IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 10 July 2023, accepted 12 August 2023, date of publication 18 August 2023, date of current version 23 August 2023.

*Digital Object Identifier 10.1109/ACCESS.2023.3306422*

SURVEY

# Recent Advances in Text-to-Image Synthesis: Approaches, Datasets and Future Research Prospects

**YONG XUAN TAN[1], CHIN POO LEE[1], (Senior Member, IEEE), MAI NEO[2], KIAN MING LIM[1], (Senior Member, IEEE), JIT YAN LIM[1], AND ALI ALQAHTANI[3,4]**

[1]Faculty of Information Science and Technology, Multimedia University, Melaka 75450, Malaysia
[2]Faculty of Creative Multimedia, Multimedia University, Cyberjaya, Selangor 63100, Malaysia
[3]Department of Computer Science, King Khalid University, Abha 61421, Saudi Arabia
[4]Center for Artificial Intelligence (CAI), King Khalid University, Abha 61421, Saudi Arabia

Corresponding author: Chin Poo Lee (cplee@mmu.edu.my)

This work was supported in part by the Fundamental Research Grant Scheme of the Ministry of Higher Education under Grant FRGS/1/2021/ICT02/MMU/02/4; in part by the Telekom Malaysia (TM) Research and Development Grant under Grant RDTC/190995; and in part by the Deanship of Scientific Research, King Khalid University, Saudi Arabia, under Grant RGP2/332/44.

**ABSTRACT** Text-to-image synthesis is a fascinating area of research that aims to generate images based on textual descriptions. The main goal of this field is to generate images that match the given textual description in terms of both semantic consistency and image realism. While text-to-image synthesis has shown remarkable progress in recent years, it still faces several challenges, mainly related to the level of image realism and semantic consistency. To address these challenges, various approaches have been proposed, which mainly rely on Generative Adversarial Networks (GANs) for optimal performance. This paper provides a review of the existing text-to-image synthesis approaches, which are categorized into four groups: image realism, multiple scene, semantic enhancement, and style transfer. In addition to discussing the existing approaches, this paper also reviews the widely used datasets for text-to-image synthesis, including Oxford-102, CUB-200-2011, and COCO. The evaluation metrics used in this field are also discussed, including Inception Score, Fréchet Inception Distance, Structural Similarity Index, R-precision, Visual-Semantic Similarity, and Semantic Object Accuracy. The paper also offers a compilation of the performance of existing works in the field.

**INDEX TERMS** Text-to-image synthesis, generative model, GAN, generative adversarial networks, review, survey.

## I. INTRODUCTION

Text-to-image synthesis is an emerging field that seeks to generate images based on textual descriptions. The ultimate goal is to create an automated model that can understand the visual representation of important words and produce corresponding image contents. This task is challenging because it involves multimodal learning with two modalities, text, and visual, that require high levels of creativity and fluidity. Despite the potential of text-to-image synthesis, there are still few publications in this field compared to other machine learning domains such as object recognition. This is because it is a complex task that requires the integration of natural language processing and computer vision.

One of the most widely used neural network architectures in text-to-image synthesis is the Generative Adversarial Networks (GANs) [1]. GANs have a generator network that synthesizes images and a discriminator network that evaluates the visual realism of the input images. Later, conditional GANs (cGAN) [2] were introduced to condition the generator with additional inputs, such as class labels. This conditioning property is particularly useful in text-to-image synthesis, as it

The associate editor coordinating the review of this manuscript and approving it for publication was Okyay Kaynak.

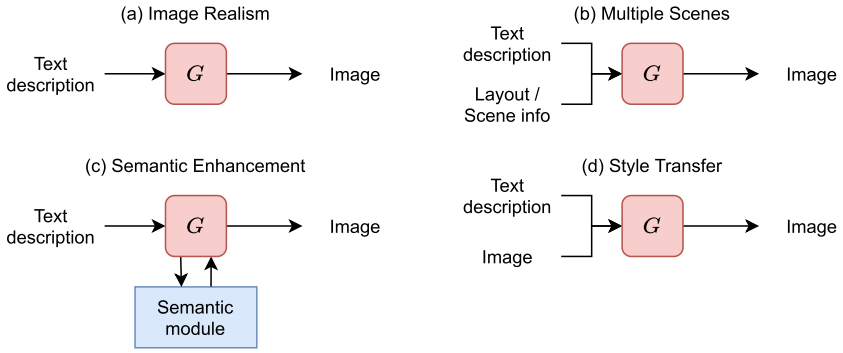



**FIGURE 1.** The general architecture of different categories of GAN-based text-to-image synthesis approaches.

allows the model to generate images that reflect the meaning of the text description. In recent years, several approaches based on cGAN have been proposed to improve the performance of text-to-image synthesis. These approaches aim to generate images that are not only visually realistic but also semantically consistent with the textual description. Some of the main challenges in text-to-image synthesis include ensuring the semantic consistency of the generated images, preserving the fine details, and handling multiple objects or scenes.

While previous surveys primarily focused on text-to-image synthesis model architectures (such as basic network, stacked architectures, attention mechanisms, Siamese architectures, layout text-to-image, dialog text-to-image, etc.) [3], [4], or enhancement GANs categories (Semantic Enhancement, Resolution Enhancement, Diversity Enhancement, and Motion Enhancement) [5], our survey paper takes a unique perspective by adopting a different taxonomy that specifically targets image realism, multiple scene synthesis, semantic enhancement, and style transfer.

By employing this taxonomy, our paper delves deeper into the specific techniques associated with these four crucial aspects of image synthesis. The image realism category focuses on generating visually realistic images that are indistinguishable from real ones. In contrast, the multiple scene category aims to generate multiple objects in a single image that correspond to different parts of the textual description. The semantic enhancement category focuses on generating images that are not only visually realistic but also semantically consistent with the given text. Lastly, the style transfer category focuses on altering specific parts of the image content based on the textual description. Figure 1 illustrates the four categories: image realism, multiple scene, semantic enhancement, and style transfer.

## II. FUNDAMENTAL

This section describes the fundamental components of the existing text-to-image synthesis approaches, namely Generative Adversarial Networks (GANs) and text encoder.

### A. GENERATIVE ADVERSARIAL NETWORKS

Generative Adversarial Networks (GANs) are a type of neural network architecture that consists of two different networks: a generator network $G$ and a discriminator network $D$. The generator network $G$ takes a random noise vector $z$ as input and generates a new image, while the discriminator network $D$ takes the output of the generator and a real image $x$ and builds a classifier that tries to discriminate between them. The objective of GANs is to train both networks against each other, where the generator network tries to produce an image that can be predicted as a real image by the discriminator network, while the discriminator network tries to distinguish the fake image (generated) from the real image.

To achieve this objective, GANs use a min-max game theory. The generator network tries to maximize the chance of fooling the discriminator network, while the discriminator network tries to detect all images generated by the generator network. Both networks are trained in a two-player min-max game, where the objective function is defined as:

$$\min_G \max_D V(D, G) = \mathbb{E}_{x \sim p_{data}} \left[\log D(x)\right] + \mathbb{E}_{z \sim p_z} \left[\log \left(1 - D(G(z))\right)\right] \quad (1)$$

where $z$ denotes the random noise sampled from a multivariate standard normal distribution $p_z = N(0, 1)$ and $x$ denotes the real images from true data distribution $p_{data}$.

Since the synthesized image content was completely random in GANs, cGAN was proposed. cGAN receives other inputs such as class labels and uses them to condition the generator from synthesizing desired outputs. With the additional conditioning variable, the objective of cGAN is defined as follows:

$$\min_G \max_D V(D, G) = \mathbb{E}_{(x,c) \sim p_{data}} \left[\log D(x, c)\right] + \mathbb{E}_{z \sim p_z, c \sim p_{data}} \left[\log \left(1 - D(G(z, c), c)\right)\right] \quad (2)$$

where $c$ denotes the additional conditioning variable for cGAN to produce the corresponding image content.

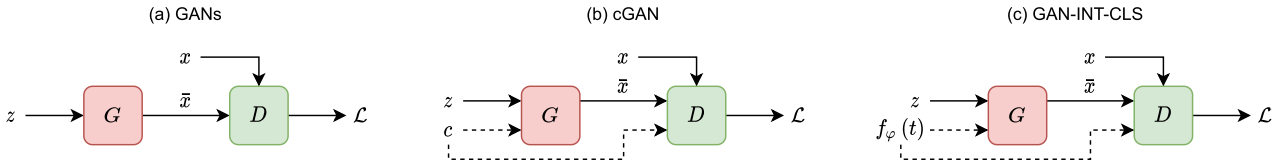



**FIGURE 2.** The overall architecture of GANs, cGAN, and GAN-INT-CLS. $x$ denotes the real image while $\bar{x}$ denotes the synthesized image. $f_\varphi$ denotes the text encoder while $t$ denotes the text description. $\mathcal{L}$ denotes the adversarial loss produced by the discriminator.

The overall architecture of cGAN is similar to GANs and is illustrated in Figure 2.

In the context of text-to-image synthesis, the conditioning property of cGAN is particularly useful as it allows the generator network to produce image content that reflects the meaning of the input text. Therefore, cGAN has been widely adopted in this field and has brought significant improvements. One notable example is GAN-INT-CLS [6], which uses text embedding as the conditioning variable. The generator takes a random noise vector and the text embedding of the input text as input and outputs a synthesized image that is conditioned on the input text. The discriminator network takes the synthesized image and the corresponding text embedding. It classifies whether the image is real or fake and whether the image is semantically consistent with the corresponding text description.

By conditioning the generator network on the input text, the synthesized images can be more consistent with the input text and have a higher degree of realism. Since the introduction of GAN-INT-CLS, numerous works have been proposed that build upon the cGAN architecture to further improve the quality of the synthesized images.

### B. TEXT ENCODER

In text-to-image synthesis, the goal is to generate realistic images that match the given textual description. However, to use the textual description as a conditioning variable, it needs to be transformed into a text embedding. Reed et al. [6] proposed a pre-trained hybrid of character-level convolutional neural network with a recurrent neural network (char-CNN-RNN) [7] to obtain the text embedding. The char-CNN-RNN consists of a character-level CNN or LSTM to encode the text and a GoogLeNet image classification model to encode the image. The network aims to minimize the distance between the encoded image and text by using the image vector to guide the text vector based on the image similarity. After training, the output text embedding contains the intended visual attributes of the image and is more effective than traditional text embeddings such as Word2Vec [8] and Bag-of-Words [9]. Besides char-CNN-RNN that has been widely used in prior works [10], [11], [12], [13], [14], [15], Dash et al. [16] proposed to use Skip-Thought vectors [17] as the conditioning variable.

However, using a fixed text embedding as a conditioning variable can cause a data discontinuity problem and affect the performance of the generator. This is due to the large dimension of the text embedding being transformed into a smaller dimension and most of the information is lost during this process. To solve this issue, Zhang et al. [10] introduced a text conditioning augmentation function that synthesizes more text embedding samples from a small amount of text embedding samples. The text embedding is transformed to produce the mean $c_\mu$ and covariance $c_\sigma$ of the text embeddings. The augmented text embedding is then computed by adding a random noise vector from a Gaussian distribution to the scaled covariance and adding the mean as shown below:

$$\bar{c} = v \times c_\sigma + c_\mu \tag{3}$$

where $v$ is the random noise vector from the Gaussian distribution.

To ensure the smoothness of the conditioning manifold and prevent overfitting, a regularization term is formulated as an additional objective function to the generator. The Kullback-Leibler (KL) divergence is computed between the conditioned Gaussian distribution and standard Gaussian distribution. This regularization term helps to increase the semantic consistency of the model and ensure that the synthesized images are associated with more semantically related text embeddings. This function has been continuously adopted by the rest of the prior works.

Additionally, Xu et al. [18] proposed to use the bi-directional LSTM (BiLSTM) to obtain the features of each word as a word vector and the features of the whole sentence as a sentence vector. They pre-trained a Deep Attentional Multimodal Similarity Model (DAMSM) to obtain the text encoder and image encoder for producing the text embedding that matched each image region. Recently, pre-trained transformer-based models such as BERT [19] have become popular in text-to-image synthesis for obtaining text embeddings [20], [21].

## III. IMAGE REALISM APPROACHES

Reed et al. [6] proposed the Matching-aware Manifold-interpolated GAN (GAN-INT-CLS) for synthesizing images based on text descriptions. By using a hybrid character-level convolutional recurrent neural network (char-CNN-RNN), GAN-INT-CLS is able to encode text descriptions into image features that can be used as input to a deep convolutional GANs (DCGAN). This enables the generator to produce images conditioned on the encoded text description, while the discriminator is trained to distinguish between realistic and semantically consistent images based on both the received image and encoded text description.

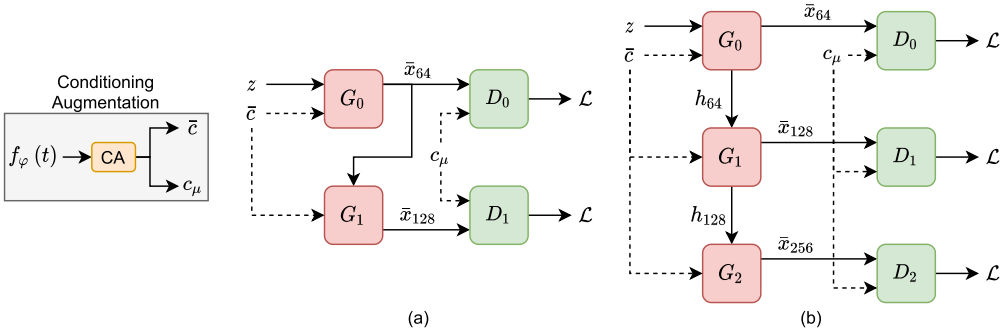



**FIGURE 3.** The overall architecture of (a) StackGAN and (b) StackGAN++. The main difference between them is that StackGAN is trained stage by stage while StackGAN++ is trained in an end-to-end manner. $h$ denotes the hidden features produced by the generator.

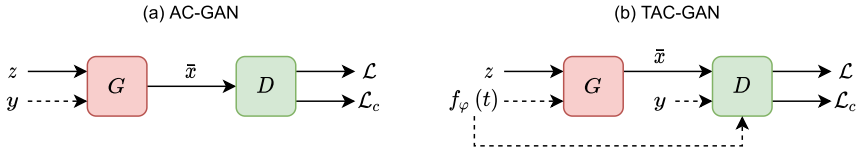



**FIGURE 4.** The overall architecture of AC-GAN and TAC-GAN. $y$ denotes the class label. The discriminator of both models is trained with an additional loss $\mathcal{L}_c$ from the auxiliary classification task.

While GAN-INT-CLS is effective at generating images based on text descriptions, it lacks the ability to control the location and pose of the object in the image. To address this, Reed et al. [22] presented the Generative Adversarial What-Where Network (GAWWN) that generates objects based on the text description at the location where they should appear. GAWWN includes a generator and a discriminator, and has two models for conditioning: a bounding-box-conditional GAWWN that uses a bounding box to set the location of the object, and a keypoint-conditional GAWWN that uses a set of coordinates to set the location of the object part. The conditioning model provides an additional supervised signal during the learning process of GAWWN.

Nguyen et al. [23] proposed an additional condition network, referred to as Plug and Play Generative Network (PPGN) that can control the generative model to produce different types of images. The architecture of PPGN consists of a generator, a discriminator, and a condition network. The generator produces images from the input noise vector and the output of the condition network. The discriminator distinguishes between real and generated images. The condition network is responsible for mapping the textual description to the corresponding noise vector that controls the generative model. The generator in PPGN is based on the Deep Generator Network-based Activation Maximization (DGN-AM) method, which produces good quality high-resolution images. The condition network is trained using a pre-trained model (VGG) to extract features for unseen image types. PPGN uses an iterative optimization approach to generate the noise vector that maximizes the diversity of the synthesized image through the condition network.

Zhang et al. [10] proposed StackGAN which is a multi-stage GAN architecture that generates high-resolution images with sufficient details and important information about the objects. The architecture of StackGAN consists of two GANs: Stage-I GAN and Stage-II GAN. Stage-I GAN constructs a low-resolution image with basic color and primitive shape of the object based on the text description. The layout of the background is built from random noise. The low-resolution image generated from Stage-I still has many defects and rough content. Hence, Stage-II GAN continues to enhance the low-resolution images from Stage-I GAN by fixing the defects in the images, enhancing the detail of the object, and improving the overall image quality to produce high-resolution realistic images.

However, StackGAN has a risk of mode collapse if Stage-I GAN cannot synthesize the image correctly based on the text description. Therefore, StackGAN++ [11] was introduced as an improved version of StackGAN. The architecture of StackGAN++ consists of multiple generators and discriminators arranged in a tree-like structure. Each generator generates images at different scales, from small to large, to achieve the final output. This multi-scale structure helps to mitigate the mode collapse problem and generates more diverse and realistic images. The architecture of StackGAN and StackGAN++ is illustrated in Figure 3.

Auxiliary classification is a technique used in Generative Adversarial Networks (GANs) to enhance image synthesis by increasing global coherence. The method involves utilizing class information to improve image structural coherence, and this has been demonstrated to be effective in prior works [24]. To further improve the performance of auxiliary classification

in GANs, Dash et al. [16] proposed a novel method called Text Conditioned Auxiliary Classifier GAN (TAC-GAN), as shown in Figure 4. TAC-GAN consists of a generator and discriminator, and the discriminator is trained with an additional loss from the auxiliary classification task. The discriminator of TAC-GAN predicts image realism and semantic consistency in addition to the class label of the input image and text description.

However, TAC-GAN may struggle to generate a variety of image types. To address this issue, Cha et al. [25] put forward an improved version of TAC-GAN called Text Conditioned Semantic Classifier GAN (Text-SeGAN). Text-SeGAN overcomes this limitation by using a triplet selection strategy during training. The triplet selection strategy selects the mismatched text-image pair between a real or fake image with different descriptions. This ensures that the model is trained to generate images that are semantically consistent with the input text descriptions, and not just limited to a pre-defined set of classes. Furthermore, Text-SeGAN uses a semantic classifier as the discriminator, which outputs classified results and predicts the semantic consistency of the input image pair along with the relationship with the class label. This allows the model to improve the semantic consistency between the generated image and the input text description, resulting in more diverse and realistic images.

Zhang et al. [26] proposed a single stream generator with a hierarchically nested discriminator structure that contains multiple discriminators with only one generator trained end-to-end, called Hierarchically-nested Adversarial Network (HD-GAN). This architecture can synthesize higher resolution images (512 $\times$ 512 pixels) with photorealistic content, and it has the advantage of training the generators to synthesize more complex images and increase the resolution of the generated image. The discriminators are located at the intermediate layer of the generator, and the generator needs to compete with all discriminators at different hierarchies to learn the features of different image scales provided by the discriminators. The lower resolution side constructs the basic image structure while the higher resolution side enhances the image details. The lower generator output can use the knowledge from higher discriminators due to end-to-end training. Compared to other GANs, HD-GAN does not require multiple internal conditioning from text descriptions like StackGAN and additional object labels like TAC-GAN.

One potential drawback of multi-stage architectures is that the final generated image is highly dependent on the initial image. If the initial image is not well-generated, the method may struggle to generate the final image accurately. To address this issue, Zhu et al. [27] proposed a Dynamic Memory Generative Adversarial Network (DM-GAN), which uses a multi-stage GAN architecture with a memory module to handle the initial image generated after the first stage of generation. The authors added a key-value memory structure to the DM-GAN model, where the feature of the initial image becomes a query to obtain features from the memory module and use them for image refinement. Additionally, DM-GAN leverages a memory writing gate to dynamically choose the words related to the generated images instead of using the same word throughout the whole image generation process. This approach helps to mitigate the issue of highly dependent generated images and leads to more diverse and realistic image generation.

Gao et al. [28] introduced a Perceptual Pyramid Adversarial Network (PPAN), which incorporates the pyramid framework into the generator architecture to produce multi-scale images directly from text descriptions. The PPAN architecture includes a generator and three discriminators, each with a specific focus on different aspects of the generated images. During training, the PPAN model employs a perceptual loss based on pre-trained VGG features, as well as an auxiliary classification loss, to synthesize highly realistic images. The inclusion of both perceptual and auxiliary classification losses further enhances the realism of the generated images.

Instead of using multiple discriminators in the networks, Huang et al. [29] presented a Hierarchically-fused Generative Adversarial Network (HfGAN) that contains only a single discriminator. The HfGAN approach adaptively fuses multi-scale visual features from different layers to synthesize large-scale images directly. Another approach proposed by Souza et al. [30] is a simpler text-to-image model architecture. This model is trained directly on 256 $\times$ 256 large-scale images without the involvement of multiple generators and discriminators. The authors also introduced a new sentence interpolation strategy for smoother conditional space.

Ensuring semantic consistency in initially generated images is crucial to producing high-quality results. To address this issue, Qi et al. [31] proposed a Multi-resolution Parallel Generative Adversarial Networks (MRP-GAN). The MRP-GAN structure maintains the initial image semantics throughout the generation process and includes an attention mechanism to fine-tune the fine-grained details of the synthesized images. MRP-GAN comprises one generator and three discriminators, including a response gate to merge multiple resolution feature maps to enhance image realism.

Inspired by the effectiveness of self-supervision in diversifying the model representation, Tan et al. [13] investigated self-supervision in the text-to-image synthesis field, proposing a self-supervised text-to-image synthesis (SSTIS) method. SSTIS creates additional supervision signals to enhance the performance of both the generator and the discriminator. The authors also integrated several techniques to stabilize the training of GANs for better performance. After that, several works [14], [15] have been proposed to investigate self-supervision into multi-stage training. Tan et al. [14] proposed Self-Supervised Bi-Stage GANs (SSBi-GAN) that applies self-supervision to all the stages of the model while Tan et al. [15] proposed Self-Supervision Text-to-Image GANs (SS-TiGAN) that applies self-supervision to only the last stage of the model. The self-supervision is able to diversify the model representation in

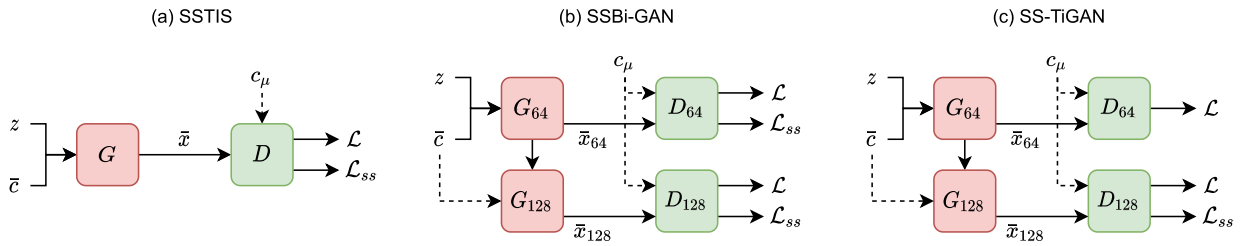



**FIGURE 5.** The overall architecture of SSTIS, SSBi-GAN, and SS-TiGAN. $\mathcal{L}_{ss}$ denotes the self-supervision loss.

**TABLE 1.** Summary of text-to-image synthesis approaches in the image realism category.

| Method | Description |
|---|---|
| GAN-INT-CLS [6] | First GAN-based text-to-image synthesis approach |
| GAWWN [22] | Keypoint or bounding box coordination |
| PPGN [23] | Additional conditional network |
| StackGAN [10] | Two-stage training pipeline |
| StackGAN++ [11] | Multi-stage end-to-end architecture |
| TAC-GAN [16] | Auxiliary classification signal |
| Text-SeGAN [25] | Triplet selection training strategy |
| HD-GAN [26] | Single stream generator with multiple hierarchically nested discriminators |
| DM-GAN [27] | Additional memory module |
| PPAN [28] | Perceptual loss and auxiliary classification loss |
| HfGAN [29] | Multi-stage generator with single discriminator |
| [30] | Large network with single generator and discriminator |
| MRP-GAN [31] | Response gate to merge multiple resolution feature maps |
| SSTIS [13] | Integrated self-supervision to enhance the model performance |
| SSBi-GAN [14] | Integrated self-supervision to multi-stage architecture for optimal performance |
| SS-TiGAN [15] | Integrated self-supervision into last stage of multi-stage model |
| [32] | Proposed DSM and ATD to improve both the generator and discriminator in the model |
| EruditeGAN [33] | Incorporate multiple relevant image distributions for each input to optimize the model performance |

multi-stage architecture to increase the synthesized image realism. The architectures of [13], [14], and [15] are illustrated in Figure 5.

To further improve text-to-image synthesis, Zhang et al. [32] presented a multi-perspective fusion method. Their approach involves improvements to both the generator and discriminator. In the generator, a dynamic selection method (DSM) is proposed to enhance the connection between the text and image features. During training, DSM can dynamically pick the relevant word vectors for various picture attributes, resulting in a more effective fusion of features. In the discriminator, the authors proposed a multi-class discriminant method (ATD) with a mask segmentation picture as an additional type to increase the discrimination performance. This method improves the accuracy of the discriminator, ensuring that the generated images are more realistic.

Another model, known as Erudite Generative Adversarial Network (EruditeGAN) was presented by Zhang et al. [33]. EruditeGAN aims to incorporate multiple image distributions that are relevant to the input image to familiarize the image distribution and synthesize high-quality images. By incorporating relevant image distributions, EruditeGAN ensures that the distribution of the image that needs to be synthesized is more prominent, resulting in high-quality outcomes.

The summary of the existing text-to-image synthesis with image realism approaches is presented in Table 1.

## IV. MULTIPLE SCENE APPROACHES

The synthesis of complex scenes from text descriptions requires an approach that can generate a semantic structure for the image. Hong et al. [34] proposed a hierarchical method that generates complex scenes by inferring image layout. The model consists of a layout generator and an image generator. The layout generator creates a semantic layout by placing bounding boxes around every object in the image, providing a useful structure to the image. The image generator then refines the object shape inside the bounding box and converts the layout to the final image. A single discriminator is responsible for predicting the image realism and semantic consistency of the generated image and text description. This approach allows for the generation of desired images by modifying the semantic layout, such as adding or deleting objects or changing the size and location of the object.

To control the objects in the generated image, Hinz et al. [35] proposed a method called the object pathway, which was incorporated into the generator and discriminator of AttnGAN. The authors used a bounding box and object label for object generation instead of a semantic layout to reduce computation. In Li et al. [36], an Object-driven attentive GAN (Obj-GAN) was described, which used an attention mechanism and semantic layout to focus on the object. By using these techniques and a multi-stage GAN architecture, Obj-GAN can effectively improve the object details based on related words and produce fine-grained high-quality images.

Another method is presented by Hinz et al. [37] where a global pathway is used to construct the overall image background structure and the location where the objects should be located. The authors added an object pathway to synthesize the desired objects to the background constructed by the global pathway. The object pathway learns the features of

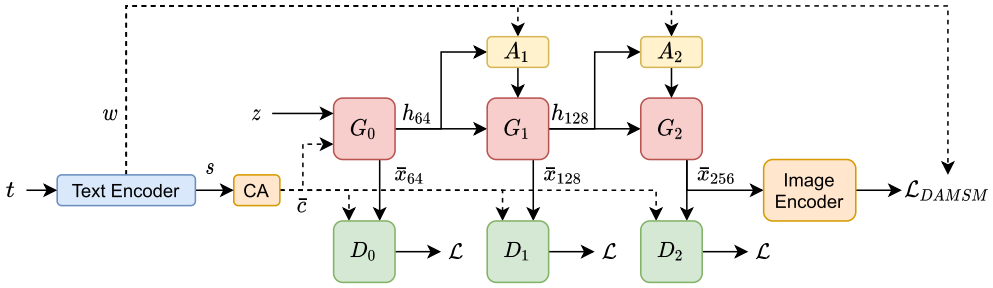



**FIGURE 6.** The overall architecture of AttnGAN. $A_1$ and $A_2$ denote the attention modules, $s$ and $w$ denote the sentence-level features and word-level features, and $\mathcal{L}_{DAMSM}$ denotes the DAMSM loss.

**TABLE 2.** Summary of text-to-image synthesis approaches in the multiple scenes category.

| Method | Description |
|---|---|
| Semantic layout [34] | Synthesise box and shape layout to refine multiple objects |
| Object pathway [35] | Synthesise multiple objects with lower computation |
| Obj-GAN [36] | Multi-stage attention model with the semantic layout |
| Chatpainter [38] | Dialogue as a conditioning variable |
| OP-GAN [37] | New global and object pathway module |
| [39] | Additional layout input pair for the discriminator |

each object, and the objects are generated iteratively with the corresponding text description and object class label. OP-GAN is able to synthesize a larger image compared to Obj-GAN by using a larger generator. In addition, the paper introduced a new evaluation metric called semantic object accuracy (SOA) that takes into consideration a single object, the image subregion, and the corresponding text description.

Sharma et al. [38] used dialogue instead of text description to provide more information about the scene to generate better quality images with multiple objects. The authors proposed Chatpainter, which uses a dialogue module that can be added to any text-to-image synthesis method. Their method is based on StackGAN, where the dialogue is encoded and combined with text description features before being sent to the conditional augmentation module.

Wang et al. [39] proposed an end-to-end text-to-image synthesis, which is capable of generating multi-object images using object and shape information. By fusing the synthesized semantic layout with text semantics and hidden visual features, the approach is able to manipulate complex image scenes. This approach involves training a GAN architecture, which mainly follows the StackGAN++ model, but with the additional input of image layout for the discriminators. During training, the network iteratively optimizes the spatial layouts to produce coarse-to-fine images. This means that the model first generates a rough sketch of the image, which gradually becomes more detailed and refined. The use of object and shape information helps to guide the image generation process, resulting in more accurate and visually coherent images. The summary of the existing text-to-image synthesis with multiple scene approaches is presented in Table 2.

## V. SEMANTIC ENHANCEMENT APPROACHES

One limitation of using global sentence features as the conditioning input is that it lacks fine-grained detail for each individual word, which can adversely affect the quality of the generated image.

In order to address this limitation, Xu et al. [18] proposed an approach called Attentional Generative Adversarial Network (AttnGAN). AttnGAN combines the attention technique with a multi-stage architecture from StackGAN++ to achieve fine-grained text-to-image synthesis. The model consists of a text encoder and an image encoder that is pretrained to capture the relationship between each word in the text description and the corresponding visual features in the generated image. By leveraging both word-level and sentence-level information, the Deep Attentional Multimodal Similarity Model (DAMSM) loss is computed to measure the similarity between the generated images and the corresponding text descriptions. With the attention technique, AttnGAN can generate fine-grained details in the generated images based on the related words in the text description. This allows for a more accurate and visually pleasing representation of the input text, resulting in higher quality synthesized images. The overall architecture of AttnGAN is presented in Figure 6.

The Multi-Modal Vector Representation (MMVR) presented by Sah et al. [40] is a two-way image and text generation approach that aims to generate images based on textual descriptions and vice versa. MMVR comprises an image generator that generates images from random noise, and a caption generator that generates the desired text description based on the generated image. One unique feature of MMVR is that the caption generator is used to update the image generator to generate images that are more relevant to the original text description. Additionally, the n-gram cost function is used to improve the generalization capability of the network in the text feature. To further improve image quality, MMVR leverages multiple sentence conditioning by involving several text descriptions with similar meanings. This helps the model generate images that are consistent with different text descriptions that share a similar semantic meaning.

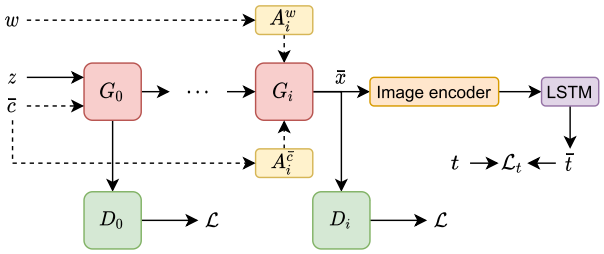



**FIGURE 7.** The overall architecture of MirrorGAN. $\bar{t}$ denotes the generated text description. $\mathcal{L}_t$ denotes the recaption loss.

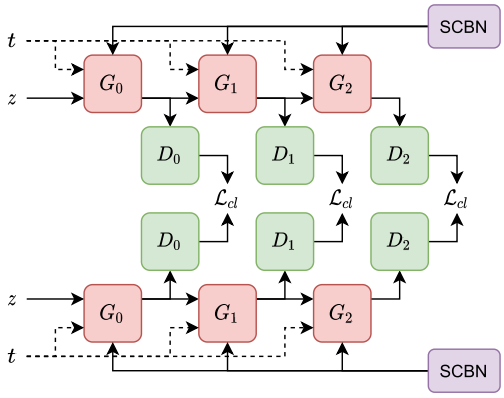



**FIGURE 8.** The overall architecture of SD-GAN. $\mathcal{L}_{cl}$ denotes the contrastive loss used to minimize the distance between the features to explore semantic commons. The rest of the loss computation is similar to AttnGAN.

Similarly, Qiao et al. [41] proposed MirrorGAN, which is a modified version of AttnGAN that aims to improve the semantic consistency between the generated image and the input text. MirrorGAN generates an image from the input text and then reproduces a text description based on the generated image using a text-regeneration module. The semantic consistency between the input text and the generated text is measured to ensure that the generated image is semantically consistent with the input text. The overall architecture of MirrorGAN is depicted in Figure 7.

Generating images from different text descriptions that express the same image is a significant challenge in text-to-image generation. The Semantics Disentangling Generative Adversarial Network (SD-GAN) proposed by Yin et al. [42] addresses this challenge by capturing the semantic commons between different text descriptions to ensure consistency in the image generation process, while retaining the diversity and detail of the text descriptions. SD-GAN uses a Siamese mechanism to capture the semantic commons by training the discriminator to determine the similarity between the generated image for different descriptions with the same original image and different original images. In each Siamese branch, the generator is built using a multi-stage GAN architecture with a Semantic-Conditioned Batch Normalization (SCBN) layer that embeds the semantic details and diversity into the visual representation. The overall architecture is illustrated in Figure 8.

Tan et al. [43] introduced Semantics-enhanced GAN (SE-GAN) that also utilizes the Siamese network architecture. Instead of using a single image and text description, SE-GAN takes two images with different text descriptions as input to the Siamese network. The objective of the Siamese network is to maximize the similarity between the original and generated images while minimizing the similarity between images of different text descriptions to ensure semantic consistency. Previous text-to-image synthesis models like StackGAN, StackGAN++, GAN-INT-CLS, and HD-GAN used the whole text description to synthesize the image, while AttnGAN utilized attention mechanisms to focus on individual words at both the sentence and word level. Unlike AttnGAN and Obj-GAN, which use object-driven attention mechanisms to focus on the object, SE-GAN only considers important words and ignores the others, thereby improving the accuracy and stability of the generation process.

To enhance the semantic consistency during text-to-image synthesis, Wang et al. [44] devised Textual-Visual Bidirectional Generative Adversarial Network (TVBi-GAN). This model utilizes several semantic related modules to enhance semantic consistency during the image synthesis process. It is built on the BiGAN architecture that consists of three main components: generator, discriminator, and encoder. The encoder maps the real image into latent data, while the discriminator identifies whether the received data is from the encoder or synthesized.

To improve the quality of the generated images further, Cheng et al. [45] proposed the Rich Feature generation text-to-image synthesis (RiFeGAN) model. This model retrieves multiple related text descriptions and utilizes the text features to enrich the input vector used for image synthesis. The model is based on the AttnGAN architecture, which allows the model to focus on different regions of the image while generating it.

In contrast to the multi-stage architecture that has been widely used, Tao et al. [46] introduced a Deep Fusion Generative Adversarial Networks (DF-GAN) that uses a one-stage text-to-image backbone to synthesize high-resolution images directly. The authors also introduced a novel Target-Aware Discriminator that consists of a Matching-Aware Gradient Penalty (MAGP) and One-Way Output. This approach improves text-image semantic consistency without introducing new networks. Moreover, they presented a revolutionary Deep text-image Fusion Block (DFBlock), which deeply and effectively fuses text features into every layer of the generator.

On the other hand, Zhang et al. [47] proposed a Cross-Modal Contrastive Generative Adversarial Network (XMC-GAN) that maximizes the mutual information between image and text descriptions to handle text-to-image synthesis. This is achieved through various contrastive losses that capture inter-modality and intra-modality correspondences. They proposed an attentional self-modulation generator with a one-stage backbone to produce a strong text-image correlation during the synthesizing process. Additionally, they

**TABLE 3. Summary of text-to-image synthesis approaches in the semantic enhancement category.**

| Method | Description |
|---|---|
| AttnGAN [18] | Attentional multi-stage GAN architecture |
| MMVR [40] | Image and caption generators |
| MirrorGAN [41] | Text-regeneration module |
| SD-GAN [42] | Siamese network mechanism |
| SE-GAN [43] | Siamese network with important word attention |
| TVBi-GAN [44] | Several semantic related modules with BiGAN |
| RiFeGAN [45] | Additional knowledge-based module |
| DF-GAN [46] | One-stage generator backbone with a Target-Aware discriminator |
| XMC-GAN [47] | Incorporate various contrastive loss in the synthesizing process |
| T2IGAN [48] | Investigating NAS to search for a suitable backbone design with a lightweight transformer |
| PBGN [49] | Investigating the bidirectional generative mechanism to improve the semantic consistency |

**TABLE 4. Summary of text-to-image synthesis approaches in the style transfer category.**

| Method | Description |
|---|---|
| [50] | Modified object based on the text description |
| MC-GAN [51] | Generate specific objects in selected background |
| Control-GAN [52] | Control part of the image region based on the text description |


**FIGURE 9. Some examples of the Oxford-102 dataset.**

introduced a contrastive discriminator that serves as both a critic and a feature encoder for contrastive learning. The contrastive learning is carried out on the discriminator side with three aspects: (1) between image and sentence, (2) between real image and fake image, and (3) between image region and words.

Although prior works [11], [18] have achieved good results in text-to-image synthesis, determining the model architectures are critical to ensure optimal performance for the increasingly complex task. In response, Li et al. [48] proposed a novel approach named T2IGAN, which utilized neural architecture search (NAS) and a lightweight transformer to effectively integrate the text and vision feature spaces. This approach outperformed existing methods in terms of both quantitative and qualitative evaluation.

Zhu et al. [49] also tackled the challenge of improving semantic consistency during image generation with their proposed Phased Bidirectional Generative Network (PBGN). PBGN uses a bidirectional generative mechanism based on a multi-level generative adversarial network to generate images that are constrained by a reconstruction loss to be similar to their corresponding text descriptions. They also explored the use of the self-attention mechanism and spectrum normalization approaches to improve the performance of generative networks. Table 3 summarizes the existing text-to-image synthesis with semantic enhancement approaches.

## VI. STYLE TRANSFER APPROACHES

Style transfer approaches aim to modify an image in a way that it takes on the style of another image or text while preserving the content of the original image. Dong et al. [50] proposed a multi-modal condition GAN for style transfer problems that takes both image and text as input and modifies the image to match the target image based on the text description. Another approach is the multi-conditional GAN (MC-GAN) presented by Park et al. [51]. This approach generates the object described in the text and places it in the background of the original image. This allows for more flexibility in generating objects that are not constrained to similar objects in the base image.

However, changing the visual content based solely on the text description may lead to inconsistent results. Li et al. [52] introduced the Controllable Text-to-Image GAN (Control-GAN) to address this issue. This approach allows for more control over the generated image by using a multi-stage architecture and attention techniques to generate image parts based on highly relevant words. Additionally, the perceptual loss is used to reduce randomness in the image synthesis process, resulting in higher quality images. The summary of the existing text-to-image synthesis by style transfer approaches is presented in Table 4.

## VII. DATASETS

This section provides an overview of the benchmark image datasets that are commonly used to evaluate the performance of text-to-image synthesis approaches. These datasets are crucial for assessing the ability of these models to generate realistic images that are consistent with the accompanying text descriptions. The three most widely used benchmark datasets in text-to-image synthesis are Oxford-102, CUB-200-2011, and COCO.

### A. OXFORD-FLOWER-102

The Oxford-Flower-102 (Oxford-102) dataset was originally collected by Nilsback et al. [53] and contains 8189 images of 102 different flower categories. To facilitate training and testing, the dataset has been divided into 82 training classes and 20 testing classes in prior research. Each image is paired with 10 captions during both training and testing. Figure 9 provides some sample images from the Oxford-102 dataset.

### B. CALTECH-UCSD BIRD

The Caltech-UCSD Bird (CUB-200-2011) dataset was introduced by Welinder et al. [54] and included 11,788 images

**TABLE 5.** The summary of the used benchmark dataset in text-to-image synthesis.

| Dataset | Number of Classes | Number of Captions per Image | Number of Training Images | Number of Testing Images | Number of Total Images |
|---|---|---|---|---|---|
| Oxford-102 [53] | 102 | 10 | 7,034 | 1,155 | 8,189 |
| CUB-200-2011 [54] | 200 | 10 | 8,855 | 2,933 | 11,788 |
| COCO [55] | 80 | 5 | 82,783 | 40,504 | 123,287 |

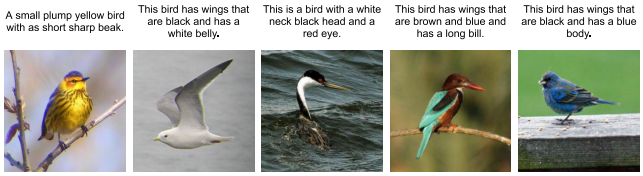



**FIGURE 10.** Some examples of the CUB-200-2011 dataset.


**FIGURE 11.** Some examples of the COCO dataset.

of 200 different bird species. Similar to other benchmark datasets, CUB-200-2011 is typically divided into 150 classes for training and 50 classes for testing in previous studies. Each image is accompanied by 10 captions for text-based analysis. To ensure a high-quality and consistent dataset, images in the CUB-200-2011 are preprocessed before training to maintain a ratio of object to image region greater than 75%. Some sample images from the CUB-200-2011 dataset are shown in Figure 10.

### C. COMMON OBJECTS IN CONTEXT

The Common Object in Context (COCO) [55] includes 80 different object classes and is primarily used for object detection and recognition. Most existing text-to-image synthesis approaches use the 2014 version of COCO. Unlike the previously discussed datasets, which typically include one object per image, many images in COCO contain multiple objects within a single scene, which makes the text-to-image synthesis task more challenging. Examples of images from the COCO dataset are shown in Figure 11.

At last, Table 5 provides a summary of the key statistics for each of these datasets.

## VIII. PERFORMANCE EVALUATION

Since text-to-image synthesis is a challenging task, evaluating the performance of these approaches is crucial. Two primary methods are commonly used for performance evaluation, which are quantitative and qualitative measurements [56]. The following subsections describe the quantitative and qualitative metrics used in text-to-image synthesis evaluation.

### A. QUANTITATIVE MEASUREMENT

Quantitative measurement involves using evaluation metrics to calculate numerical scores based on a set of images, which summarize the synthesized image quality.

#### 1) INCEPTION SCORE

The Inception Score is a commonly used metric in evaluating the performance of GANs, and it measures two aspects of image quality: object distinctness and variety. The Inception v3 classification model [57] is used to obtain the score, and it measures the quality of the set of images by analyzing the label probability distribution for each image. The more distinct the object detected in the image, the higher the score. Additionally, the number of different object varieties detected in the image set is also considered. This is measured by combining the label probability distributions from all images to create the marginal distribution. If the scores of each class in the marginal distribution are equally high, the object variety in the image set is diverse enough.

The Inception Score is computed using the Kullback-Leibler divergence between the label probability distribution and the marginal distribution. The score is calculated using the formula:

$$I = \exp\left(\mathbb{E}_x D_{KL}\left(p\left(y \mid x\right) \parallel p\left(y\right)\right)\right) \tag{4}$$

where $p\left(y \mid x\right)$ denotes label probability distribution, $y$ represents the set of predicted labels and $x$ denotes the images synthesized by the target model. $p\left(y\right)$ denotes marginal distribution by combining the predicted labels $y$. The better the performance of the target model, the larger the KL divergence between the distribution of $p\left(y\right)$ and $p\left(y \mid x\right)$. Therefore, a higher score is better for Inception Score. For a fair comparison with other existing methods, the fine-tuned Inception v3 model [10] is leveraged to compute the Inception Score. The Inception Score results of existing approaches are presented in Table 6.

#### 2) FRÉCHET INCEPTION DISTANCE

The Fréchet Inception Distance (FID) is a commonly used metric to evaluate the performance of GANs in generating realistic and diverse images. Like the Inception Score, FID also uses the pre-trained Inception v3 model to measure the similarity between the distribution of the generated images and the real images in the feature space. Unlike Inception Score which uses the fine-tuned model, the Inception v3 model used by FID is only pre-trained on ImageNet without any fine-tuning.

**TABLE 6.** The Inception Score results of the existing works.

| Model | Oxford-102 | CUB-200-2011 | COCO |
|---|---|---|---|
| GAN-INT-CLS [6] | 2.66 | 2.88 | 7.88 |
| GAWWN [22] | - | 3.62 | - |
| PPGN [23] | - | - | 9.58 |
| MMVR [40] | - | - | 8.30 |
| StackGAN [10] | 3.20 | 3.70 | 8.45 |
| StackGAN++ [11] | 3.26 | 4.04 | 8.30 |
| TAC-GAN [16] | 3.45 | - | - |
| Text-SeGAN [25] | 4.03 | - | - |
| HD-GAN [26] | 3.45 | 4.15 | 11.86 |
| AttnGAN [18] | - | 4.36 | 25.89 |
| DM-GAN [27] | - | 4.75 | 30.49 |
| PPAN [28] | 3.52 | 4.38 | - |
| HfGAN [29] | 3.57 | 4.48 | 27.53 |
| MRP-GAN [31] | - | 4.77 | 31.10 |
| SSTIS [13] | 3.41 | 3.93 | - |
| SSBi-GAN [14] | 3.44 | 4.13 | - |
| SS-TiGAN [15] | 3.45 | 4.09 | - |
| [32] | - | 4.74 | 30.57 |
| [30] | 3.71 | 4.23 | - |
| Semantic layout [34] | - | - | 11.46 |
| AttnGAN+OP [35] | - | - | 24.76 |
| Obj-GAN [36] | - | - | 27.37 |
| Chatpainter [38] | - | - | 9.74 |
| OP-GAN [37] | - | - | 27.88 |
| [39] | - | 5.06 | 29.03 |
| MirrorGAN [41] | - | 4.56 | 26.47 |
| SD-GAN [42] | - | 4.67 | 35.69 |
| SE-GAN [43] | - | 4.67 | 27.86 |
| TVBi-GAN [44] | - | 5.03 | 31.01 |
| RiFeGAN [45] | 4.53 | 5.23 | 31.70 |
| Control-GAN [52] | - | 4.58 | 24.06 |
| DF-GAN [46] | - | 5.10 | - |
| XMC-GAN [47] | - | - | 30.45 |
| T2IGAN [48] | 4.89 | 5.12 | 31.93 |
| PBGN [49] | 4.59 | 5.13 | 32.42 |
| PBGN-EE [49] | 4.71 | 5.23 | 26.84 |

The FID is computed in the feature space rather than the image space. In other words, instead of using the images themselves, FID uses the features of the images to quantify the quality of the generated images. Specifically, the images are embedded by the last average pooling layer of the Inception v3 model to produce 2048-dimensional feature vectors. The mean and covariance of the feature sets are then estimated for computing the FID score.

The Fréchet distance is a measure of similarity between two multivariate Gaussian distributions. The FID score measures the distance between the distributions of the real image set and the generated image set in the feature space. The FID score is computed as:

$$\begin{aligned} \mathrm{FID}\left(v, \bar{v}\right) &= \left\|\mu_v - \mu_{\bar{v}}\right\|^2 \\ &\quad + \mathrm{T}_r\left(\Sigma_v + \Sigma_{\bar{v}} - 2\left(\Sigma_v \Sigma_{\bar{v}}\right)^{\frac{1}{2}}\right) \end{aligned} \tag{5}$$

where $v$ and $\bar{v}$ denote the real and fake image features, respectively, $\mu$ and $\Sigma$ denote the mean and covariance of $v$ and $\bar{v}$, and $\mathrm{T}_r$ denotes the trace of the matrix. The smaller the FID score, the closer the distributions of real and fake images in the feature space, and hence the better the generated images.

It is worth noting that the FID score may vary depending on the machine learning libraries used in the implementation of the existing works, such as TensorFlow and PyTorch.

**TABLE 7.** The FID results of the existing works.

| Model | Oxford-102 | CUB-200-2011 | COCO |
|---|---|---|---|
| GAN-INT-CLS [6] | 79.55 | 68.79 | 60.62 |
| GAWWN [22] | - | 67.22 | - |
| PPGN [23]† | - | - | 43.77 |
| StackGAN [10] | 55.28 | 51.89 | 74.05 |
| StackGAN++ [11] | 48.68 | 15.30 | 81.59 |
| AttnGAN [18]† | - | 23.19 | 34.97 |
| AttnGAN [18]* | - | 14.01 | 28.76 |
| DM-GAN [27]† | - | 15.34 | 25.22 |
| DM-GAN [27]* | - | 11.91 | 24.24 |
| MRP-GAN [31] | - | 19.25 | 23.75 |
| SSTIS [13]† | 42.95 | 15.80 | - |
| SSBi-GAN [14]† | 41.13 | 14.07 | - |
| SS-TiGAN [15]† | 40.54 | 14.20 | - |
| [32] | - | 9.63 | 27.83 |
| EruditeGAN [33] | 17.69 | 9.58 | - |
| [30] | 16.47 | 11.17 | - |
| AttnGAN+OP [35] | - | - | 33.35 |
| Obj-GAN [36]† | - | - | 21.21 |
| Obj-GAN [36]* | - | - | 17.04 |
| OP-GAN [37] | - | - | 24.70 |
| [39]† | - | 16.87 | 20.06 |
| [39]* | - | 12.34 | 16.28 |
| MirrorGAN [41] | - | 26.80 | 37.86 |
| SE-GAN [43] | - | 18.17 | 32.28 |
| TVBi-GAN [44] | - | 11.83 | 31.97 |
| DF-GAN [46] | - | 14.81 | 19.32 |
| XMC-GAN [47] | - | - | 9.33 |
| T2IGAN [48] | 13.55 | 10.48 | 26.45 |

† indicates using PyTorch implementation. * indicates using TensorFlow implementation.

Therefore, it is important to ensure that the same library is used to compare the FID scores across different models. The FID results of existing works are presented in Table 7.

### 3) STRUCTURAL SIMILARITY INDEX

While metrics like Inception Score and FID can evaluate image realism, they do not take into account the semantic consistency between the real and generated images. The Structural Similarity Index (SSIM) is widely used in the image generation domain to measure the similarity between two images. In text-to-image synthesis, the real and generated images should have high SSIM since they have similar visual contents conditioned on the same text description. SSIM measure the corresponding pixels and their neighborhoods in two images $x$ and $y$ with three aspects: luminance $I$, contrast $C$, and structure $S$ as below:

$$\mathrm{SSIM}\left(x, y\right) = I\left(x, y\right)^{\alpha} C\left(x, y\right)^{\beta} S\left(x, y\right)^{\gamma} \tag{6}$$

where $\alpha$, $\beta$, and $\gamma$ are the coefficients for each component. The comparison functions $I$, $C$, and $S$ are computed as follows:

$$\begin{aligned} I\left(x, y\right) &= \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1} \\ C\left(x, y\right) &= \frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2} \\ S\left(x, y\right) &= \frac{\sigma_{xy} + C_3}{\sigma_x\sigma_y + C_3} \end{aligned} \tag{7}$$

**TABLE 8.** The SSIM results of the existing works.

| Model | Oxford-102 | CUB-200-2011 |
|---|---|---|
| GAN-INT-CLS [6] | 0.1948 | 0.2934 |
| GAWWN [22] | - | 0.2370 |
| StackGAN [10] | 0.1837 | 0.2812 |
| HD-GAN [26] | 0.1886 | 0.2887 |
| AttnGAN [18] | 0.1873 | 0.3129 |
| SSTIS [13] | 0.7290 | 0.7982 |
| SSBi-GAN [14] | 0.7394 | 0.8144 |
| SS-TiGAN [15] | 0.7353 | 0.8195 |

where $C_1$, $C_2$, and $C_3$ are constant values added to the equation to prevent the denominator from becoming zero. Luminance $I$ measures the average value of the images, and contrast $C$ measures the square root of all pixel values. Thus, $\mu$ and $\sigma$ denote the mean and standard deviation of image $x$ or $y$, which are computed as:

$$\mu_x = \frac{1}{N}\sum_{i=1}^{N} x_i \qquad \sigma_x = \left(\frac{1}{N-1}\sum_{i=1}^{N}(x_i - \mu_x)^2\right)^{\frac{1}{2}} \tag{8}$$

where $N$ denotes the total number of pixels in image $x$. Structure $S$ is measured by the sample correlation coefficient between corresponding pixels $\sigma_{x,y}$ that are centered in $x$ and $y$, which is defined as:

$$\sigma_{xy} = \frac{1}{N-1}\sum_{i=1}^{N}(x_i - \mu_x)\left(y_i - \mu_y\right) \tag{9}$$

The real and generated images with the same text description are evaluated as a pair in SSIM, and the average score of all the pairs is taken as the final result. A higher score indicates better semantic consistency between the real and generated images. SSIM has not been widely adopted by the text-to-image research community. Nonetheless, some studies [13], [58], [59] have reported their SSIM scores, which are presented in Table 8.

#### 4) R-PRECISION

R-precision [18] is a metric used to evaluate the visual-semantic similarity between a generated image and its original text description. It takes into account a set of descriptions and compares the similarity of the generated image with each of them.

Given an original text description $t$, R-precision selects $t$ along with 99 randomly selected non-related descriptions to form a description set $S = \{t_1, t_2, \ldots, t_{100}\}$. For each description $t_i$ in $S$, the cosine distance between the generated image feature vector $f_g$ and the corresponding description feature vector $f_i$ is computed. The feature vectors can be obtained using the pre-trained image and text models such as VGG-19 and BERT.

The resulting set of distances is sorted in ascending order to obtain a ranked list of the descriptions in $S$. R-precision is then computed as the fraction of cases where the original description $t$ appears in the top $k$ ranked descriptions in $S$, where $k$ can be 1, 2, or 3. The R-precision score is the average of the top-$k$ hits over a set of images. The R-precision score is defined as:

$$\text{R-precision} = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{k}\sum_{j=1}^{k}[\text{rank}(t_i, t) \leq j] \tag{10}$$

where $N$ is the total number of images, $t_i$ is the original text description for the $i$-th image, $k$ is the number of top ranked descriptions to consider, $\text{rank}(t_i, t)$ is the position of the original description $t$ in the ranked list of descriptions for the $i$-th image, and $[\cdot]$ is the Iverson bracket that returns 1 if the condition is true and 0 otherwise. The R-precision results of existing works are presented in Table 9.

**TABLE 9.** The R-precision results of the existing works.

| Model | Oxford-102 | CUB-200-2011 | COCO |
|---|---|---|---|
| AttnGAN [18] | - | 67.82 | 85.47 |
| MirrorGAN [41] | - | 57.67 | 74.52 |
| Control-GAN [52] | - | 69.33 | 82.43 |
| DM-GAN [27] | - | 72.31 | 88.56 |
| AttnGAN+OP [35] | - | - | 82.44 |
| OP-GAN [37] | - | - | 89.01 |
| Obj-GAN [36] | - | - | 91.05 |
| MRP-GAN [31] | - | 74.84 | 89.30 |
| [32] | - | 77.49 | 88.58 |
| EruditeGAN [33] | 80.25 | 77.62 | - |
| XMC-GAN [47] | - | - | 71.00 |
| PBGN [49] | 91.58 | 87.72 | 92.29 |
| PBGN-EE [49] | 93.56 | 88.33 | 89.32 |

#### 5) VISUAL-SEMANTIC SIMILARITY

Visual-semantic (VS) similarity [26] measures the semantic consistency between the generated image and its corresponding text description. The alignment between the image and text description is measured by a visual-semantic embedding model, which contains two mapping functions: a text encoder and an image encoder, that map the input into a common feature space. The similarity is computed using the dot product of the encoded text and image features and normalized by their $l_2$-norms, as follows:

$$VS = \frac{f_t(t) \cdot f_x(x)}{\|f_t(t)\|_2 \cdot \|f_x(x)\|_2} \tag{11}$$

where $f_t$ denotes the text encoder, $f_x$ denotes the image encoder, $t$ denotes the input text description, and $x$ denotes the synthesized image.

Although VS similarity is a useful metric for measuring semantic consistency, it has not been widely adopted by researchers due to the high standard deviation of the results and the variation in results when using different pre-trained models. Nonetheless, the results of VS similarity for existing works are reported in Table 10, where a higher score indicates better semantic consistency between the generated image and the input text description.

**TABLE 10.** The VS similarity results of the existing works.

| Model | Oxford-102 | CUB-200-2011 | COCO |
|---|---|---|---|
| GAN-INT-CLS [6] | - | 8.2 | - |
| GAWWN [22] | - | 11.4 | - |
| StackGAN [10] | 27.8 | 22.8 | - |
| HD-GAN [26] | 29.6 | 24.6 | 19.9 |
| HfGAN [29] | 30.3 | 25.3 | 22.7 |
| PPAN [28] | 29.7 | 29.0 | - |
| AttnGAN* [18] | - | 22.5 | 7.1 |
| SE-GAN* [43] | - | 30.2 | 8.9 |

* model evaluated with different pre-trained model.

**TABLE 11.** The summary of the SOA results on the existing works on the COCO dataset.

| Model | SOA-C | SOA-I |
|---|---|---|
| AttnGAN [18] | 25.88 | 39.01 |
| DM-GAN [27] | 33.44 | 48.03 |
| AttnGAN+OP [35] | 25.46 | 40.48 |
| OP-GAN [37] | 35.85 | 50.47 |
| Obj-GAN [36] | 27.14 | 41.24 |
| XMC-GAN [47] | 50.94 | 71.33 |

#### 6) SEMANTIC OBJECT ACCURACY

Semantic Object Accuracy (SOA) was introduced by [37] to measure semantic consistency for complex scenes that consist of multiple objects. SOA explicitly evaluates each object with the text description to measure performance. The metric uses a pre-trained object detector model, specifically a YOLOv3 network [60] that is trained on a COCO dataset to check whether the object mentioned in the text description appears in the generated image.

SOA can be calculated using two different methods, class average (SOA-C) and image average (SOA-I). SOA-C is computed by taking the average of the SOA score for each class of objects present in the image, while SOA-I is computed by taking the average of the SOA scores for each generated image in the dataset. The SOA score is defined as:

$$SOA = \frac{\text{no. of correctly detected objects}}{\text{no. of objects mentioned in text description}} \quad (12)$$

The results of existing works using the SOA metric are presented in Table 11. A higher SOA score indicates better semantic consistency between the generated image and the text description, and therefore the better performance of the model.

#### 7) HUMAN EVALUATION

Human evaluation is an important aspect of evaluating the performance of text-to-image synthesis models, as it provides a measure of the model's ability to generate images that are visually and semantically consistent with the input text descriptions. Human evaluation involves presenting a set of synthesized images and their corresponding text descriptions to human evaluators, who are then asked to provide a rating or ranking of the images based on their quality and consistency with the text description.

Several factors can affect the results of human evaluation, such as the number of samples, the number of users, the instructions given to the users, and the experiment time limitations. Therefore, there is currently no standardized protocol for conducting the human evaluation of text-to-image synthesis models. Human evaluation can be time-consuming and resource-intensive, which makes it challenging to scale up the evaluation process. Hence, it is mostly used in conjunction with machine-based metrics to provide a more complete and reliable evaluation [34], [37], [42], [61].

### B. QUALITATIVE MEASUREMENT

Qualitative measurement is a subjective evaluation that involves human judgment based on the visual inspection of synthesized images. It is often used to compare the image quality and semantic consistency of the generated images across different models. Additionally, it can be used to identify mode collapse in the model training, which occurs when the model generates similar or identical images for different text inputs.

To perform qualitative evaluation, researchers often select a few synthesized images from different models and display them side-by-side for comparison. The images are evaluated based on their visual appearance and how well they match the corresponding text description. This type of evaluation is subjective, as different individuals may have different opinions on the quality and consistency of the images. Figure 12, Figure 13, and Figure 14 demonstrate the qualitative measurement performed by the existing works on Oxford-102, CUB-200-2011, and COCO datasets.

## IX. LIMITATIONS AND FUTURE RESEARCH PROSPECTS

Text-to-image synthesis is a challenging research area with several limitations and future research prospects.

### A. LIMITATIONS

One of the main limitations of the existing text-to-image synthesis approaches is that they rely heavily on large-scale datasets for training. These datasets are often expensive and time-consuming to create, and they might not represent the diversity of real-world scenes.

Moreover, the generated images might suffer from mode collapse, where the model generates a limited set of images that are highly similar to each other, leading to a lack of diversity in the generated images. Additionally, current models have difficulties in generating images with fine-grained details, such as textures and shapes, especially for complex scenes with multiple objects.

Another limitation of existing text-to-image synthesis approaches is that they do not incorporate any explicit reasoning mechanisms. This means that the models cannot reason about the relationships between objects, such as their spatial and semantic relationships, which often leads to inconsistencies in the synthesized images.


**FIGURE 12.** Example of images synthesized based on the Oxford-102 dataset (Adopted from PPAN [28]). GT denotes real images.

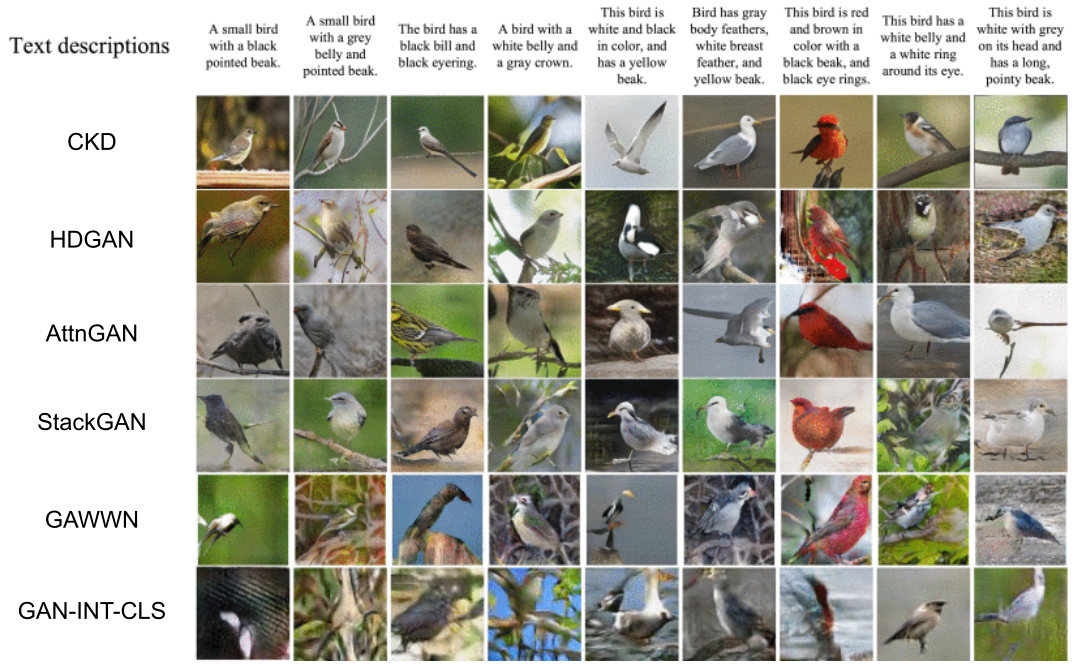



**FIGURE 13.** Example of images synthesised based on the CUB-200-2011 dataset (Adopted from CKD [59]).

### *B. FUTURE RESEARCH PROSPECTS*

There are several promising research directions for text-to-image synthesis. One of them is integrating text-to-image synthesis with other tasks, such as image editing and manipulation, which can lead to more versatile and flexible image generation systems. This can be achieved through approaches such as style transfer and image-to-image translation, where the models can learn to transfer the style and content of an image to another domain. Combining these approaches with text-to-image synthesis can reduce the requirements for large-scale training datasets.

Another promising direction is improving the diversity of generated images by addressing the issue of mode collapse through techniques such as diversity-promoting regularization. This can lead to more diverse and realistic images, reducing the repetition of certain image features. Moreover, incorporating explicit reasoning mechanisms into the models, such as spatial and semantic reasoning, can generate more consistent and coherent images. This approach can allow the models to better understand the relationships between different objects in a scene and generate images that are more visually plausible.


**FIGURE 14.** Example of images synthesised based on the COCO dataset (Adopted from OP-GAN [37]).

Besides that, integrating semantic reasoning is an interesting direction for text-to-image synthesis. This capability empowers the model to generate images that faithfully represent the intended scenes, resulting in heightened consistency, reduced ambiguity, and enriched creativity. By comprehending implied details, contextual nuances, and relationships, semantic reasoning ensures that the generated images are not only semantically aligned with the text but also possess fine-grained visual attributes, coherent styles, and a sensitivity to the varying levels of detail in different textual inputs, ultimately elevating the quality, realism, and relevance of the synthesized images. These future research directions can greatly improve the capabilities and quality of text-to-image synthesis models.

## X. CONCLUSION

In summary, this paper provides an overview of the current state-of-the-art GAN-based text-to-image synthesis approaches. By categorizing the approaches into four main categories, the paper gives a clear understanding of the different goals and challenges faced in this field. The benchmark datasets discussed in this paper are useful for researchers to compare their models with existing works. Additionally, the evaluation metrics presented in this paper provide insights into the strengths and weaknesses of the models. It is important to note that each of the metrics has its own advantages and limitations, and it is recommended to use multiple metrics to evaluate the model performance comprehensively. Text-to-image synthesis is a promising research area with several limitations and future research prospects. Overcoming these limitations and exploring these prospects can lead to more robust and versatile text-to-image synthesis models with a wide range of applications in fields such as computer vision, graphics, and design.

## REFERENCES

[1] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Proc. 27th Int. Conf. Neural Inf. Process. Syst.*, vol. 2, 2014, pp. 2672–2680.

[2] M. Mirza and S. Osindero, "Conditional generative adversarial nets," 2014, *arXiv:1411.1784*.

[3] S. Frolov, T. Hinz, F. Raue, J. Hees, and A. Dengel, "Adversarial text-to-image synthesis: A review," *Neural Netw.*, vol. 144, pp. 187–209, Dec. 2021.

[4] R. Zhou, C. Jiang, and Q. Xu, "A survey on generative adversarial network-based text-to-image synthesis," *Neurocomputing*, vol. 451, pp. 316–336, Sep. 2021.

[5] J. Agnese, J. Herrera, H. Tao, and X. Zhu, "A survey and taxonomy of adversarial neural networks for text-to-image synthesis," *Wiley Interdiscipl. Rev., Data Mining Knowl. Discovery*, vol. 10, no. 4, p. e1345, 2020.

[6] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee, "Generative adversarial text-to-image synthesis," in *Proc. 33rd Int. Conf. Mach. Learn.*, 2016, pp. 1060–1069.

[7] S. Reed, Z. Akata, H. Lee, and B. Schiele, "Learning deep representations of fine-grained visual descriptions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 49–58.

[8] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proc. 26th Int. Conf. Neural Inf. Process. Syst.*, vol. 2. Red Hook, NY, USA: Curran Associates, 2013, pp. 3111–3119.

[9] Z. S. Harris, "Distributional structure," *Word*, vol. 10, nos. 2–3, pp. 146–162, 1954.

[10] H. Zhang, T. Xu, H. Li, S. Zhang, X. Wang, X. Huang, and D. Metaxas, "StackGAN: Text to photo-realistic image synthesis with stacked generative adversarial networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 5908–5916.

[11] 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," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 8, pp. 1947–1962, Aug. 2019.

[12] Y. X. Tan, C. P. Lee, M. Neo, K. M. Lim, and J. Y. Lim, "Enhanced text-to-image synthesis conditional generative adversarial networks," *IAENG Int. J. Comput. Sci.*, vol. 49, no. 1, pp. 149–155, 2022.

[13] Y. X. Tan, C. P. Lee, M. Neo, and K. M. Lim, "Text-to-image synthesis with self-supervised learning," *Pattern Recognit. Lett.*, vol. 157, pp. 119–126, May 2022.

[14] Y. X. Tan, C. P. Lee, M. Neo, K. M. Lim, and J. Y. Lim, "Text-to-image synthesis with self-supervised bi-stage generative adversarial network," *Pattern Recognit. Lett.*, vol. 169, pp. 43–49, May 2023.

[15] Y. X. Tan, C. P. Lee, M. Neo, K. M. Lim, and J. Y. Lim, "Enhanced text-to-image synthesis with self-supervision," *IEEE Access*, vol. 11, pp. 39508–39519, 2023.
[16] A. Dash, J. C. B. Gamboa, S. Ahmed, M. Liwicki, and M. Z. Afzal, "TAC-GAN-text conditioned auxiliary classifier generative adversarial network," 2017, *arXiv:1703.06412*.
[17] R. Kiros, Y. Zhu, R. Salakhutdinov, R. S. Zemel, A. Torralba, R. Urtasun, and S. Fidler, "Skip-thought vectors," in *Proc. NIPS*, vol. 2. Cambridge, MA, USA: MIT Press, 2015, pp. 3294–3302.
[18] T. Xu, P. Zhang, Q. Huang, H. Zhang, Z. Gan, X. Huang, and X. He, "AttnGAN: Fine-grained text to image generation with attentional generative adversarial networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1316–1324.
[19] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Human Lang. Technol.*, J. Burstein, C. Doran, and T. Solorio, Eds., Minneapolis, MN, USA: Association for Computational Linguistics, 2019, pp. 4171–4186.
[20] T. Wang, T. Zhang, and B. Lovell, "Faces a la carte: Text-to-face generation via attribute disentanglement," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2021, pp. 3379–3387.
[21] D. Pavllo, A. Lucchi, and T. Hofmann, "Controlling style and semantics in weakly-supervised image generation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2020, pp. 482–499.
[22] S. E. Reed, Z. Akata, S. Mohan, S. Tenka, B. Schiele, and H. Lee, "Learning what and where to draw," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 217–225.
[23] A. Nguyen, J. Clune, Y. Bengio, A. Dosovitskiy, and J. Yosinski, "Plug & play generative networks: Conditional iterative generation of images in latent space," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3510–3520.
[24] A. Odena, C. Olah, and J. Shlens, "Conditional image synthesis with auxiliary classifier GANs," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 2642–2651.
[25] M. Cha, Y. L. Gwon, and H. Kung, "Adversarial learning of semantic relevance in text to image synthesis," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, 2019, pp. 3272–3279.
[26] Z. Zhang, Y. Xie, and L. Yang, "Photographic text-to-image synthesis with a hierarchically-nested adversarial network," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6199–6208.
[27] M. Zhu, P. Pan, W. Chen, and Y. Yang, "DM-GAN: Dynamic memory generative adversarial networks for text-to-image synthesis," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 5795–5803.
[28] L. Gao, D. Chen, J. Song, X. Xu, D. Zhang, and H. T. Shen, "Perceptual pyramid adversarial networks for text-to-image synthesis," in *Proc. AAAI Conf. Artif. Intell.*, Jul. 2019, vol. 33, no. 1, pp. 8312–8319.
[29] X. Huang, M. Wang, and M. Gong, "Hierarchically-fused generative adversarial network for text to realistic image synthesis," in *Proc. 16th Conf. Comput. Robot Vis. (CRV)*, May 2019, pp. 73–80.
[30] D. M. Souza, J. Wehrmann, and D. D. Ruiz, "Efficient neural architecture for text-to-image synthesis," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2020, pp. 1–8.
[31] Z. Qi, C. Fan, L. Xu, X. Li, and S. Zhan, "MRP-GAN: Multi-resolution parallel generative adversarial networks for text-to-image synthesis," *Pattern Recognit. Lett.*, vol. 147, pp. 1–7, Jul. 2021.
[32] Z. Zhang, C. Fu, J. Zhou, W. Yu, and N. Jiang, "Text to image synthesis based on multi-perspective fusion," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2021, pp. 1–8.
[33] Z. Zhang, W. Yu, N. Jiang, and J. Zhou, "Text to image synthesis with erudite generative adversarial networks," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2021, pp. 2438–2442.
[34] S. Hong, D. Yang, J. Choi, and H. Lee, "Inferring semantic layout for hierarchical text-to-image synthesis," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7986–7994.
[35] T. Hinz, S. Heinrich, and S. Wermter, "Generating multiple objects at spatially distinct locations," in *Proc. Int. Conf. Learn. Represent.*, 2019, pp. 1–23.
[36] 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 *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 12166–12174.
[37] T. Hinz, S. Heinrich, and S. Wermter, "Semantic object accuracy for generative text-to-image synthesis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 3, pp. 1552–1565, Mar. 2022.
[38] S. Sharma, D. Suhubdy, V. Michalski, S. E. Kahou, and Y. Bengio, "ChatPainter: Improving text to image generation using dialogue," 2018, *arXiv:1802.08216*.
[39] M. Wang, C. Lang, L. Liang, S. Feng, T. Wang, and Y. Gao, "End-to-end text-to-image synthesis with spatial constrains," *ACM Trans. Intell. Syst. Technol.*, vol. 11, no. 4, pp. 1–19, May 2020.
[40] S. Sah, D. Peri, A. Shringi, C. Zhang, M. Dominguez, A. Savakis, and R. Ptucha, "Semantically invariant text-to-image generation," in *Proc. 25th IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2018, pp. 3783–3787.
[41] T. Qiao, J. Zhang, D. Xu, and D. Tao, "MirrorGAN: Learning text-to-image generation by redescription," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1505–1514.
[42] G. Yin, B. Liu, L. Sheng, N. Yu, X. Wang, and J. Shao, "Semantics disentangling for text-to-image generation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 2322–2331.
[43] H. Tan, X. Liu, X. Li, Y. Zhang, and B. Yin, "Semantics-enhanced adversarial nets for text-to-image synthesis," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 10500–10509.
[44] Z. Wang, Z. Quan, Z.-J. Wang, X. Hu, and Y. Chen, "Text to image synthesis with bidirectional generative adversarial network," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jul. 2020, pp. 1–6.
[45] J. Cheng, F. Wu, Y. Tian, L. Wang, and D. Tao, "RiFeGAN: Rich feature generation for text-to-image synthesis from prior knowledge," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 10908–10917.
[46] M. Tao, H. Tang, F. Wu, X. Jing, B.-K. Bao, and C. Xu, "DF-GAN: A simple and effective baseline for text-to-image synthesis," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 16494–16504.
[47] H. Zhang, J. Y. Koh, J. Baldridge, H. Lee, and Y. Yang, "Cross-modal contrastive learning for text-to-image generation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2021, pp. 833–842.
[48] W. Li, S. Wen, K. Shi, Y. Yang, and T. Huang, "Neural architecture search with a lightweight transformer for text-to-image synthesis," *IEEE Trans. Netw. Sci. Eng.*, vol. 9, no. 3, pp. 1567–1576, May 2022.
[49] J. Zhu, Z. Li, J. Wei, and H. Ma, "PBGN: Phased bidirectional generation network in text-to-image synthesis," *Neural Process. Lett.*, vol. 54, no. 6, pp. 5371–5391, Dec. 2022.
[50] H. Dong, S. Yu, C. Wu, and Y. Guo, "Semantic image synthesis via adversarial learning," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 5707–5715.
[51] H. Park, Y. Yoo, and N. Kwak, "MC-GAN: Multi-conditional generative adversarial network for image synthesis," in *Proc. Brit. MachineVision Conf. (BMVC)*, 2018, pp. 1–13.
[52] B. Li, X. Qi, T. Lukasiewicz, and P. Torr, "Controllable text-to-image generation," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, 2019.
[53] M.-E. Nilsback and A. Zisserman, "Automated flower classification over a large number of classes," in *Proc. 6th Indian Conf. Comput. Vis., Graph. Image Process.*, Dec. 2008, pp. 722–729.
[54] P. Welinder, S. Branson, T. Mita, C. Wah, F. Schroff, S. Belongie, and P. Perona, "Caltech-UCSD birds 200," California Inst. Technol., Pasadena, CA, USA, Tech. Rep., CNS-TR-2010-001, 2010.
[55] 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 *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 740–755.
[56] A. Borji, "Pros and cons of GAN evaluation measures," *Comput. Vis. Image Understand.*, vol. 179, pp. 41–65, Feb. 2019.
[57] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2818–2826.
[58] M. Yuan and Y. Peng, "Text-to-image synthesis via symmetrical distillation networks," in *Proc. 26th ACM Int. Conf. Multimedia*, New York, NY, USA, Oct. 2018, pp. 1407–1415.
[59] M. Yuan and Y. Peng, "CKD: Cross-task knowledge distillation for text-to-image synthesis," *IEEE Trans. Multimedia*, vol. 22, no. 8, pp. 1955–1968, Aug. 2020.

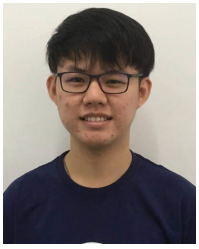
[60] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," 2018, *arXiv:1804.02767*.
[61] F. Tan, S. Feng, and V. Ordonez, "Text2Scene: Generating compositional scenes from textual descriptions," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 6703–6712.

**YONG XUAN TAN** received the B.I.T. degree (Hons.) in artificial intelligence from Multimedia University, where he is currently pursuing the M.Sc. degree in information technology. His research interests include pattern recognition and computer vision.

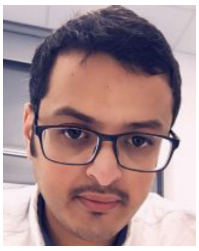
**CHIN POO LEE** (Senior Member, IEEE) received the M.Sc. and Ph.D. degrees in information technology in the area of abnormal behavior detection and gait recognition. She is a Senior Lecturer with the Faculty of Information Science and Technology, Multimedia University, Malaysia. Her research interests include action recognition, computer vision, gait recognition, and deep learning.

**MAI NEO** is a Professor with the Faculty of Creative Multimedia, specializing in e-learning and digital learning curriculum, and the former Director of the Academic Development for Excellence in Programs and Teaching (ADEPT), Multimedia University. She is the Director of the award-winning MILE Research Laboratory and the Founding Chairperson of the Centre for Adaptive Multimedia, Education and Learning cOntent Technologies (CAMELOT). Her research interests include the design of constructivist learning environments, micro-learning, team-based learning, and web-based education. She was a recipient of the 2014 Excellent Researcher Award, the AKEPT Certified Trainer for Interactive Lectures (Level 1, 2, and 3), and the HRDF Certified Trainer. She is certified in team-based learning from the Team-Based Learning Collaborative, USA. She has been a recipient of several TMRnD Grants, since 2010, and her projects have won several Gold Medals in innovation competitions (ITEX, IUCEL, and IPHEX). She is the Managing Editor of the *International Journal of Creative Multimedia* (IJCM).

**KIAN MING LIM** (Senior Member, IEEE) received the B.I.T. degree (Hons.) in information systems engineering and the M.Eng.Sc. and Ph.D. (I.T.) degrees from Multimedia University. He is currently a Lecturer with the Faculty of Information Science and Technology, Multimedia University. His research and teaching interests include machine learning, deep learning, computer vision, and pattern recognition.



**JIT YAN LIM** received the B.I.T. degree (Hons.) in artificial intelligence and the Ph.D. (I.T.) degree from Multimedia University. He is currently a Lecturer with the Faculty of Information Science and Technology, Multimedia University. His research interests include machine learning, computer vision, few-shot learning, and image generation.



**ALI ALQAHTANI** received the Ph.D. degree in computer science from Swansea University, Swansea, U.K., in 2021. He is currently an Assistant Professor with the Department of Computer Science, King Khalid University, Abha, Saudi Arabia. He has published several refereed journals and conferences. His research interests include various aspects of pattern recognition, deep learning, and machine intelligence and their applications to real-world problems.