D. J. Fleet, M. Norouzi, and T. Salimans, "Cascaded diffusion models for high fidelity image generation," *JMLR*, vol. 23, no. 1, pp. 2249–2281, 2022.
- [193] K. Pearson, "Lii. on lines and planes of closest fit to systems of points in space," *The London, Edinburgh, and Dublin philosophical magazine and journal of science*, vol. 2, no. 11, pp. 559–572, 1901.
- [194] P. Foret, A. Kleiner, H. Mobahi, and B. Neyshabur, "Sharpness-aware minimization for efficiently improving generalization," in *ICLR*, 2021.
- [195] N. Shazeer, A. Mirhoseini, K. Maziarz, A. Davis, Q. Le, G. Hinton, and J. Dean, "Outrageously large neural networks: The sparsely-gated mixture-of-experts layer," in *ICLR*, 2016.
- [196] Z. Feng, Z. Zhang, X. Yu, Y. Fang, L. Li, X. Chen, Y. Lu, J. Liu, W. Yin, S. Feng *et al.*, "Ernie-vilg 2.0: Improving text-to-image diffusion model with knowledge-enhanced mixture-of-denoising-experts," in *CVPR*, 2023, pp. 10\,135–10\,145.
- [197] Y. Balaji, S. Nah, X. Huang, A. Vahdat, J. Song, K. Kreis, M. Aittala, T. Aila, S. Laine, B. Catanzaro *et al.*, "Ediffi: Text-to-image diffusion models with an ensemble of expert denoisers," *arXiv preprint arXiv:2211.01324*, 2022.
- [198] H. Liu, X. Shen, F. Shang, F. Ge, and F. Wang, "Cu-net: Cascaded u-net with loss weighted sampling for brain tumor segmentation," in *Multimodal Brain Image Analysis and Mathematical Foundations of Computational Anatomy, 4th International Workshop, MBIA 2019, and 7th International Workshop, MFCA 2019, Held in Conjunction with MICCAI 2019, Shenzhen, China, October 17, 2019, Proceedings 4*. Springer, 2019, pp. 102–111.
- [199] A. Blattmann, R. Rombach, K. Oktay, J. Müller, and B. Ommer,

- "Semi-parametric neural image synthesis," *NeurIPS*, vol. 11, 2022.
- [200] S. Sheynin, O. Ashual, A. Polyak, U. Singer, O. Gafni, E. Nachmani, and Y. Taigman, "knn-diffusion: Image generation via large-scale retrieval," in *The Eleventh ICLR*, 2022.
- [201] R. Rombach, A. Blattmann, and B. Ommer, "Text-guided synthesis of artistic images with retrieval-augmented diffusion models," *arXiv preprint arXiv:2207.13038*, 2022.
- [202] B. Li, P. H. Torr, and T. Lukasiewicz, "Memory-driven text-to-image generation," *arXiv preprint arXiv:2208.07022*, 2022.
- [203] W. Chen, H. Hu, C. Saharia, and W. W. Cohen, "Re-imagen: Retrieval-augmented text-to-image generator," *arXiv preprint arXiv:2209.14491*, 2022.
- [204] S. Borgeaud, A. Mensch, J. Hoffmann, T. Cai, E. Rutherford, K. Millican, G. B. Van Den Driessche, J.-B. Lespiau, B. Damoc, A. Clark *et al.*, "Improving language models by retrieving from trillions of tokens," in *ICML*. PMLR, 2022, pp. 2206–2240.
- [205] U. Khandelwal, O. Levy, D. Jurafsky, L. Zettlemoyer, and M. Lewis, "Generalization through memorization: Nearest neighbor language models," in *ICLR*, 2019.
- [206] K. Guu, K. Lee, Z. Tung, P. Pasupat, and M. Chang, "Retrieval augmented language model pre-training," in *ICML*. PMLR, 2020, pp. 3929–3938.
- [207] S. Borgeaud, A. Mensch, J. Hoffmann, T. Cai, E. Rutherford, K. Millican, G. B. Van Den Driessche, J.-B. Lespiau, B. Damoc, A. Clark *et al.*, "Improving language models by retrieving from trillions of tokens," in *ICML*. PMLR, 2022, pp. 2206–2240.
- [208] X. Yuan, Z. Lin, J. Kuen, J. Zhang, and J. Collomosse, "Text-to-image generation via implicit visual guidance and hypernetwork," *arXiv preprint arXiv:2208.08493*, 2022.
- [209] B. Li, P. H. Torr, and T. Lukasiewicz, "Memory-driven text-to-image generation," *arXiv preprint arXiv:2208.07022*, 2022.
- [210] F. Yu, A. Seff, Y. Zhang, S. Song, T. Funkhouser, and J. Xiao, "Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop," *arXiv preprint arXiv:1506.03365*, 2015.
- [211] P. Sharma, N. Ding, S. Goodman, and R. Soricut, "Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, 2018, pp. 2556–2565.
- [212] S. Changpinyo, P. Sharma, N. Ding, and R. Soricut, "Conceptual 12m: Pushing web-scale image-text pre-training to recognize long-tail visual concepts," in *CVPR*, 2021, pp. 3558–3568.
- [213] C. Schuhmann, R. Vencu, R. Beaumont, R. Kaczmarczyk, C. Mullis, A. Katta, T. Coombes, J. Jitsev, and A. Komatsuzaki, "Laion-400m: Open dataset of clip-filtered 400 million image-text pairs," *arXiv preprint arXiv:2111.02114*, 2021.
- [214] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training gans," *NeurIPS*, vol. 29, 2016.
- [215] 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," *NeurIPS*, vol. 30, 2017.
- [216] J. Hessel, A. Holtzman, M. Forbes, R. Le Bras, and Y. Choi, "Clip-score: A reference-free evaluation metric for image captioning," in *EMNLP*, 2021, pp. 7514–7528.
- [217] 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.
- [218] J. Cho, A. Zala, and M. Bansal, "Dall-eval: Probing the reasoning skills and social biases of text-to-image generative models," *arXiv preprint arXiv:2202.04053*, 2022.
- [219] A. Voynov, K. Aberman, and D. Cohen-Or, "Sketch-guided text-to-image diffusion models," *arXiv preprint arXiv:2211.13752*, 2022.
- [220] W. Li, P. Zhang, L. Zhang, Q. Huang, X. He, S. Lyu, and J. Gao, "Object-driven text-to-image synthesis via adversarial training," in *CVPR*, 2019, pp. 12 174–12 182.
- [221] A. Shoshan, N. Bhonker, I. Kviatkovsky, and G. Medioni, "Gan-control: Explicitly controllable gans," in *ICCV*, 2021, pp. 14 083–14 093.
- [222] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," *arXiv preprint arXiv:1706.05587*, 2017.
- [223] E. Xie, P. Sun, X. Song, W. Wang, X. Liu, D. Liang, C. Shen, and P. Luo, "Polarmask: Single shot instance segmentation with polar representation," in *CVPR*, 2020, pp. 12 193–12 202.
- [224] "Common crawl," <https://commoncrawl.org/>.
- [225] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *CVPR*, 2016, pp. 2818–2826.
- [226] "Comptes rendus hebdomadaires des seances de l'academie des sciences," 1835–1965.
- [227] D. Dowson and B. Landau, "The fréchet distance between multivariate normal distributions," *Journal of multivariate analysis*, vol. 12, no. 3, pp. 450–455, 1982.
- [228] M. J. Chong and D. Forsyth, "Effectively unbiased fid and inception score and where to find them," in *CVPR*, 2020, pp. 6070–6079.
- [229] S. Liu, Y. Wei, J. Lu, and J. Zhou, "An improved evaluation framework for generative adversarial networks," *arXiv preprint arXiv:1803.07474*, 2018.
- [230] T. Salimans and J. Ho, "Progressive distillation for fast sampling of diffusion models," *arXiv preprint arXiv:2202.00512*, 2022.
- [231] X. Liu, C. Gong, and Q. Liu, "Flow straight and fast: Learning to generate and transfer data with rectified flow," *arXiv preprint arXiv:2209.03003*, 2022.
- [232] C. Lu, Y. Zhou, F. Bao, J. Chen, C. Li, and J. Zhu, "Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps," *Advances in Neural Information Processing Systems*, vol. 35, pp. 5775–5787, 2022.
- [233] J. Ho, T. Salimans, A. Gritsenko, W. Chan, M. Norouzi, and D. J. Fleet, "Video diffusion models," *arXiv preprint arXiv:2204.03458*.
- [234] W. Yan, Y. Zhang, P. Abbeel, and A. Srinivas, "Videogpt: Video generation using vq-vae and transformers," *arXiv preprint arXiv:2104.10157*, 2021.
- [235] Z. Chen, X. Tan, K. Wang, S. Pan, D. Mandic, L. He, and S. Zhao, "Infergrad: Improving diffusion models for vocoder by considering inference in training," in *ICASSP*. IEEE, 2022, pp. 8432–8436.
- [236] W. Harvey, S. Naderiparizi, V. Masrani, C. Weilbach, and F. Wood, "Flexible diffusion modeling of long videos," *NeurIPS*, vol. 35, pp. 27 953–27 965, 2022.
- [237] U. Singer, A. Polyak, T. Hayes, X. Yin, J. An, S. Zhang, Q. Hu, H. Yang, O. Ashual, O. Gafni *et al.*, "Make-a-video: Text-to-video generation without text-video data," in *The Eleventh ICLR*, 2022.
- [238] J. Ho, W. Chan, C. Saharia, J. Whang, R. Gao, A. Gritsenko, D. P. Kingma, B. Poole, M. Norouzi, D. J. Fleet *et al.*, "Imagen video: High definition video generation with diffusion models," *arXiv preprint arXiv:2210.02303*, 2022.
- [239] X. Wang, H. Yuan, S. Zhang, D. Chen, J. Wang, Y. Zhang, Y. Shen, D. Zhao, and J. Zhou, "Videocomposer: Compositional video synthesis with motion controllability," *arXiv preprint arXiv:2306.02018*, 2023.
- [240] M. Yang, Y. Du, B. Dai, D. Schuurmans, J. B. Tenenbaum, and P. Abbeel, "Probabilistic adaptation of text-to-video models," *arXiv preprint arXiv:2306.01872*, 2023.
- [241] J. Xing, M. Xia, Y. Liu, Y. Zhang, Y. Zhang, Y. He, H. Liu, H. Chen, X. Cun, X. Wang *et al.*, "Make-your-video: Customized video generation using textual and structural guidance," *arXiv preprint arXiv:2306.00943*, 2023.
- [242] W. Chen, J. Wu, P. Xie, H. Wu, J. Li, X. Xia, X. Xiao, and L. Lin, "Control-a-video: Controllable text-to-video generation with diffusion models," *arXiv preprint arXiv:2305.13840*, 2023.
- [243] F.-Y. Wang, W. Chen, G. Song, H.-J. Ye, Y. Liu, and H. Li, "Gen-l-video: Multi-text to long video generation via temporal denoising," *arXiv preprint arXiv:2305.18264*, 2023.
- [244] L. Ruan, Y. Ma, H. Yang, H. He, B. Liu, J. Fu, N. J. Yuan, Q. Jin, and B. Guo, "Mm-diffusion: Learning multi-modal diffusion models for joint audio and video generation," in *CVPR*, 2023, pp. 10 219–10 228.
- [245] Y. He, T. Yang, Y. Zhang, Y. Shan, and Q. Chen, "Latent video diffusion models for high-fidelity video generation with arbitrary lengths," *arXiv preprint arXiv:2211.13221*, 2022.
- [246] L. Khachatryan, A. Movsisyan, V. Tadevosyan, R. Henschel, Z. Wang, S. Navasardyan, and H. Shi, "Text2video-zero: Text-to-image diffusion models are zero-shot video generators," *arXiv preprint arXiv:2303.13439*, 2023.
- [247] D. Zhou, W. Wang, H. Yan, W. Lv, Y. Zhu, and J. Feng, "Mag-icvideo: Efficient video generation with latent diffusion models," *arXiv preprint arXiv:2211.11018*, 2022.
- [248] S. Yu, K. Sohn, S. Kim, and J. Shin, "Video probabilistic diffusion models in projected latent space," in *CVPR*, 2023, pp. 18 456–18 466.
- [249] H. Ni, C. Shi, K. Li, S. X. Huang, and M. R. Min, "Conditional image-to-video generation with latent flow diffusion models," in

- CVPR, 2023, pp. 18444–18455.
- [250] J. Karras, A. Holynski, T.-C. Wang, and I. Kemelmacher-Shlizerman, “Dreampose: Fashion image-to-video synthesis via stable diffusion,” *arXiv preprint arXiv:2304.06025*, 2023.
 - [251] Y. Ma, Y. He, X. Cun, X. Wang, Y. Shan, X. Li, and Q. Chen, “Follow your pose: Pose-guided text-to-video generation using pose-free videos,” *arXiv preprint arXiv:2304.01186*, 2023.
 - [252] J. Sun, X. Wang, L. Wang, X. Li, Y. Zhang, H. Zhang, and Y. Liu, “Next3d: Generative neural texture rasterization for 3d-aware head avatars,” in CVPR, 2023, pp. 20991–21002.
 - [253] K. Schwarz, Y. Liao, M. Niemeyer, and A. Geiger, “Graf: Generative radiance fields for 3d-aware image synthesis,” *NeurIPS*, vol. 33, pp. 20154–20166, 2020.
 - [254] R. Wu and C. Zheng, “Learning to generate 3d shapes from a single example,” *ACM Transactions on Graphics*, vol. 41, no. 6, pp. 1–19, 2022.
 - [255] P. Friedrich, J. Wolleb, F. Bieder, F. M. Thieringer, and P. C. Cattin, “Point cloud diffusion models for automatic implant generation,” *arXiv preprint arXiv:2303.08061*, 2023.
 - [256] X. Zeng, A. Vahdat, F. Williams, Z. Gojcic, O. Litany, S. Fidler, and K. Kreis, “Lion: Latent point diffusion models for 3d shape generation,” in *NeurIPS*, 2022.
 - [257] J. Gu, Q. Gao, S. Zhai, B. Chen, L. Liu, and J. Susskind, “Learning controllable 3d diffusion models from single-view images,” *arXiv preprint arXiv:2304.06700*, 2023.
 - [258] T. Wang, B. Zhang, T. Zhang, S. Gu, J. Bao, T. Baltrusaitis, J. Shen, D. Chen, F. Wen, Q. Chen *et al.*, “Rodin: A generative model for sculpting 3d digital avatars using diffusion,” in CVPR, 2023, pp. 4563–4573.
 - [259] G. Nam, M. Khelifi, A. Rodriguez, A. Tono, L. Zhou, and P. Guerrero, “3d-ldm: Neural implicit 3d shape generation with latent diffusion models,” *arXiv preprint arXiv:2212.00842*, 2022.
 - [260] J. R. Shue, E. R. Chan, R. Po, Z. Ankner, J. Wu, and G. Wetzstein, “3d neural field generation using triplane diffusion,” in CVPR, 2023, pp. 20875–20886.
 - [261] K.-H. Hui, R. Li, J. Hu, and C.-W. Fu, “Neural wavelet-domain diffusion for 3d shape generation,” in *SIGGRAPH Asia 2022 Conference Papers*, 2022, pp. 1–9.
 - [262] Z. Liu, Y. Feng, M. J. Black, D. Nowrouzezahrai, L. Paull, and W. Liu, “Meshdiffusion: Score-based generative 3d mesh modeling,” in ICLR, 2023.
 - [263] E. R. Chan, K. Nagano, M. A. Chan, A. W. Bergman, J. J. Park, A. Levy, M. Aittala, S. De Mello, T. Karras, and G. Wetzstein, “Generative novel view synthesis with 3d-aware diffusion models,” *arXiv preprint arXiv:2304.02602*, 2023.
 - [264] K. Nakayama, M. A. Uy, J. Huang, S.-M. Hu, K. Li, and L. J. Guibas, “Diffacto controllable part-based 3d point cloud generation with cross diffusion,” *arXiv preprint arXiv:2305.01921*, 2023.
 - [265] R. Wang, Y. Yang, and D. Tao, “Art-point: Improving rotation robustness of point cloud classifiers via adversarial rotation,” in CVPR, 2022.