Artificial Intelligence Review
https://doi.org/10.1007/s10462-023-10434-2

Image synthesis: a review of methods, datasets, evaluation metrics, and future outlook

Samah Saeed Baraheem[1,2] · **Trung-Nghia Le**[3,4] · **Tam V. Nguyen**[2]

Abstract
Image synthesis is a process of converting the input text, sketch, or other sources, *i.e.*, another image or mask, into an image. It is an important problem in the computer vision field, where it has attracted the research community to attempt to solve this challenge at a high level to generate photorealistic images. Different techniques and strategies have been employed to achieve this purpose. Thus, the aim of this paper is to provide a comprehensive review of various image synthesis models covering several aspects. First, the image synthesis concept is introduced. We then review different image synthesis methods divided into three categories: image generation from text, sketch, and other inputs, respectively. Each sub-category is introduced under the proper category based upon the general framework to provide a broad vision of all existing image synthesis methods. Next, brief details of the benchmarked datasets used in image synthesis are discussed along with specifying the image synthesis models that leverage them. Regarding the evaluation, we summarize the metrics used to evaluate the image synthesis models. Moreover, a detailed analysis based on the evaluation metrics of the results of the introduced image synthesis is provided. Finally, we discuss some existing challenges and suggest possible future research directions.

Keywords Image synthesis · Image generation · Generative adversarial networks · Machine learning · Computer vision

1 Introduction

Image synthesis is a means to generate artificial images from various input forms, *i.e.*, text, sketch, audio, or another image. It plays a significant role in many practical applications, *i.e.*, art generation (Elgammal et al. 2017), computer-aided design (Thaung 2020),

✉ Samah Saeed Baraheem
ssbaraheem@uqu.edu.sa

[1] Department of Computer Science, Umm Al-Qura University, Al-lith, Saudi Arabia

[2] Department of Computer Science, University of Dayton, Dayton, USA

[3] University of Science, VNU-HCM, Ho Chi Minh City, Vietnam

[4] Vietnam National University, Ho Chi Minh City, Vietnam

Published online: 28 February 2023

photo-editing (Chen et al. 2017a; Yan et al. 2014), photo inpainting (Yu et al. 2018; Nazeri et al. 2019), education (Finlayson et al. 2018), human–computer interaction (Zhao et al. 2021), and security (Chen and Jiang 2019; Adiban et al. 2020). Therefore, image synthesis has recently become a hot topic with growing interest from researchers. The broad categories of image synthesis methods are CNNs, VAEs, GANs, image retrieval, and diffusion models. This review paper discusses all these categories; however, it concentrates more on GAN-based image synthesis because there has been a rapid growth of research in GANs, especially in image synthesis domain. Since recent diffusion models are leveraged in the image generation field and demonstrate great potential for image synthesis, we discuss diffusion model-based image synthesis in the future outlook section. Convolutional Neural Network (CNN) (Springenberg et al. 2014) is a subtype of Neural Network (NN) that is commonly used in visual imagery because the convolution layer is capable of reducing the high dimensionality of images while preserving its information. Specifically, it automatically detects the most important and meaningful features without human intervention and supervision. In the meantime, Variational Autoencoder (VAE) (Kingma and Welling 2013) is a type of generative model that consists of an encoder and a decoder trained to minimize the reconstruction error between the original data and the encoded-decoded data. It encodes the input as a distribution over the latent space rather than encoding it as a single point as in the standard autoencoder. Generative Adversarial Network (GAN) (Goodfellow et al. 2014) is composed of two networks, a generator and a discriminator, which compete against each other in the minmax game. While the generator attempts to fool the discriminator by generating realistic images that look like real images, the discriminator attempts to distinguish between real and generated/fake images. Sketch-to-image synthesis started from sketch-based image retrieval, which queries a database to retrieve the most matched image to the input. During the retrieval process, a descriptor is used to extract the image features. Then, a composition technique might be incorporated to synthesize the final image if the system retrieves parts of the image. Finally, the diffusion model (Sohl-Dickstein et al. 2015) starts by slowly adding random noise to the input via forward diffusion steps. Then, it learns to reverse the diffusion process to reconstruct the input from the noise (see Sect. 5 for details).

Text-to-image synthesis (Zhu et al. 2007; Dosovitskiy et al. 2015; Yan et al. 2015; Gregor et al. 2015; Mansimov et al. 2015; Cai et al. 2017; Mirza and Osindero 2014; Reed et al. 2016a; Odena et al. 2016; Zhang et al. 2016, 2017, 2021a; Sharma et al. 2018; Park et al. 2018; Xu et al. 2017; Qiao et al. 2019; Gou et al. 2020; Wang et al. 2020; Li et al. 2020a; Gao et al. 2021; Baraheem and Nguyen 2020a, 2020b) is a way to represent the human-written sentence visually where the semantic meaning in the text is preserved. In the early stage of research, generating an image from text depended basically on the analysis of word-to-image correlation incorporated supervised approaches to discover the best matching of image content to the textual descriptions. Notwithstanding, this solution has a significant gap if the text description is unseen in the training dataset during the training process, leading to image synthesis failure. However, the research community has made good efforts in synthesizing images from the text after the development in unsupervised deep learning approaches where the research community has shifted to Generative Adversarial Networks (GANs) (Goodfellow et al. 2014) to generate images from the text whether the textual descriptions are seen or not during the training stage. Synthesizing a realistic image on a simple dataset of a single object is a solved task. However, generating a realistic scene that is composed of many objects is a challenging task. This process is naturally accomplished by humans without efforts involved, especially during the childhood learning

period when children attempt to visualize the words by their imaginations. Nonetheless, it is a complicated task when it comes to a machine since we need not only to convert the characters to pixels but also, we need to match the generated image to the corresponding specification in the real world. Thus, much research has been carried out in this field to enhance the quality of synthetic images.

Sketch-to-image synthesis (Chen et al. 2009; Eitz et al. 2011; Szanto et al. 2011; Rajput and Prashantha. 2019; Yu et al. 2016; Sangkloy et al. 2016a, 2016b; Failed 2017; Xian et al. , 2018; Chen and Hays 2018; Liu et al. 2019, 2020; Zhang et al. 2020; Gao et al. 2020; Osahor et al. 2020; Lu et al. 2018; Li et al. 2021) is the process of converting a sketch to a color image. Sketch is a rough, simple, easy way to draw even if the person does not have any artistic skills. To translate the sketch, which conveys fewer details and features than the image, researchers first start the task by using sketch-based image retrieval (SBIR) systems (Chen et al. 2009; Eitz et al. 2011; Szanto et al. 2011; Rajput and Prashantha. 2019). Particularly, a database or a search engine is incorporated into the systems to retrieve the most similar image to the input sketch. Due to the limitation in SBIR systems, such as the inability to perform fine-grained retrieval, researchers have shifted to deep convolutional neural networks (CNNs), which obtains better results than SBIR due to feature learning rather than feature engineering. However, to generate more realistic images from the sketches, the research committee incorporates GANs to map the sketch to the corresponding image. Although GAN-based sketch-to-image models generate photorealistic images on a single sketched object, the task of sketch-to-image for complex scenes is still a challenging problem.

Image-to-image synthesis (Radford et al. 2015; Arjovsky et al. 2017; Failed 2016; Karras et al. 2017; Park et al. 2019a; Isola et al. 2016; Zhu et al. 2017, 2020; Huang et al. 2018a; Wu et al. 2019; Lin et al. 2019; Sushko et al. 2020; Zhang et al. 2021b) is the process of mapping an input image from a source domain to an output image in another different domain. The goal is to transfer a source image to a target desired image by changing some properties in the source image while preserving the content. One example is to map segmentation maps or edge maps to colorful images. Recently, many researchers have studied this task, and promising results have been achieved. Most of the models incorporate GAN to obtain the goal of translation in many ways since GAN has shown great potential for image generation even with unseen data (Huang et al. 2018b).

Speech-to-image synthesis (Chen et al. 2017b; Hao et al. 2018; Li et al. 2020b; Wang et al. 2021) is the process that takes audio as an input and generates a counterpart image in which the visual component conveys similar semantics as the sound component. In fact, speech-to-image generation has recently attracted researchers to develop models that are able to convert sounds into images. This task is trivial to perceive as a human, where the human can easily map sound into appearance through imagination power. However, this task is challenging when it comes to machine perception.

The remaining sections of this paper are as follows. We first review various image generation models classified into three categories: Text-to-image, sketch-to-image, and other-to-image synthesis divided into image-to-image and speech-to-image synthesis. In the next section, benchmarked datasets used by the discussed models are reviewed in brief detail. After that, evaluation metrics used to validate the introduced models are summarized. In the end, we discuss some unsolved challenges with image generation and conclude by suggesting possible future research directions.

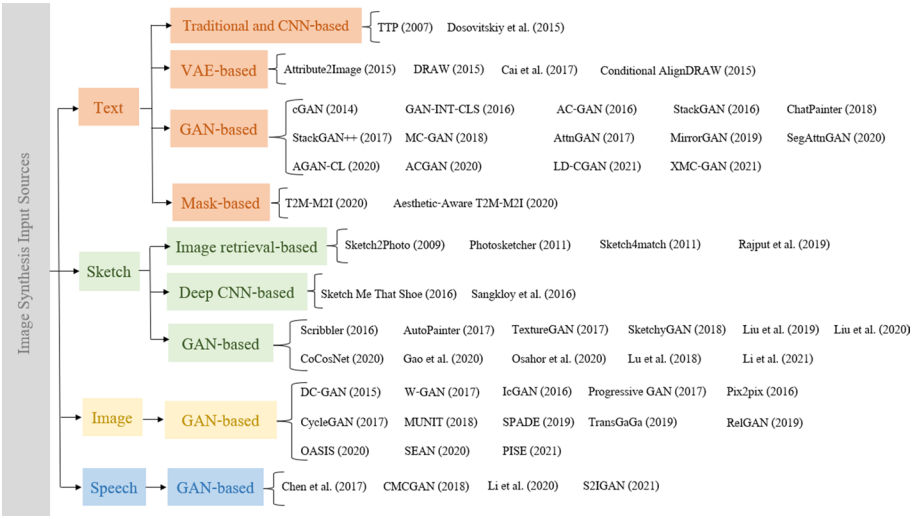

Fig. 1 Taxonomy of image synthesis research

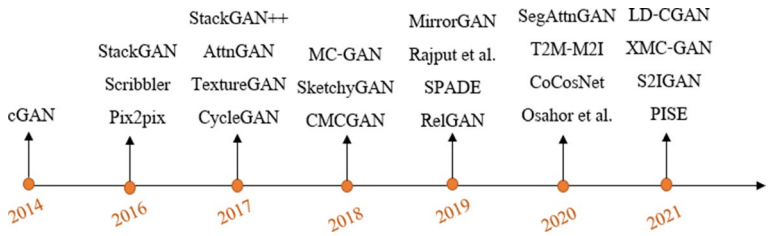

Fig. 2 Timeline of important image synthesis models

2 Image Synthesis Models

This section introduces a comparative review of various image synthesis models divided into subsections based on the input forms, *i.e.*, text, sketch, audio, or another image. Then, each subsection is divided into subsubsections based on the architecture and framework. While Fig. 1 illustrates the taxonomy of image generation, Fig. 2 demonstrates the timeline of some introduced models. Additionally, Table 1 summarizes the main methodological approaches of the reviewed models divided based on the input modality.

2.1 Text-to-image synthesis models

Text-to-image synthesis is the process of generating photorealistic images from the text where the visual content should be semantically consistent with the text. It converts the meaning of the text, *i.e.*, natural language descriptions, class labels, keywords, and attributes, into an image correspondingly. Text is passed into the models via

Table 1 Summary of the main methodological approach of the reviewed image generation models

Input	Main methodological approach	Method
Text	Supervised / paired	TTP (Zhu et al. 2007)
		Dosovitskiy et al. (Dosovitskiy et al. 2015)
		Attribute2Image (Yan et al. 2015)
		Conditional AlignDRAW (Mansimov et al. 2015)
		Conditional GAN (cGAN) (Mirza and Osindero 2014)
		GAN-INT-CLS (Reed et al. 2016a)
		AC-GAN (Odena et al. 2016)
		StackGAN (Zhang et al. 2016)
		ChatPainter (Sharma et al. 2018)
		StackGAN + + (Zhang et al. 2017)
		MC-GAN (Park et al. 2018)
		AttnGAN (Xu et al. 2017)
		MirrorGAN (Qiao et al. 2019)
		SegAttnGAN (Gou et al. 2020)
		ACGAN (Li et al. 2020a)
		LD-CGAN (Gao et al. 2021)
		XMC-GAN (Zhang et al. 2021a)
		T2M-M2I (Baraheem and Nguyen 2020a)
		Aware T2M-M2I (Baraheem and Nguyen 2020b)
	Unsupervised / unpaired	DRAW (Gregor et al. 2015)
		Cai et al. (Cai et al. 2017)
Speech	Supervised / paired	Chen et al. (Chen et al. 2017b)
		CMCGAN (Hao et al. 2018)
		Li et al. (Li et al. 2020b)
		S2IGAN (Wang et al. 2021)
Sketch	Supervised / paired	Sketch2Photo (Chen et al. 2009)
		Photosketcher (Eitz et al. 2011)
		Sketch Me That Shoe (Yu et al. 2016)
		Sangkloy et al. (Sangkloy et al. 2016a)
		Scribbler (Sangkloy et al. 2016b)
		Auto-painter (Failed 2017)
		TextureGAN (Xian et al. , 2018)
		SketchyGAN (Chen and Hays 2018)
		Gao et al. (Gao et al. 2020)
		Lu et al. (Lu et al. 2018)
	Unsupervised / unpaired	Sketch4match (Szanto et al. 2011)
		Rajput et al. (Rajput and Prashantha. 2019)
		Liu et al. (Liu et al. 2019)
	Self-supervised	Liu et al. (Liu et al. 2020)
	Weak supervised	CoCosNet (Zhang et al. 2020)
	Semi-supervised	Li et al. (Li et al. 2021)

Table 1 (continued)

Input	Main methodological approach	Method
Image	Supervised / paired	Invertible Conditional GAN (IcGAN) (Failed 2016)
		Pix2pix (Isola et al. 2016)
		SPADE (Park et al. 2019a)
		OASIS (Sushko and E. Sch¨onfeld, D. Zhang, J. Gall, B. Schiele, and A. Khoreva. 2020)
		SEAN (Zhu et al. 2020)
		PISE (Zhang et al. 2021b)
	Unsupervised / unpaired	DC-GAN (Radford et al. 2015)
		W-GAN (Arjovsky et al. 2017)
		Progressive GAN (Karras et al. 2017)
		CycleGAN (Zhu et al. 2017)
		MUNIT (Huang et al. 2018a)
		TransGaGa (Wu et al. 2019)
		RelGAN (Lin et al. 2019)

deterministic mappings, functions, or networks. Different ways have been proposed and used to encode the text and obtain the text embeddings which used to generate images.

Bag-of-Words (Harris 1981) and Word2Vec (Distributed representations of words and phrases and their compo- sitionality. 2013) are some traditional text representations that are less efficient; and hence, they are less used in text-to-image synthesis field. Skip-Thought vectors (Kiros et al. 2015) are used to encode the text descriptions in an unsupervised manner, where only the encoder network is utilized for text embedding. Reed et al. (Reed et al. 2016b) propose to use a pre-trained character-level convolutional recurrent neural network (char-CNN-RNN) to encode the input text. The pre-trained char-CNN-RNN learns a correspondence function between text and image depending on the class labels. Rather than using the static, fixed text embedding produced by a pre-trained encoder, Conditioning Augmentation (CA) (Zhang et al. 2016) is proposed to encourage smoothness over the conditioning manifold. CA works by randomly sampling the latent variables from a distribution, in particular, a Gaussian distribution. Both the covariance matrix and the mean are functions of the text embedding. Furthermore, Sentence Interpolation (SI) (Souza and Jˆonatas Wehrmann, and Duncan D. Ruiz. 2020) is introduced as a deterministic method to obtain a continuous and smooth embedding. In AttnGAN (Xu et al. 2017), char-CNN-RNN (Reed et al. 2016b) is substituted with a bi-directional LSTM (BiLSTM) (Schuster and Paliwal 1997) to encode the input text, where the feature vectors are extracted via concatenating the hidden states to create a feature matrix for every and each word in the input text. Following this, the global sentence vector is created. A Deep Attentional Multimodal Similarity Model (DAMSM) is pre-trained to find out the word features that match the subregions of the image. Moreover, BERT (Devlin et al. 2018), which stands for Bidirectional Encoder Representations from Transformers, is very popular and commonly used in natural language processing since the contextual relations between words and even sub-words in the text are learned.

Thus, the pre-trained BERT is leveraged as a text encoder to obtain text embeddings in text-to-image generation.

In this section, image synthesis from text is organized into four main categories, which are traditional learning-based and CNN-based, VAE-based, GAN- based, and mask-based text-to-image synthesis.

2.1.1 Traditional learning-based and CNNs- based text-to-image synthesis

Text-to-Picture (TTP) Synthesis System (Zhu et al. 2007) relies on a combination of natural language processing, computer vision, computer graphics, and machine learning. The process is concentrated mainly on a search technique and supervised learning methods, as can be seen in Fig. 3. TTP utilizes words and images correlation to map the text descriptions to images forming "text-picture" pairs. Then, TTP searches based on these pairs for the most appropriate image parts conditioned on the given text. The key limitation in TTP system is when the word-image correlation pair does not exist in the training data, meaning that the model does not have the ability to produce new images where the text descriptions are unseen in the training dataset. Moreover, TTP system could suffer during the image layout step, generating images that are not semantically matched with the text or generating images that are visually incorrect.

Dosovitskiy et al. (Dosovitskiy et al. 2015) proposed a model to generate 2D images of 3D chairs. A deep convolutional decoder is used to produce images based on the respective parameters, *i.e.*, type, color, saturation, viewpoint, lighting condition, brightness, zoom, and position, using supervised learning. Because of this, it is only able to generate objects that were seen during the training process; therefore, it lacks the ability to generate new image content.

2.1.2 Variational autoencoders-based text-to-image synthesis

Attribute2Image (Yan et al. 2015) leverages a generative model to generate new visual content based on conditional variational auto-encoder (CVAE) to convert unsupervised learning of VAE into supervised training mode by incorporating a one-hot encoded label vector in both the encoder and decoder. It uses variational auto-encoders (VAEs) (Kingma and Welling 2013) to disentangle an image into a foreground latent variable and

Fig. 3 A generated image by TTP model for the given text

a background latent variable. Two encoders and two decoders are leveraged for foreground and background latent variables and for producing a foreground image along with the full final image through a composition process, respectively. The final full image is composed of the foreground image and the background image gated by the visibility map, which is the foreground's mask map. This foreground's mask map is used to determine the shape and position of the foreground; hence, guiding the composition process of foreground and background image. The image synthesis is conditioned on the visual attributes, *i.e.*, age, gender, hair color, and expression extracted from the text description and passed through the network as text embeddings, as illustrated in Fig. 4. Yan et al. convert text describing visual attributes into text embeddings for Labeled Faces in the Wild (LFW) (Huang et al. 2008) and Caltech-UCSD Birds-200–2011 (CUB) (The caltech-ucsd birds200-2011 dataset. Advances in Water Resources - ADV WATER 2011) datasets. For LFW dataset (Huang et al. 2008), 73-dimensional attribute score vector obtained by Kumar et al. (2009) is used to describe various facial attributes, *i.e.*, age, gender, and expression, just to name a few. With regards to CUB dataset (The caltech-ucsd birds200-2011 dataset. Advances in Water Resources - ADV WATER 2011), 312-dimensional binary attribute vector is utilized to describe bird parts and colors provided by The caltech-ucsd birds200-(2011) dataset. Advances in Water Resources - ADV WATER (2011). Therefore, after the two encoders encode the original input image into foreground and background latent variables, the foreground latent variable is fed with the extracted attribute vector into the decoder to generate the foreground image. In the meantime, the background latent variable along with the foreground latent variable and attribute vector are fed into the other decoder to produce the background image. Following this, the target final image is generated and a composite of both foreground and background images. Obviously, the major limitation of the attribute-conditioned generative model is that the learnable representations are limited to the provided visual attributes extracted from the text during the supervised training.

DRAW (Gregor et al. 2015) uses a deep recurrent neural network incorporating VAE and a spatial attention technique to iteratively construct an image through patches. Hence, imitating the human drawing where only particular part of an image is under observation, while the other parts are ignored is the process behind DRAW. Fig. 5 shows an example of generating MNIST digits, where each digit in each row is constructed successively, and the red rectangle determines the part of image under observation by the network. One drawback in this model is the image blurriness, especially with generating complex images.

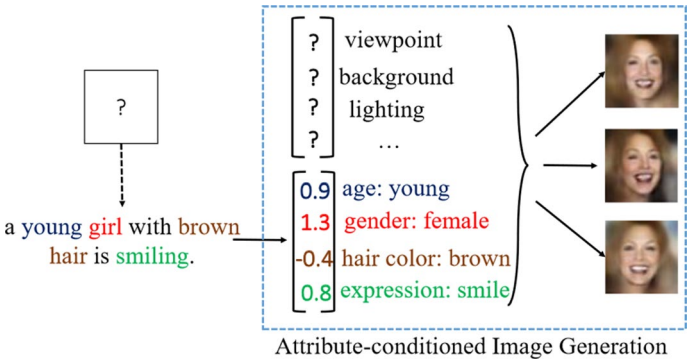

Fig. 4 Attribute-conditioned image generation example. Image courtesy from (Yan et al. 2015)

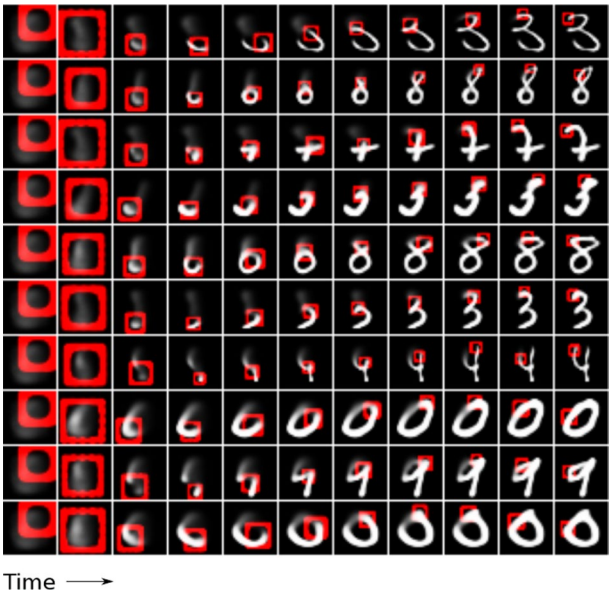

Fig. 5 Example of generating MNIST digits. Image courtesy from (Gregor et al. 2015)

Conditional AlignDRAW (Mansimov et al. 2015) incorporates a recurrent VAE that generates an image from the caption using an attention mechanism and heavily depending on DRAW (Gregor et al. 2015) model. It iteratively draws parts of the image (patches) focusing on the relevant words in the text to make attention sharper. During a post-preprocessing step, the synthesized image is refined by an adversarial network called Deterministic Laplacian Pyramid (Denton et al. 2015) to reduce the image blurriness, which is the major drawback in DRAW (Gregor et al. 2015) model. However, image blurriness is still a problem in this model.

Cai et al. (Cai et al. 2017) proposed a multi-stage VAE. In fact, research has enhanced the capability of VAE to reduce blurriness (Chen et al. 2016; Gulrajani et al. 2016; Kingma et al. 2016). Although VAE works with simple distribution, it suffers from image blurriness with multimodal distribution due to L2 loss (Mao et al. 2016). To overcome this problem, a multi-stage VAE is used. In the first phase, a coarse low-quality image is generated, followed by a refinement phase. In this second phase, a fine high-quality image is produced based on the coarse image as an input. This method enhances the clarity of the images as opposed to the original VAE (Kingma and Welling 2013) since it generates images from coarse to fine in two stages. Nevertheless, this method requires more computations. Fig. 6 shows the multi-stage VAE process of generating an image from the CelebA dataset.

2.1.3 Generative adversarial network GAN-based text-to-image synthesis

Goodfellow et al. (Goodfellow et al. 2014) pioneered the generative adversarial network (GAN) model, where the generator (G) competes against the discriminator (D). The generator (G) attempts to fool the discriminator by generating photo-realistic images; thus, the discriminator perceives these generated images as real images. Meanwhile, the discriminator (D) attempts to differentiate between real and generated images. GAN trains the two networks through a minimax two-player game. While the generator (G) tries to minimize the gap between the generated image and the real image, the discriminator (D)

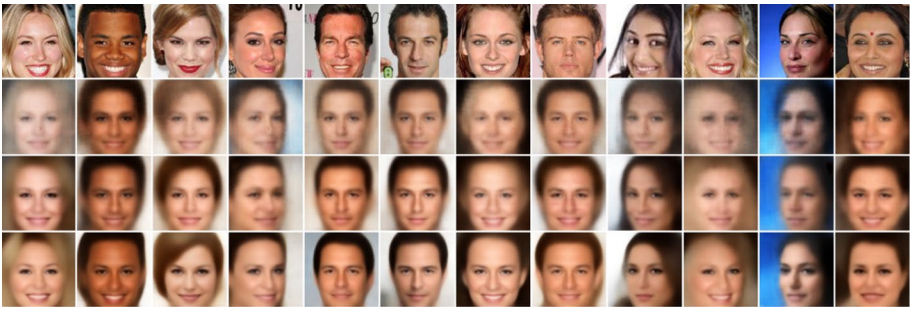

Fig. 6 An example of the multi-stage VAE process, where first row is the original input images, followed by a second row which is the generated images produced by the original VAE. Third row is the generated images of deep residual VAE, while fourth row is the generated images of multi-stage VAE model. Image courtesy from (Cai et al. 2017)

tries to maximize the correctness of the classification probability that is assigned to both the generated and real images. Fig. 7 demonstrates GAN architecture. Indeed, GAN is trained iteratively, so the generator (G) and the discriminator (D) are updated in each iteration, resulting in a better performance for each model.

One main challenge in GAN is that the generator and discriminator are not balanced, meaning that the performance of the discriminator significantly surpasses the performance of the generator because of the generating samples' limitation with limited varieties of samples, leading to overfitting problem, where the model is not able to generalize. Furthermore, GANs might suffer from mode collapse. This problem originates when the generator fails to produce diverse data samples as the distribution of the real original data. Hence, the produced data samples are highly similar or even identical. In fact, sometimes, the generator produces the same output or a limited number of outputs repeatedly since the generator finds out that the one output or the limited number of outputs is most realistic and plausible to the discriminator so that the generator is able to fool the discriminator. Meanwhile, the best strategy of the discriminator is to reject that sample. However, if the discriminator's next generation remains stuck in the local minimum and unable to find the best strategy, the next iteration of the generator finds easily the most realistic and plausible output for the present discriminator. Therefore, the generator's each iteration over-improves for a specific discriminator, leading to the

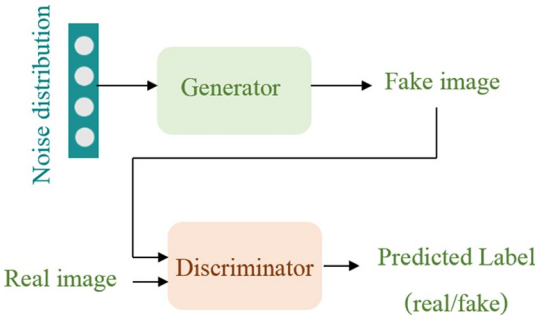

Fig. 7 A common GAN architecture for data synthesis. There are two main components: generator to generate synthesized sample, and discriminator to classify the synthesized sample is fake or real

discriminator's inability to find a way to get out of the trap. At the end, the generator produces limited undiversified samples with little representative of the population.

Another probable limitation in GAN is the training instability due to the fact that GAN consists of two networks that are constantly competing against others, and each network has its loss function, making the training process unstable and slow, and causing the non-convergence problem. Moreover, the discriminator might be too successful so that the discriminator is able to easily distinguish real and generated samples, resulting in diminishing or vanishing the generator gradient. Hence, the generator is unable to learn anything.

However, the research community has shifted to leveraging GAN in the image synthesis domain because of the ability to produce sharper, clearer, and more realistic images, compared to previous methods.

Conditional GAN (cGAN) (Mirza and Osindero 2014) is a conditional version of GAN. In a basic unconditioned generative model, the model generates images without any control on the modes. However, with conditioned generative model, the model produces images conditioned on additional input, *i.e.*, class labels. Therefore, unlike the basic unconditioned generator (G), where only noise vector is fed to the network, noise vector along with condition extension (*i.e.*, label) are fed to the generator to produce a synthetic image, which in turn is fed to the discriminator (D) along with the real image and the condition extension (*i.e.*, label). Hence, the discriminator (D) differentiates between real and fake images conditioned on the conditional feature. Fig. 8a shows the architecture of cGAN. This method works well on MNIST (Deng 2012) and MIR Flickr 25 k (Huiskes and Lew 2008) datasets, but it suffers from complex text descriptions which prevent the model from generating high-quality fine-grained images. Because of this, cGAN struggles with complicated datasets such as COCO dataset (Lin et al. 2014).

GAN-INT-CLS (Reed et al. 2016a) is an extension of DC-GAN (Radford et al. 2015) (see Sect. 2.3.1 for details). It extends DC-GAN by incorporating conditional recurrent neural network conditioned on word-level of the input text. It takes the text descriptions as input and converts them into visually paired text features. In the next stage, these text features are fed into the RNN model to generate an image. Unlike cGAN (Mirza and Osindero 2014) where the condition features are the additional input, *i.e.*, labels or tags, the condition features of GAN-INT-CLS are extracted directly from the input text. This model has two main components, namely, image-text matching aware discriminator GAN-CLS and manifold-interpolation regularizer GAN-INT. GAN-CLS supplies the discriminator with additional information, which is a pair consisting of the

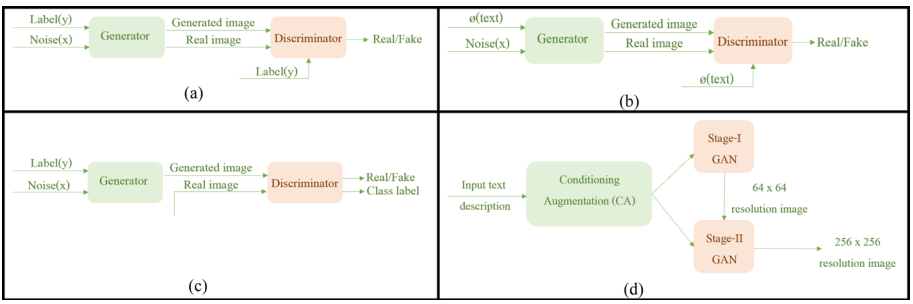

Fig. 8 **a** Conditional GAN architecture. **b** GAN-INT-CLS architecture. **c** AC-GAN architecture. **d** StackGAN architecture

real image and the mismatched text as an input in which the discriminator should learn to score these pairs as fake. This way, the discriminator is able to provide the generator with more information since not only the image realism is provided, but also image/text matching is learned by the discriminator. GAN-INT produces additional synthetic text embeddings through interpolation between embeddings of training dataset captions, leading to improving the quality of the generated images. An overview of GAN-INT- CLS model is represented in Fig. 8b. **AC-GAN** (Odena et al. 2016), the short form of Auxiliary Classifier GAN, is another variant of GAN. It extends conditional GAN (Mirza and Osindero 2014) in a way that the class label is provided only to the generator as a condition but not provided to the discriminator. As a result, the discriminator should be able to predict the class label of the provided image instead of perceiving it as an input, leading to producing two outputs which are the image realness probability and class label probability, as shown in Fig. 8c. Moreover, due to the use of the auxiliary classifier layer that controls the generated images by predicting the class label of images, this model enhances the generated images' diversity.

StackGAN (Zhang et al. 2016) is a multi-GAN model that uses a sketch-refinement process; thus, it consists of two stages stacked on top of each other in a cascaded manner to generate 256×256 high-quality naturalistic images. In Stage-I, the input text is encoded into text embedding conditioned on the main features that are used then to produce a low-resolution image of 64×64. The output of the first stage contains only main information, such as the shape and colors of the object. Then, the output of the first phase is fed into the second stage along with the text embedding as inputs to add fine details, resulting in a high-resolution realistic image of 256×256. Fig. 8d shows an overview of StackGAN model. In fact, StackGAN randomly selects the conditioning features from a Gaussian distribution whose mean and variance are estimated from the text embeddings. These conditioning features are then leveraged via Conditioning Augmentation technique that helps in smoothing the latent condition manifold to improve the stability during training and improve the generated images' diversity. However, it only works on global information, *i.e.*, sentence-level, not word-level. Therefore, it fails to produce fine-grained high-quality detailed images. **Chat- Painter** (Sharma et al. 2018) is an improvement over StackGAN (Zhang et al. 2016). An additional dialogue module is leveraged to generate fine-grained naturalistic images for complicated images such as COCO dataset (Lin et al. 2014), where the caption might not include details about every object in the image or might provide only general information about the background. This dialogue module is made up of the captions provided with COCO dataset (Lin et al. 2014) along with dialogues from Visual Dialog dataset (VisDial) (Das et al. 2017). Then, based on questions on the captions and answers to these questions based on the captions and the corresponding images, the model is trained. This model generates a photo-realistic images of 256×256 resolution; however, distortion is one major problem with ChatPainter model. Later, **StackGAN++** (Zhang et al. 2017) was proposed by Zhang et al. (Zhang et al. 2017) to extend StackGAN to solve the lack of generating high-quality detailed images. This model consists of two models. The first model is StackGAN-v1 which is StackGAN model (Zhang et al. 2016) that contains two stages as described previously to generate images from the input text via a sketch-refinement process. The second model is StackGAN-v2 which is a multi-stage GAN of StackGANs, for both conditional and unconditional image synthesis. StackGAN-v2 incorporates multiple generators and multiple discriminators organized in a tree-like structure. StackGAN-v2 takes the out- put of StackGAN (Zhang et al. 2016) as an input and generates images with various scales for each input, relying upon various branches of the tree. Although this model demonstrates more training stability and better high-quality synthetic

images as compared to StackGAN (Zhang et al. 2016), it is sometimes unable to converge especially with complicated dataset, such as COCO dataset (Lin et al. 2014).

MC-GAN (Park et al. 2018) focuses on both the foreground and the background by taking three inputs which are a text description that describes the foreground object, a base image containing only the background to be used as a canvas, and the object location. To tackle complicated multi-modal conditions in GAN, a synthesis block is used to separate the foreground objects and the background during the training process. In the synthesis block, while the background feature is easily extracted from the base image using only convolution and batch normalization (Ioffe and Szegedy 2015) layers, the foreground feature is the output of the previous layer. This synthesis block allows to synthesize a foreground object described in the input text in a given location with the targeted background to produce photorealistic images of size 128×128. **AttnGAN** (Xu et al. 2017) incorporates an attention module to concentrate on word-level, leading to generation of fine-grained high-quality images. Therefore, not only the input text is encoded into a global sentence vector, but also words in the input text are encoded into a word vector. Then, based on the global sentence vector, a low-quality image is generated in the first phase. In the subsequent phases and through multi-stage refinement, the generator of AttnGAN concentrates on a specific part of the image each time based on the related words in the word vector to refine the image independently and successively, resulting in fine-grained details. Moreover, it has a Deep Attentional Multimodal Similarity Model (DAMSM) that is used after the final phase's outcome to compute the degree of similarity between the generated image and the input text in both sentence-level and word-level during the generator training. Because this model uses global and local information, it generates photo-realistic images. However, it sometimes fails to represent the global structure, resulting in distorted images that may impact the naturality of the local semantic details.

MirrorGAN (Qiao et al. 2019) is a combination of text-to-image and image-to-text methods, where image-to-text model is considered as a mirrored model of text-to-image model. This helps in reflecting the semantics of the input text visually during generating an image. Therefore, MirrorGAN consists of three main modules: Semantic Text Embedding Module (STEM), Global–Local collaborative Attentive Module (GLAM), and Semantic Text REgeneration and Alignment Module (STREAM). STEM is responsible for taking the input text and encoding it into sentence-level and word-level embeddings through a recurrent neural network (RNN). Meanwhile, GLAM is multi-feature transformers arranged on top of each other to extract visual features. Finally, STREAM is responsible for regenerating the text description from the generated image by GLAM to semantically align the generated caption with the input text. **SegAttnGAN** (Gou et al. 2020) uses segmentation data provided by a spatial self-attention network as additional input with the text description to guide image synthesis via global spatial constraints. It works on both sentence-level and word-level to extract features through LSTM encoder. The architecture of SegAttnGAN is shown in Fig. 9a. The segmentation attention module enhances the realism of the generated images by regulating generated image layouts and maintaining object shapes. However, the segmentation map is a necessary input that should be explicitly provided to the model during the inference stage.

AGAN-CL (Wang et al. 2020) consists of two components. The first component is a contextual network that is used to generate image contours. Image contours are considered as spatial constraints that are fed into the second model. The second component is a cycle transformation autoencoder that is used to convert the image contours to naturalistic image. Image contours attempt to guide the image generation process by concentrating on object positions and shapes; and thus, it provides an ability to align the generated image with the

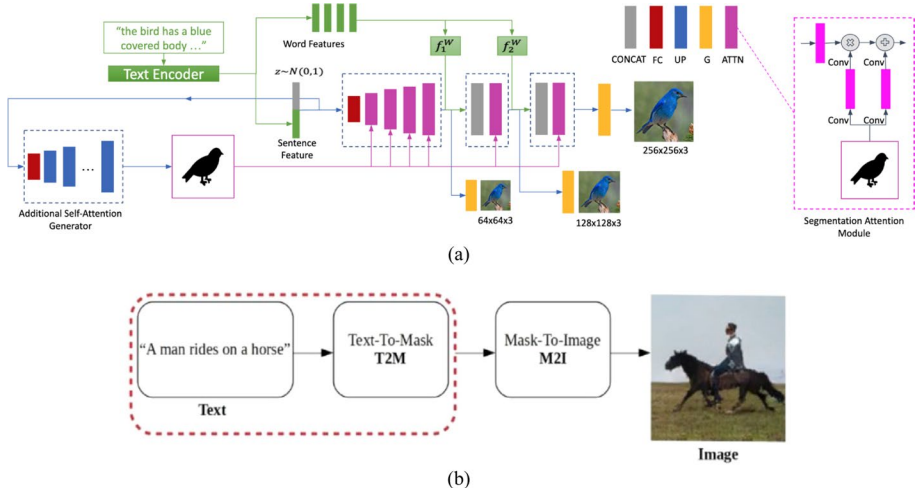

Fig. 9 An overview of **a** SegAttnGAN model. **b** T2M-M2I model. Images courtesy from (Gou et al. 2020; Baraheem and Nguyen 2020a), respectively

input text. **ACGAN** (Li et al. 2020a), the short form of Attentional Concatenation Generative Adversarial Network, relies on a multilevel cascade GAN architecture, starting from a low- resolution image to gradually generate a high-resolution photorealistic image of up to 1024×1024. During the training stage, new layers are incrementally added to the generator and the discriminator to generate a large image. In addition, each subsequent layer takes the output and the word vector from the previous layer as inputs to generate fine-grained details. It uses a deep attentional multimodal similarity model to concentrate on matching the word vectors and image regions in a semantic space, leading to generating fine-grained images based on the semantics of word-level. The limitation lies when generating complex images of multiple objects. **LD-CGAN** (Gao et al. 2021), the acronym of Lightweight Dynamic Conditional GAN, was introduced to solve the problem of generating high-resolution images. It consists of Pyramid Attention Refine Block (PAR-B) that is used to strengthen the multi-scale features via incorporating the spatial coherence of multi-scale context. Thus, it generates large images with the resolution of 642×642 and 1282×1282. Moreover, it reduces the training computation complexity in comparison with GAN-based text-to-image models. **XMC-GAN** (Zhang et al. 2021a), Cross-Modal Contrastive Generative Adversarial Network, was proposed to enhance the realism of the generated images and the alignment of the generated images with the input text. The generator contains a self-modulation layer for global information and an attentional self-modulation layer for local information to generate fine-grained images. The discriminator attempts to judge the generated image as real or fake and to encode both global image features and regional features for contrastive learning. Thus, it generates high-quality fine-grained images for short and long text descriptions.

2.1.4 Mask-based Text-to-image Synthesis

Instead of a direct mapping from an input text to an image, an intermediate output is generated first which is the mask map. The mask map is generated based on the semantics of the input text. Then, the generated mask map is used as an input to the mask-to-image model

to generate photorealistic images. Since the images are not generated from the text in one shot, the generated images preserve the semantic and spatial information, leading to clearer layouts with the delineated shape of the objects.

In **T2M-M2I** (Baraheem and Nguyen 2020a), to generate naturalistic and semantically well-aligned images, Baraheem et al. (2020) divided the text-to-image task into two subtasks. The first component is text-to-mask (T2M), where the mask maps are generated conditioned on the text descriptions and the mask dataset. They proposed using anchor points during generating the mask map and after capturing the object masks to reflect the spatial relationship among the objects and the overall layout. The second component is mask-to-image (M2I), where it takes the generated mask map as an input and generates the texture, leading to generating a photorealistic image. In the second stage, SPADE (Park et al. 2019b) mask-to-image model is used (see Sect. 2.3.1 for details). The framework overview is shown in Fig. 9b. This model produces more realistic images without distortion due to leveraging the two components along with utilizing the anchor points.

Aesthetic-Aware T2M-M2I (Baraheem and Nguyen 2020b) is the extension of T2M-M2I model, where aesthetic criteria are preserved to generate not only photorealistic images but also appealing images. It is composed of several parts. The first part has the role of generating a set of mask maps conditioned on the input text and with the help of a mask dataset. Then, based on aesthetic composition rules, in particular, the rule of thirds (Gadde and Karlapalem 2011) and the rule of formal balance (Liu et al. 2010), the aesthetic score is computed for each mask map in the generated set and then ranked. Following this step, only three mask maps are selected to be fed into mask-to-image model based on the aesthetic score. Specifically, mask maps with the highest, the lowest, and the average aesthetic score are chosen. This subset of mask maps is fed into the mask-to-image model, in particular, SPADE (Park et al. 2019b), to generate naturalistic images, followed by another ranking based on the same used aesthetic criteria to specify the most aesthetically appealing image in the subset.

2.2 Sketch-to-Image Synthesis Models

Sketch is a rough, simple, and faster way of graphically representing an image. It can be used to record an idea or to draw what we see at the moment for later modifications. However, it only provides essential features and lacks details, *i.e.*, the color, the saturation, and the brightness. Therefore, the research community has developed many models to translate a sketch into an image. This allows to create photorealistic images without artistic skills or expertise in the art domain. In this paper, sketch-to-image methods are organized into three categories which are sketch-based image retrieval and synthesis, deep convolutional neural networks (CNNs) sketch-to-image synthesis, and generative adversarial network GAN-based sketch-to-image synthesis.

2.2.1 Sketch-based Image Retrieval and Synthesis

Sketch-based image retrieval and synthesis methods use sketches as inputs to query large image databases in order to retrieve matched images. The retrieval process depends on the used descriptor that is utilized to extract the features from the image such as global and local descriptors. Edge histogram descriptor (EHD) (Eitz et al. 2009), Histogram of gradients (Salembier et al. 2002), and Angular Radial Partitioning (ARP) (Chalechale et al. 2004) are global descriptors commonly used for analyzing and classifying images. Local

descriptors such as Histograms of Oriented Gradient (HOG) (Dalal and Triggs 2005), Scale Invariant Feature Transform (SIFT) (Lowe 2004), and Local Binary Patterns (LBP) (Ojala and M. Pietik¨ainen, and D. Harwood. 1996) are widely used in extracting image features. Thus, researchers have developed various descriptors to tackle this problem. Myriad works have been done in sketch-based image retrieval systems (SBIR). We review some of the effective methods in this paper.

Sketch2Photo (Chen et al. 2009) consists of several stages: image search, image segmentation, and image composition. It takes a sketch annotated with labels, where each label describes a foreground object or a background. Then, by seamlessly composing multiple images retrieved from the Internet and based on the given labels, a photorealistic image is generated. To tackle the problem of retrieving improper images online, this model uses a filtering scheme. After that, each discovered image is segmented to locate only the element that matches the corresponding element in the sketch. To allow seamless image synthesis, an image blending technique is utilized to obtain multiple compositions based on the retrieved images. Then, the model selects the optimal synthetic image depending on the estimated quality score. The structure of Sketch2Photo is shown in Fig. 10a. Since this model relies on search engine and image composition, several problems may emerge, such as incorrect occlusion, incorrect perspective, or incorrect element size. Another problem might appear with the complex scene of multiple objects. These problems lead to reducing the realism of the generated image and producing artifact effects. Meanwhile, **Photosketcher** (Eitz et al. 2011) first retrieves best the matches from the database, and then the user interactively chooses the best match. Following this step, an interactive composition is utilized to create an image by composing all parts together via extracting the foreground and matting approaches, where each retrieved image is segmented, and only the queried part is extracted, followed by pasting the extracted element into the intermediate result. The composition step relies on Gaussian Mixture Models (Rasmussen 1999) that are learned to identify both desired and undesired elements from the retrieved matched set of images. In addition, the composition step depends on Graphcut (Rother et al. 2004) to extract the foreground, followed by a blending step to paste the element in the final image. The major challenge with Photosketcher is when the model is tried to compose dissimilar images, leading to difficulties during the composition step; therefore, unnaturalistic image is obtained.

Sketch4match (Szanto et al. 2011) uses three different descriptors, namely Histogram of Oriented Gradients (HOG) (Salembier et al. 2002), Edge Histogram Descriptor (EHD) (Eitz et al. 2009), and Scale Invariant Feature Transform (SIFT) (Lowe 2004). These descriptions are used to extract the features of the sketch and the image after multi-step preprocessing. The preprocessing step is required to bridge the gap between the sketch and the image due to enriching details in the color image compared to the drawing sketch. This helps in reducing the variations of the feature vectors to ensure better comparison and matching. The preprocessing step along with generating feature vectors on the images stored in a database are occurred offline before the retrieval process starts. Then, when the user enters or draws a sketch, a preprocessing step along with generating feature vectors are happened online during the retrieval process. The feature vectors of the sketch and list of images are compared based on Minkowski distance (Kruskal 1964) and classification-based retrieval (Liu and Dellaert 1998). Therefore, the closer images to the input sketch in terms of Minkowski distance (Kruskal 1964) are classified based on k-means clustering method (Comaniciu and Meer 2002), and then a cluster of images is displayed to the user. The retrieval system is implemented on a small dataset; and thus, finding correct matches for a sketch query might be a problem in this system. **Rajput et al.** (Rajput and Prashantha.

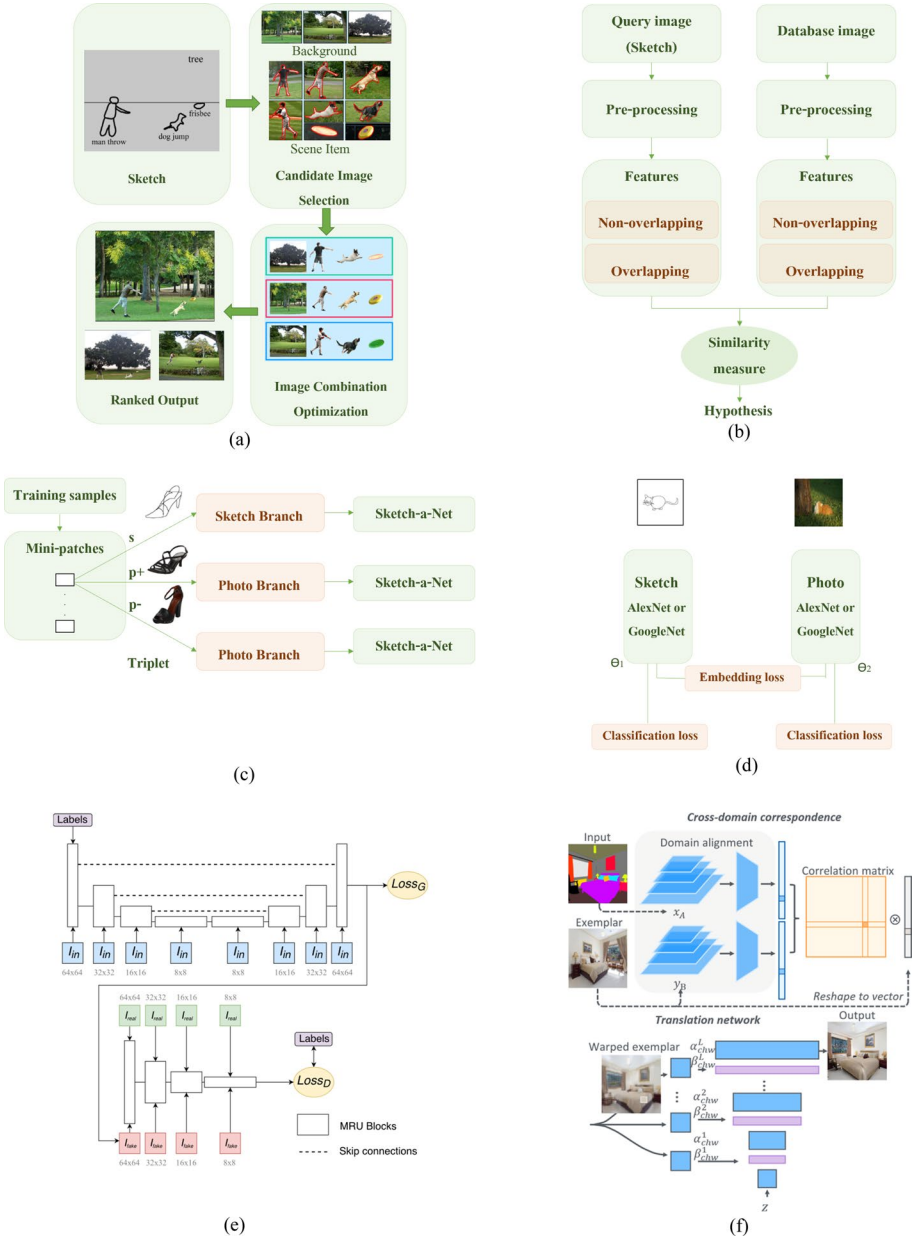

Fig. 10 The structure of **a** Sketch2Photo model. **b** Rajput et al. model (Rajput and Prashantha. 2019). **c** Sketch Me That Shoe model. **d** Cross-domain embedding model proposed by Sangkloy et al. (Sangkloy et al. 2016a). **e** SketchyGAN model, where image courtesy from the original paper (Chen and Hays 2018). **f** CoCosNet model, where image courtesy from the original paper (Zhang et al. 2020)

2019) utilizes a large-scale image database to ensure that the sketch-based retrieval system retrieves correct matches. Before extracting the features from the sketch and images, a preprocessing step is necessary. Global information is extracted based on Otsu's method

(Otsu 1979) to extract the contours and eliminate the weak contours. Then, the features are extracted from the sketch and the image in two stages. First, the global contours of both the sketch and image are divided into non-overlapping grids of size 10×10, and then the features are extracted for each grid based on the mean of the pixel's values. In the next stage, the global contours of both the sketch and image are divided into overlapping grids, where each grid is overlapped by 20% on left and upper side, and the features are extracted for each grid. Following the feature extraction step, a weighted similarity method is used, where the weights for both overlapping and non-overlapping grids are assigned in the range [0,1]. Then, the system retrieves the most similar images to the query sketch from the database based on Euclidean distance (Dokmanic et al. 2015). An overview of the system architecture is shown in Fig. 10b.

2.2.2 Deep convolutional neural networks CNNs sketch-to-image synthesis

One major issue in sketch-based image retrieval is that it is unable to perform fine-grained retrieval because of extracting the features manually. Moreover, it is unable to translate sketch edges that is drawn badly into image boundaries. To solve these issues, a deep convolution neural network (CNN) is trained to map sketches to images.

Sketch Me That Shoe (Yu et al. 2016) uses a deep CNN to learn rather than extract hand-crafted features. In this paper, the authors first create a dataset of 1,432 sketch-image pairs based on two classes (shoes and chairs) with 32,000 ground truth triplet ranking annotations. Then, a deep convolutional neural network, in particular, Siamese network (Chicco 2021) with a triplet ranking goal is implemented, so it uses three identical Sketch-a-Net model (Yu et al. 2015) for each network branch as illustrated in Fig. 10c. Although the dataset contains 1,432 pairs which is considered a large dataset, it is not sufficient to train a deep triplet ranking network since the model would overfit. **Sangkloy et al.** (Sangkloy et al. 2016a) first collect a large dataset of sketch-image pairs, namely, Sketchy dataset of size 75,471 of 12,500 objects categorized into 125 classes. The sketches are accomplished by crowd workers with fine-grained details. Then, Sketchy dataset is trained on cross-domain CNNs for sketches and images to retrieve not only instances of the correct class but also instances with fine-grained similarity to the input sketch. Fig. 10d shows the architecture of the cross-domain embedding model, where two deep convolutional neural networks, namely, AlexNet (Krizhevsky et al. 2017) developed in Caffe (Jia et al. 2014) and deep GoogLeNet (Szegedy et al. 2015), are utilized. The model is separately trained on sketches and images so that the model independently learns the appropriate weights for each domain. A pre-training process begins first with a pre-trained model of AlexNet or GoogLeNet on ImageNet dataset (Deng et al. 2009) to separately classify objects in sketches and images. Then, sketches and images are trained to embed their features into 1024 dimensions with Siamese contrastive loss and Triplet ranking loss functions, where both of these losses incorporate a softmax classification loss.

2.2.3 Generative adversarial network GAN-based sketch-to-image synthesis

Following the introduction of GAN, the research community has shifted to incorporate GAN in sketch-to-image models. The reason is that GAN-based models have the ability to generate more photorealistic images from the input sketches than other approaches, leading to better results. Therefore, lots of GAN-based sketch-to-image models have been developed to solve this challenge.

Scribbler (Sangkloy et al. 2016b) is conditioned on sketches, *i.e.*, edges and color strokes, where the color is identified by the user over the sketch. This allows the user to select the color of the sketched object; thus, generating images that satisfy the chosen sketch boundaries and the selected color. It is only trained to generate images of particular objects such as, cars, faces, and bedrooms. During training, the input sketches augmented with random color strokes are fed into the generator that consists of an encoder-decoder model with residual blocks (Failed 2015) to ease the training process, especially with deeper models. The generator results in generating an image with the same size as the input sketch. The objective function contains several losses to ensure realism and diversity. It has a pixel loss, a feature loss, an adversarial loss, and a total variation loss. **Auto-painter** (Failed 2017) translates sketches to painted cartoon images relying on cGANs (Mirza and Osindero 2014). It incorporates not only pixel loss and feature loss but also texture loss and total variation loss in the objective function during training the generator to produce an image with compatible colors for the corresponding sketch and increase the variations in the generated images. The generator is a feed-forward deep neural network instead of an encoder-decoder network to prevent information loss during dawnsampling and upsampling. The information, which is the boundary of the object, is very important in sketch-to-image task to generate a realistic image. There- fore, this model uses U-net (Ronneberger et al. 2015) by concatenating the encoder layers (sketch edge information) to the corresponding decoder layers (trained color painting information). Additionally, to allow the user to select the preferable colors, a color control is used based on Scribbler (Sangkloy et al. 2016b).

In **TextureGAN** (Xian et al. , 2018), the generated image is conditioned not only on sketch and color strokes, but also texture is controlled. This helps in producing naturalistic images via GAN along with an additional object textures control, where the user specifies the texture by placing one or more selected texture patches on top of the sketched object in any location. It follows Scribbler (Sangkloy et al. 2016b) in terms of the architecture and incorporates a pixel loss, a feature loss, an adversarial loss, and a local texture loss in the objective function to improve the realism and diversity. Additionally, the generator is able to generate new textures not seen in the training data because of incorporating the local texture loss. The local texture loss is computed as the difference between the Gram-matrix representation (Dumitrescu 2017) of patches in the generated images and the texture images. The TextureGAN is trained on only three classes, namely, handbags, shoes, and clothes. **SketchyGAN** (Chen and Hays 2018) augments Sketchy database (Sangkloy et al. 2016a) to address the lack of sufficient sketch-image pairs. The edge map-image pairs dataset is formed by collecting 2,299,144 images from Flickr from 50 classes, and then the edge maps are extracted from the collected images through Holistically-nested Edge Detection (HED) (Xie and Tu 2015), followed by several post-processing steps. The generator is an encoder-decoder network, where both of them are built with Masked Residual Unit (MRU) to enable the network to be iteratively conditioned on the inputs which are the image and feature maps. It learns by extracting new features from the input image and concatenating them with feature maps provided as input from the previous learning. To concatenate the output feature maps from the encoder to the output feature maps from the corresponding decoder, skip connection is applied. To enhance the generated images' quality, a conditional instance normalization (Dumoulin et al. 2016) is used in the generator and input sketches' labels are fed into the generator. The discriminator is also built with Masked Residual Unit (MRU) to classify the realism of the generated images and predicts the class labels of the generated images. The architecture of SketchyGAN is shown in Fig. 10e.

Liu et al. (Liu et al. 2019) leverage an unsupervised sketch-to-image synthesis, where it learns from an unpaired sketch-image dataset. It concentrates on both color and shape translation by separating the mapping into two tasks. In the first stage, sketches are translated to grayscale images via geometrical shape translation. Thus, only sketches from unpaired sketch-image data are fed into the network. The generator in the first stage consists of two encoder-decoder architectures to map sketches to grayscale images and another network to convert grayscale images to sketches. Moreover, an attention module is introduced in the first stage to ignore non-important regions. In the next stage, the generated grayscale images are translated to color images by enrichening the content and filling it with colorful details, texture, and shading through an encoder-decoder generator. To guide the network in filling these colorful details, a style transfer task is applied with optional reference color images to help fill missed details. In another work, **Liu et al.** (Liu et al. 2020) leverage an unsupervised learning method to create multiple freehand sketches for each RGB color image in the dataset. Thus, eliminating the necessity of sketch-image pairs dataset. Then, based on the created sketch-image pairs, an auto-encoder (AE) (Kramer 1991; Vincent et al. 2010) incorporated with a self-supervised method (Feng et al. 2019; Kolesnikov et al. 2019) and momentum mutual- information minimization loss (Liu et al. 2010) are used to disentangle the features into content and style features for both sketches and images. This step helps to generate faithful images similar to the corresponding sketches in terms of the content and consistent with the real RGB images in terms of the style. The auto-encoder consists of two independent encoders. The first encoder is a style encoder that takes a RGB color image as input and produces a style feature map. The second encoder is a content encoder that takes a sketch as input and extracts its content feature map. Then, the extracted style and content features are fed into a decoder generator to generate an image. Since auto-encoders are utilized in the model, difficulties during extracting style features are encountered, especially for fine-grained texture and unique colors. Therefore, the decoder might depend on only the content encoder by content-to-style relations. Hence, a momentum mutual-information minimization objective is used.

CoCosNet (Zhang et al. 2020), short form of CrOss-domain COrreSpondence Network, consists of two models. The first network, which is cross-domain correspondence network with weak supervision, takes an edge map and an exemplar image from distinct domains as inputs and maps them into a shared domain via domain alignment. This shared domain allows the model to represent the semantics of both domains by leveraging the feature pyramid network (Ronneberger et al. 2015; Lin and P. Doll´ar, R. Girshick, K. He, B. Hariharan, and S. Belongie. 2016) to extract local and global context in the domains. The result of the first network is a generated warped exemplar image that is semantically matched to the edge map in terms of content and semantically aligned to the exemplar image in terms of style. Then, the second network, which is translation network, generates a photo-realistic image based on the intermediate image which is a warped exemplar image by a sequence of the SPatially-Adaptive DEnormalization (SPADE) blocks (Park et al. 2019a) to maintain the structural information from the previous layers. The architecture of this model is shown in Fig. 10f.

Gao et al. (Gao et al. 2020) proposed a method to address the problem of generating images on both instance-level and scene-level from the input freehand sketches. Its first stage is foreground generation, where the model concentrates on the foreground objects to be generated similar to the specification provided in the input sketch. Thus, foreground generation takes a scene sketch and utilizes sketch segmentation method in Zou et al. (2019) to locate and recognize the instances in the scene sketch, resulting in generating foreground image for each instance. After all foreground objects are generated, the

second stage, background generation, commences. In this stage, the model focuses on the background scene by incorporating pix2pix model (Isola et al. 2016) and the generated foreground objects produced in the first stage to produce the background. Therefore, background generation takes a background sketch with the generated foreground image as input and generates the output image. Additionally, in this paper, SketchyCOCO dataset, which is a large-scale dataset, is introduced relying on MS COCO-Stuff (Caesar et al. 2018) to evaluate the proposed model. The limitation of this model is that the adopted segmentation method (Zou et al. 2019) might fail to detect some instances in the sketches when the instances are too simple and abstract. **Osahor et al.** (Osahor et al. 2020) propose a method that generates multiple images for each single input sketch through GAN. While the input sketch is a human face sketch, the synthetic output images are human face images generated with different target attributes, *i.e.*, gender, age, and hair color. A single generator is utilized which incorporates a quality guided and an identity preserving networks. To enhance the quality and minimize the dissimilarity between the generated image and its corresponding original image in terms of latent space embedding, a quality guided encoder is used. Moreover, an identity preserving network is incorporated to preserve the biometric identity of the generated image during the training through the DeepFace pretrained model (Taigman et al. 2014). A hybrid discriminator is leveraged to predict different target attributes during the attribute classification process to generate different images with different set of attributes. Therefore, the model is able to synthesize photo-realistic images with various unique attributes while maintaining the generated images' identity. To tackle the problem of generating images from badly drawn sketches, **Lu et al.** (Lu et al. 2018) propose a contextual GAN-based model. This model allows to synthesize realistic images since it follows the sparse input content while enabling some deviation in the object shape. Thus, the input sketch is considered as a weak constraint, and the problem is solved as an image completion problem. Specifically, it learns the joint distribution of the input sketch and its relevant real image utilizing joint images. However, this model requires a big dataset of sketch-image pairs. Another drawback with this model is that it needs to train each class individually, which makes it harder with a large number of objects. Due to the difficulty of acquiring large sketch-image pairs that are needed for the model training, **Li et al.** (Li et al. 2021) propose a stages semi-supervised GAN- based sketch-to-image synthesis model. In particular, this model consists of a two-stage sketch-to-image synthesis. In the first stage, the input is the class label along with random noise. This stage produces common information for each input label which is learned through cGAN (Mirza and Osindero 2014) (see Sect. 2.1.3 for details). Furthermore, it generates an image from the mid- level features such as the objects' texture and the background. In the second stage, a synthetic image is generated by using the prior common information and the input sketch. Another cGAN is incorporated in the second stage. In- deed, the network architecture of the second stage follows the SketchyGAN (Chen and Hays 2018) structure.

2.3 Other-to-image synthesis models

Two types are discussed in this section. First, image-to-image synthesis refers to a conditioned synthesis task that translates an input image to an output image, where input and output images are from different domains. Some features of the input image are changed to produce the image from the other domain; however, the content is untouched and still the same. The second type is speech-to-image synthesis, where the input is audio, and the output is a corresponding image that semantically consistent with the speech. Recently,

many researchers have been directed to speech-to-image synthesis since some languages lack the written form. Therefore, it is impossible to generate an image from a speech directly without using any text information. Some studies in the past generate images from speech through two stages. First, the speech is converted into text. Then via text-to-image synthesis models, the image is generated. This task seems trivial from the human perception perspective, where as human, we can easily correlate between sound and appearance. Nonetheless, this task is a challenging task for machines due to the heterogeneity between audio and image domains.

2.3.1 Image-to-image synthesis models

Recently, much research has been conducted in the field of image-to-image translation, and great progress has been accomplished. The aim is to map an input image to an output image, where each image is in a distinct domain. Therefore, by changing some properties of the input image, such as style, the output image is generated while maintaining the content. Many GAN-based image-to-image synthesis methods have been introduced, where the achieved results were promising.

DC-GAN (Radford et al. 2015) incorporates a deep convolutional generative adversarial network (DC-GAN) to reduce the training instability problem in GAN model and produce better results. DC-GAN replaces any pooling layers with fractional-strided convolution layers or strided convolution layers in the generator and the discriminator, respectively. This replacement enables the generator and the discriminator to learn their own spatial upsampling and downsampling, respectively. It also removes fully connected hidden layers from the network depth. Additionally, it uses a batch normalization (Ioffe and Szegedy 2015) in both the generator and the discriminator to provide a further level of stabilization during the learning process by normalizing the input to a unit variance and a zero mean. DC-GAN model generates reasonable images, but it still suffers from the mode collapse problem, which results from a limited variety of samples produced by the generator (see Sect. 2.1.3 for details). This affects the discriminator's performance so that the discriminator follows the simplest path by rejecting the generator output instead of attempting to learn from the samples (Jinzhen et al. 2022; "Common problems", Google Developers. 2023). Meanwhile, **W-GAN** (Arjovsky et al. 2017) is an extension of GAN to improve the training stability, reduce mode collapse, and produce a loss function that better correlates with generated image quality. In the basic GAN, the discriminator attempts to classify the generated images into real or fake. However, with W-GAN, the discriminator scores the realness or fakeness of the generated images. This feature is based on the fact that the generator attempts to minimize the distance between the input training data and the generated samples, so it extends DC-GAN with minor changes by incorporating the earth mover's distance (EMD). However, as reported by Gulrajani et al. (Gulrajani et al. 2017), W-GAN sometimes generates low-quality images or fails to converge because of applying a Lipschitz constraint on the critic as a weight clipping. Thus, Gulrajani et al. (Gulrajani et al. 2017) propose an alternative way by introducing a penalty term with loss function with regards to the input. The penalty term is the gradient penalty which is a soft version of the Lipschitz constraint. In WGAN-GP (Gulrajani et al. 2017), the weight clipping is replaced with a constraint on the gradient norm of the critic to achieve Lipschitz continuity. This alternative approach (Gulrajani et al. 2017) results in more stability during training and approximately no need to tune any hyperparameters in its.

framework. Additionally, it produces high-quality synthetic images.

In cGAN (Mirza and Osindero 2014), which is discussed in Sect. 2.1.3, a noise vector along with an additional condition extension, *i.e.*, class labels, descriptive tags, attributes, or data from other modalities, are fed into the generator to produce an image. Then, the discriminator takes the real original and generated images along with the auxiliary information to differentiate between real and fake images. However, **Invertible Conditional GAN (IcGAN)** (Failed 2016) is used to overcome the problem of lacking the inference technique, where the input image suffers from finding the corresponding latent representation. It extends cGAN (Mirza and Osindero 2014) by incorporating an additional encoder with a cGAN to invert the mapping of a cGAN. Therefore, not only the conditional representation is fed, but also the real image is encoded into a high-feature latent representation which is fed to the generator network as well. This allows the model to apply different modifications and editing operations on the real image by changing the conditional attributes. Thus, in IcGAN (Failed 2016), the encoder takes the real original image as an input and returns its compressed latent representation along with the conditional vector. Then, the generator in cGAN takes the latent representation along with the conditional information obtained by the encoder as inputs and produces a reconstructed modified image based on the conditional information.

Progressive GAN (Karras et al. 2017) has both the generator and the discriminator with the same general structure and then they grow progressively. This means that it starts with a low-resolution image and progressively adds new convolutional layers to both the generator and the discriminator to help in increasing the generated image size produced by the generator and increasing the size of the input to the discriminator, as shown in Fig. 11a. As a result, the progressively GANs growing method enhances the training stability by speeding up the training process, leading to generate large high-quality images of size 1024×1024. However, it sometimes fails to produce realistic images.

Pix2pix (Isola et al. 2016) leverages conditional generative adversarial networks (cGANs) (Mirza and Osindero 2014) because the learning is conditioned on the input which makes it suitable to translate an input image of high resolution to an output image of high resolution. Its architecture follows deep convolutional generative adversarial networks (DC-GANs) (Radford et al. 2015) with some modifications (see Sect. 2.3.1 for details). The first change is that both generator and discriminator utilize modules of convolution-BatchNorm-ReLu (Ioffe and Szegedy 2015). The second change is that the generator has additional skip connections following the shape of U-Net (Ronneberger et al. 2015). These skip connections are added between layers, in particular, between deep layers and shallow layers. The discriminator adopts PatchGAN (Li and Wand 2016), which penalizes structure at the patch scale, so for each patch in an image, the discriminator attempts to classify it in terms of real or fake. This helps the model to run faster since with smaller path, fewer parameters are computed. Pix2pix shows an efficient performance for various image-to-image translation tasks, *i.e.*, label maps to images, edge maps to images, and day to night. However, it requires a paired training dataset, meaning that the dataset should contain the input images and the corresponding output images after the translation process.

CycleGAN (Zhu et al. 2017) (cycle-consistent GAN) uses unpaired images for the image-to-image synthesis task. It is composed of two independent networks (Fig. 11b). The generator of the first network attempts to map an image from a source domain X to a target domain Y, *i.e.*, summer to winter. Since the translation between one domain to another in the absence of paired data is extremely restricted, another network is trained to inverse the mapping from the target domain Y to the source domain X, *i.e.*, winter to summer. The structure of both generators is based on (Johnson et al. 2016) because of its effectiveness in style transfer. Like pix2pix (Isola et al. 2016), both discriminators leverage PatchGAN (Li

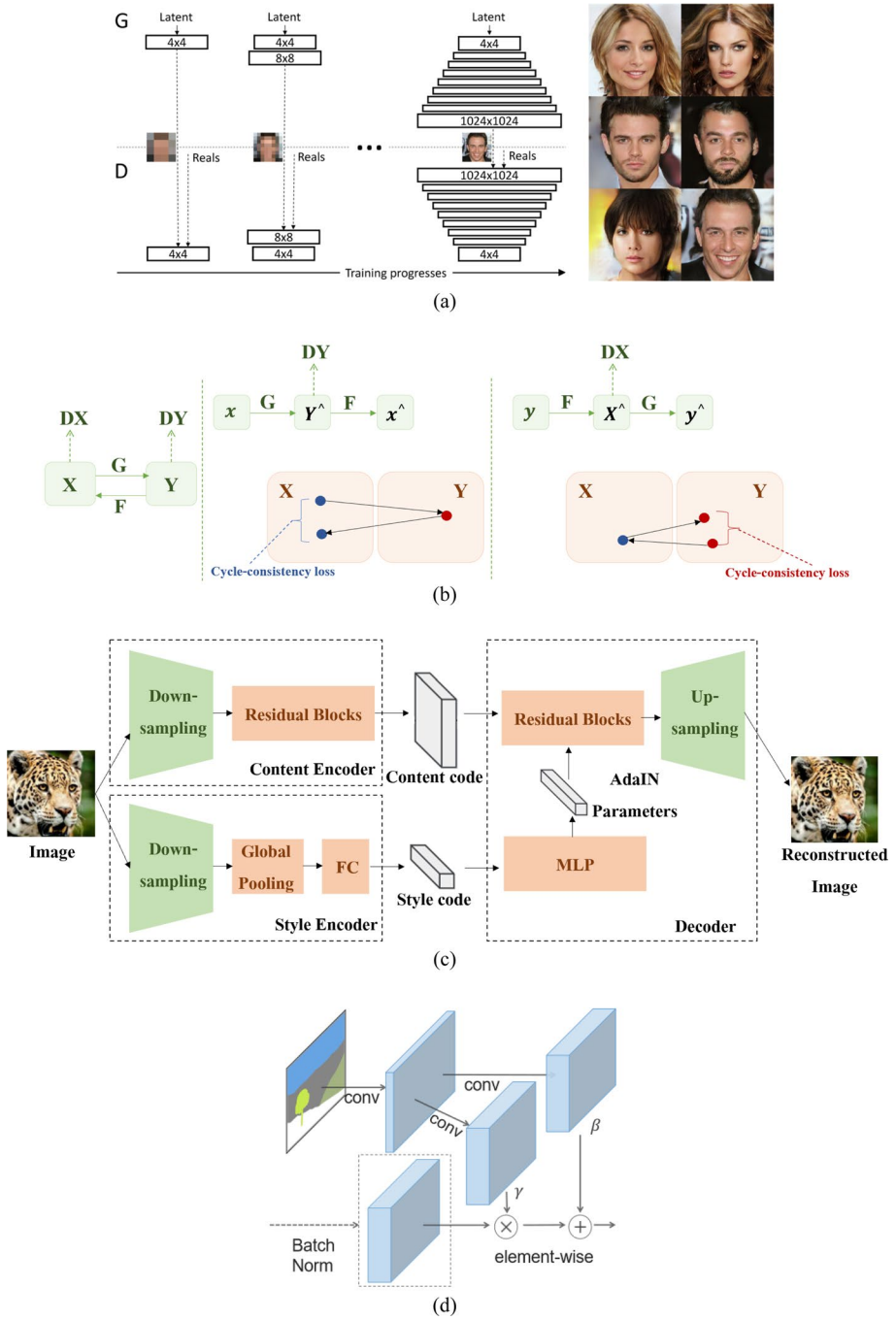

Fig. 11 **a** An overview of Progressive GAN model (Karras et al. 2017) **b** The mapping cycle in CycleGAN. **c** The architecture of the two autoencoders in MUNIT model. **d** Spatially-adaptive normalization in SPADE model, where this figure was taken from the original paper (Park et al. 2019a)

and Wand 2016). To stabilize the training process, least square function is adopted rather than a log function in computing adversarial losses in both networks, following LSGAN (Mao et al. 2016). This model is effective and applied to several image-to-image translation task, such as label maps to images, aerial to maps, edge maps to images, season transfer and style transfer. However, when the mapping requires geometric changes, this model fails.

MUNIT (Huang et al. 2018a) is a multimodal unsupervised image-to-image translation relying on the assumption that the image latent space is decomposed into a content space that is domain-invariant and a style space that captures properties of domain-specific. In addition, the two domains of images have the same content distribution, but with diverse style distributions. Therefore, it consists of two autoencoders, where each autoencoder represents different latent codes, one encodes the image content into a content code, and the other encodes the style into a style code. This allows for multimodal image generation. While content code represents the spatial structure based on the input image that should be preserved during the translation process, style code encodes the rendering of the structure that is not represented in the input image. To translate a source image to a target domain and maintain the diversity and multimodal of the outputs, the content code is recombined with various style codes sampled from the style space of the target domain. Moreover, the user can control the target style by providing a style image example. This enables to generate a high-quality image while producing image variations. Fig. 11c illustrates the architecture of the two autoencoders. The content encoder contains multiple convolutional layers, followed by residual blocks. The style encoder consists of multiple convolutional layers, followed by a global average pooling layer and a fully connected layer. The decoder leverages multilayer perceptron MLP to obtain Adaptive Instance Normalization (AdaIN) parameters (Huang and Belongie 2017) from the style code. Then, the content code is recombined with AdaIN layers via residual blocks. In the end, the combination of content and style codes is decoded to image space through up-sampling and convolutional layers.

SPADE (Park et al. 2019a) was built upon pix2pix model (Isola et al. 2016), where pix2pix model is made of convolutional, normalization, and nonlinearity layers stacking on top of each other. Since pix2pix uses normalization layers, these layers impact the semantic information because they tend to wash away the semantic information contained in the input segmentation map. To tackle this issue, SPADE was proposed, where it uses spatially-adaptive normalization, a conditional normalization layer similar to Batch Normalization (Ioffe and Szegedy 2015), where the activation is normalized and then modulated with learned parameters in an element-wise manner, as shown in Fig. 11d. Spatially-adaptive normalization is used to modulate the activations in normalization layers via a spatially-adaptive learned transformations. Hence, the semantic information is propagated through the network, leading to synthesize a photorealistic colored image from a semantic segmentation map. Furthermore, the user can control not only the style but also the semantics of the image in order to generate high-quality and diverse images.

TransGaGa (Wu et al. 2019) is a geometry-aware disentangle-and-translate model used for unsupervised image-to-image translation while maintaining the shape variations between domains. It extends CycleGAN (Zhu et al. 2017) to translate more complex objects. Rather than learning the translation on the image latent space directly, the learned transition is based on a Cartesian product of geometry structure and appearance style spaces, where the image latent space is disentangled into a geometry space and an appearance space separately. This enables the model to be decomposed into two sub-models to improve the performance and address complex image-to-image translation. To separate the image space into geometry and appearance spaces, a conditional variational autoencoder

(VAE) is applied in each domain to learn separate but complementary representations of geometry and appearance. Therefore, it has the ability to translate a com- plex image with near-rigid or non-rigid objects to high-quality images in the target domain. Different appearance references can be used as input examples to the model to enhance the diversity and multimodal outputs.

RelGAN (Lin et al. 2019) is a multi-domain image generation relying on relative attributes rather than target attributes, where previous models that take some target attributes as input fail to generate fine-grained images. This limitation is based on training the model on binary-valued attributes, which leads to an unrealistic generated image due to the lack of fine-grained control. Thus, to overcome this issue, RelGAN is trained on real-values relative attributes with auxiliary discriminators. It takes the relative attributes which describe the required change on chosen attributes along with the input image as inputs. Then, based on the chosen attributes to be changed, the model changes these particular properties of interest in the generated image successively while preserving other properties unchanged, leading to enable fine-grained control over each attribute (*i.e.*, the percentage of black hair color). RelGAN consists of one generator (G) and three discriminators (D_{Real}, D_{Match}, and D_{Interp}). The three discriminators guide the generator to learn to produce not only realistic images D_{Real}, but also precise generation in terms of relative attributes D_{Match} and naturalistic interpolations D_{Interp}. **OASIS** (Sushko and E. Sch¨onfeld, D. Zhang, J. Gall, B. Schiele, and A. Khoreva. 2020) leverages a modified version of GAN, where the discriminator is a semantic segmentation network. Hence, the semantic label maps are used as ground truth images during the training process. U-Net (Ronneberger et al. 2015), which consists of an encoder-decoder network linked through skip connections, is used as a backbone in the architecture of the discriminator. This leads to synthesizing images with better alignment to their corresponding semantic label maps. Additionally, LabelMix regularization is used to aid the discriminator in concentrating more on the content and structural differences between synthetic and original images. As a result, due to re-designing the discriminator, the generator is also re-designed to enable multi-modal synthesis via 3D noise sampling, leading to better diversity in the synthesized images.

While SPADE (Park et al. 2019a) uses a spatially-adaptive normalization layer to propagate the semantic information throughout the network, which helps in generating photo-realistic images from their corresponding semantic label maps, it is restricted to generate only one style for each output image. To tackle this problem, **SEAN** (Zhu et al. 2020) is proposed to generate different styles for each image. This could help in synthesizing better images in terms of quality as well. In particular, SEAN (Zhu et al. 2020) uses a semantic region-adaptive normalization layer. This layer helps to achieve different styles of each region separately. The SEAN generator is a modified version of SPADE (Park et al. 2019a) generator, where SEAN blocks are inserted. For each SEAN block, two inputs are given. The style codes' set for particular regions and the semantic mask, which specifies the areas of interest to apply the code, are provided as inputs. The training process is based on image reconstruction task to generate the image by adding-up each image region separately. The total loss is a composition of three losses (conditional adversarial loss, feature matching loss (Wang et al. 2017) and perceptual loss (Johnson et al. 2016)).

PISE (Zhang et al. 2021b), the short form of Person Image Synthesis and Editing, proposes a novel model to enable transferring new pose or texture to a person image. Furthermore, it allows for region editing. Rather than learning the mapping from the source image to the target image directly, PISE (Zhang et al. 2021b) uses a human parsing map as an intermediate output. This intermediate output depicts the shape of clothing. To disengage the shape and style of clothing, a joint global and local per-region encoding and

normalization are used. Specifically, the local feature of the relative area is leveraged for visible regions in the source image. Meanwhile, the global feature of the source image is incorporated for invisible regions in the source image while visible regions in the target image. These features help to predict the reasonable style of clothing. Hence, based on the human parsing map and texture control, the shape and the style of clothing are decoupled. This leads to ease the editing process. Moreover, a spatial- aware normalization is introduced to maintain the spatial context relationship in the source image and transfer it to the synthesized image. At the end, the synthesized target feature is fed into a decoder to generate the final image.

2.3.2 Speech-to-image synthesis models

Audio-visual cross-modality synthesis has recently drawn significant consideration. The ultimate goal is to generate an image from audio or vice versa. The mutual relationship between sound and appearance is easy to perceive as humans. However, it is a challenging task for machines to correlate sound and appearance because of the heterogeneity between these two domains. Speech-to-image synthesis is the task that takes audio as input and produces a corresponding visual image. Since the invention of GANs, much research has been conducted to translate sounds into images even with unseen or unheard data. This speech-to-image generation task gains lots of attention since it can be applied in numerous different disciplines, *i.e.*, neurology (Stein and Meredith 1993), psychology (Davenport et al. 1973; Vroomen and Gelder 2000), human–computer interaction (Tanveer et al. 2015), and multimedia analysis (Feng et al. 2014; Pereira et al. 2014).

Chen et al. (Chen et al. 2017b) attempt to discover cross-modal audio-visual generation leveraging conditional GANs to generate images conditioned on sounds or produce sounds conditioned on images. In this paper, two models are introduced separately for the generation of musical performances. The first model is a sound-to-image (S2I) network to generate images from sounds. The second model is an image-to-sound (I2S) network to generate sounds from images. Each network consists of three components which are an encoder network to encode sounds or images, a generator network, and a discriminator network. The encoder in the S2I network translates raw wave-sound into the time–frequency domain which is fed into CNNs. For the image encoder in the I2S network, CNN is used to encode the visual features. Both generators and discriminators in I2S and S2I are based on GAN-CLS (Reed et al. 2016a) (see Sect. 2.1.3 for details) with minor modifications to handle audio-visual cross modality. Since this study is the first study of cross-modal audio-visual generation, two datasets (Sub-URMP and INIS) are created from videos containing sound-image pairs of musical performances of different instruments. The Sub-URMP dataset contains image-sound pairs extracted from 107 videos of 13 types of instruments in the University of Rochester Musical Performance (URMP) dataset (Li et al. 2016) to include only a single-instrument musical performance. 17,555 images are extracted, and each is paired with a half-second long sound clip. The INIS dataset consists of the ImageNet dataset (Deng et al. 2009), where only five music instruments are considered which are drum, saxophone, piano, guitar, and violin. Therefore, each image from the subset of ImageNet is paired with a short sound clip of the performance of the counterpart instrument. One major limitation is that the mutual synthesis process depends on two separate models; hence, end-to-end training is not possible.

CMCGAN (Hao et al. 2018) is introduced to combine both S2I and I2S networks into one by taking into account a cross-modality cyclic generative adversarial network

(CMCGAN) to tackle the problem of cross-modal audio-visual mutual synthesis. CMCGAN consists of four components organized in a cycle architecture, following cyclic consistency principle inspired by CycleGAN (Zhu et al. 2017). Unlike CycleGAN (Zhu et al. 2017) which is discussed in Sect. 2.3.1, CMCGAN provides a latent vector to handle asymmetry in terms of structure and dimension among various modalities. Moreover, a joint corresponding adversarial loss is introduced to unify the mutual synthesis of multi-modal in one framework, leading to not only checking the realism of the generated outputs, but also checking the similarity between two different modalities sounds and images. Furthermore, a consistency loss is incorporated to produce plausible sounds and images. With regards to the four components, the first subnetwork is audio-to-visual A2V. In this subnetwork, the raw soundwave is encoded into its Log-amplitude of Mel-Spectrum LMS and then fed into CNN. After that, the embedding vector that contains the extracted features and latent vector is decoded via CNN. The second subnetwork is visual-to-audio V2A. The V2A is similar to A2V sub-network, where the image is encoded first and then decoded to map it to sound. The third subnetwork is audio-to-audio A2A. It is similar to A2V subnetwork; the difference is that both the encoder and the decoder produce a sound, where it takes a sound LMS as input and generates a sound LMS as output. The fourth subnetwork is visual-to-visual V2V, where it is similar to A2V but contains an image encoder and an image decoder. This subnetwork takes an image as an input and generate an image as an output. Later, **Li et al.** (Li et al. 2020b) propose a method to translate speech signals into visual signals directly without transcription phase. A speech encoder is introduced to learn the embedding features of speech signals, where it is trained with a pre-trained image encoder through teacher-student learning technique to generalize better and generate images for unseen visual or unheard speech signals. Specifically, the speech encoder is used to extract a low-dimensional features from the speech signal, where raw speech signal is first converted into a time frequency spectrogram. Then, the time frequency is encoded via speech encoder into a low-

dimensional feature. The speech encoder consists of a CNN inserted before a recurrent neural network (RNN) to reduce the length of speech signal. Then, the extracted embedding features are fed into cGAN (Mirza and Osindero 2014) to generate the corresponding image that aligns with the semantic information of the speech. To generate high-quality images restricted on the extracted features from speech, stack of GANs is leveraged, in particular, three GANs are used to generate images of size 256×256.

S2IGAN (Wang et al. 2021) is proposed to convert speech to image directly with two components. The first component is a speech embedding network (SEN) that is used to learn speech embeddings, which is a combination of speech and image features. In particular, SEN consists of a speech encoder that is made of CNN with bidirectional gated recurrent units (GRU) (Cho and B. Merri¨enboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio. 2014) and an image encoder of pretrained Inception-V3 on ImageNet (Deng et al. 2009) to learn the features of the corresponding images. The purpose of using a speech encoder along with an image encoder is to embed speech and images into an embedding space so that image features in the embedding space can be used to train the speech encoder in a supervised learning strategy. To minimize the difference between image-speech pairs, a matching loss is incorporated. The generated images are high-quality and consistent in terms of the semantic information with the corresponding speech.

3 Datasets

To produce effective and successful models, a benchmarked dataset plays a significant role. In image generation, not only the quality and diversity of the dataset matter but also the size of the dataset aids the model in success. There are numerous benchmarked datasets available for use, where these datasets differ from each other based on the level of scene complexity. In this section, concise details of the benchmarked datasets for image generation are introduced. While Fig. 12 shows samples of the reviewed datasets, Table 2 illustrates the size of each reviewed dataset along with the image synthesis tasks in.

which the dataset is used, where T2I, S2I, I2I, and A2I are text-to-image, sketch-to-image, image- to-image, and audio-to-image, respectively. Moreover, Table 3 demonstrates an overall view of image generation models that leverage the discussed datasets in this section.

3.1 Object-human datasets

Modified National Institute of Standards and Technology (MNIST) (Deng 2012) is a dataset of 10 handwritten digits 0–9. By remixing the samples from NIST datasets (Grother 1995), MNIST was created to be well-suited for artificial intelligence applications. In MNIST, half of the training and testing were from NIST training set, while the other halves of both training and testing were from NIST testing set. Indeed, MNIST consists of 60,000 and 10,000 for training and testing images, respectively. The size of each image is 28×28 grayscale image. It is one of the simplest datasets that is used commonly in the fields of image processing, pattern recognition, and machine learning. Meanwhile, **Caltech-UCSD Birds-200–2011 (CUB-200–2011) Dataset** (Wah et al. 2011) is an extension of CUB-200 dataset (Welinder et al. 2010a), where for each class not only the number of images was approximately doubled, but also new part locations were discovered and annotated. Images were taken via Flickr image search and then filtered by many users of Amazon Mechanical Turk (Welinder et al. 2010b). It contains 11,788 images (5,994 for training and 5,794 for testing) classified into 200 classes of bird species. Images are annotated by part locations, attribute labels, and bounding boxes. Roughly 1 subcategory label, 15-part locations, 312 binary attributes, and 1 bounding box are annotated for each image. CUB-200–2011 was expanded by gathering fine-grained natural language descriptions. For each image, ten sentences are collected through Amazon Mechanical Turk (Welinder et al. 2010b), which requires at least 10 words per sentence. **Oxford 102 Flower Dataset** (Nilsback and Zisserman 2008) consists of 102 flower categories. The chosen flowers are commonly prospered in the United Kingdom. The number of images per category is between 40 and 258 images. The dataset was split into a training, a validation, and a testing set. 2040 images in the training and validation sets were split evenly, where 10 images per category were assigned. The test set comprises 6149 images, where at least 20 images were assigned for each class. The variations in scale, pose, and light are not only among categories but also within each category.

CIFAR Datasets (Krizhevsky 2009) including CIFAR 10 and CIFAR 100 are a subset of Tiny Images dataset (Torralba et al. 2008). SIFAR 10 consists of 60,000 images categorized into 10 classes, where each class comprises 6,000 images. The 10 classes represent airplanes, cars, birds, cats, deer, dogs, frogs, horses, ships, and trucks. The ratio of training and testing sets is 5:1, meaning 5,000 and 10,000 images for the training and testing set,

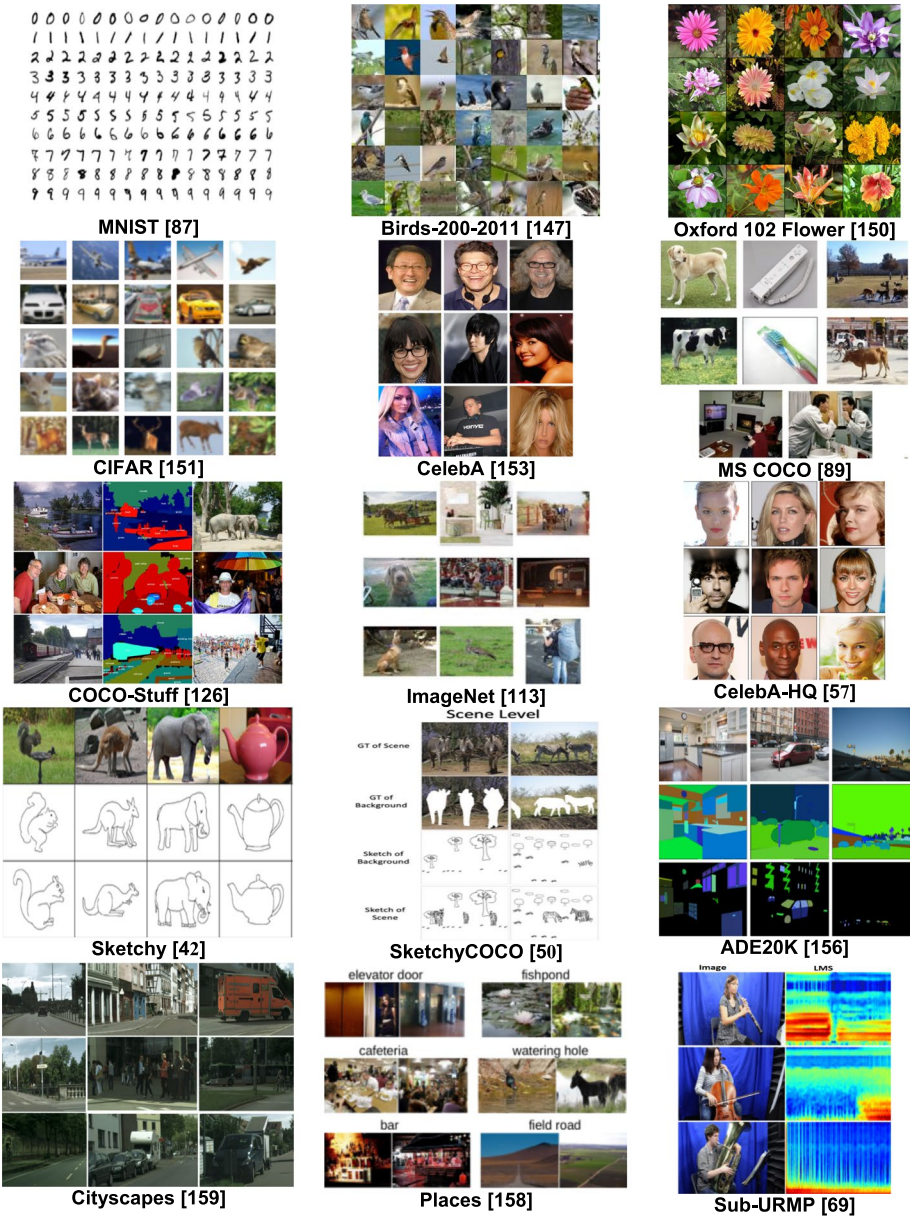

Fig. 12 Samples of reviewed datasets for image generation

respectively. The training set was divided into five batches evenly; thus, 10,000 images per batch. The images are 32×32 color images, and the categories are totally exclusive without any overlapping between classes even between automobiles and trucks classes. CIFAR 100 is similar to CIFAR 10 except that the number of classes is 100 instead of 10. Therefore, the training and testing set consist of 500 and 100 images, respectively, where 600 images

Table 2 The size of image generation datasets mentioned in this review

Category	Dataset	Image Synthesis Task	Size
Object-human datasets	MNIST	T2I	70,000
	Caltech-UCSD Birds-200–2011	T2I, S2I, A2I	11,788
	Oxford 102 Flower	T2I, A2I	6149
	CIFAR	T2I	60,000
	CelebA	T2I, I2I, S2I	202,599
	MS COCO	T2I	123,287
	COCO-Stuff	T2I, I2I	164,000
	ImageNet	T2I	14,197,122
	CelebA-HQ	T2I, I2I, S2I	30,000
Sketch datasets	Sketchy	S2I	87,971
	SketchyCOCO	S2I	20,198
Scene datasets	ADE20K	I2I, S2I	27,274
	Cityscapes	I2I	25,000
	Places	I2I, A2I	10,000,000
Sound datasets	Sub-URMP	A2I	17,555

were assigned per class. The 100 categories in CIFAR 100 were classified into 20 super-categories. Hence, two labels were assigned to each image, fine and coarse labels. The fine label is the subclass label to which the image belongs (100 classes). The coarse label is the superclass to which the image belongs (20 classes). **CelebA Dataset** (Liu et al. 2014) contains 202,599 large-scale face images of 10,177 celebrities. Each image is of the size 178×218 with complex backgrounds and is annotated with 40 binary labels describing facial attributes such as gender, age, and hair color and 5 landmark locations. The variations in images are based on pose, scale, diversity, and background clutter. This dataset is widely used in face recognition, face detection, landmark localization, face editing, and face synthesis.

MS-COCO Dataset (Lin et al. 2014) is a complex large-scale dataset published by Microsoft, where each image contains more than one instance. COCO contains 123,287 images classified into 80 classes, where 118,287 images are for the training set and 5,000 images are for the validation set. The test set comprises about 41,000 images. The 80 classes belong to things that are individual objects, *i.e.*, persons, airplanes, tables, bananas, etc. Due to its annotation, COCO is popularly leveraged for various computer vision and machine learning applications such as classification, localization, key-point detection, segmentation, captioning, and synthesis tasks. **COCO-Stuff Dataset** (Caesar et al. 2018) is an extension of MS COCO (Lin et al. 2014), where all 164,000 images from COCO 2017 (Lin et al. 2014) were augmented with dense pixel-level stuff annotations of 91 classes. Therefore, this dataset is a good choice for scene understanding tasks, such as object detection, semantic segmentation, and captioning. Similar to COCO (Lin et al. 2014) each image consists of 5 captions. It contains 80 thing classes from COCO (Lin et al. 2014) plus 91 stuff classes along with 1 class for class'unlabeled'. These classes were grouped into 27 super-categories which in turn were grouped into either indoor or outdoor. Then the super-classes were grouped into two classes which are either things or stuff.

ImageNet Dataset (Deng et al. 2009) is a fine-grained large dataset that consists of 14,197,122 images organized according to the WordNet hierarchy (Miller et al. 1990) for

Table 3 Image generation models leverage the datasets mentioned in this review. X denotes the usage of a dataset by an image synthesis method

	MNIST (Deng 2012)	CUB-Birds-200–2011 (Wah et al. 2011)	Oxford 102 Flower (Nilsback and Zisserman 2008)	CIFAR (Krizhevsky 2009)	CelebA (Liu et al. 2014)	MS COCO (Lin et al. 2014)	COCO-Stuff (Caesar et al. 2018)	ImageNet (Deng et al. 2009)	CelebA-HQ (Karras et al. 2017)	Sketchy (Sangkloy et al. 2016a)	Sketchy-COCO (Gao et al. 2020)	ADE20K (Zhou et al. 2016)	Cityscapes (Cordts et al. 2016)	Places (Zhou et al. 2018)	SUN-URMP (Hao et al. 2018)
DRAW (Gregor et al. 2015)	X			X											
Conditional AlignDRAW (Mansimov et al. 2015)	X					X									
Cai et al. (Cai et al. 2017)	X				X										
Conditional GAN cGAN (Mirza and Osindero 2014)	X														
Invertible Conditional GAN IcGAN (Failed 2016)	X				X										
DC-GAN (Radford et al. 2015)	X							X							
GAN-INT-CLS (Reed et al. 2016a)		X	X			X									
AC-GAN (Odena et al. 2016)				X				X							
StackGAN (Zhang et al. 2016)		X	X			X									
ChatPainter (Sharma et al. 2018)						X									

Table 3 (continued)

	MNIST (Deng 2012)	CUB-Birds-200–2011 (Wah et al. 2011)	Oxford 102 Flower (Nilsback and Zisserman 2008)	CIFAR (Krizhevsky 2009)	CelebA (Liu et al. 2014)	MS COCO (Lin et al. 2014)	COCO-Stuff (Caesar et al. 2018)	ImageNet (Deng et al. 2009)	CelebA-HQ (Karras et al. 2017)	Sketchy (Sangkloy et al. 2016a)	Sketchy-COCO (Gao et al. 2020)	ADE20K (Zhou et al. 2016)	Cityscapes (Cordts et al. 2016)	Places (Zhou et al. 2018)	SubURMP (Hao et al. 2018)
StackGAN++(Zhang et al. 2017)		X	X			X		X							
Progressive GAN (Karras et al. 2017)				X	X				X						
MC-GAN (Park et al. 2018)		X	X												
AttnGAN (Xu et al. 2017)		X				X									
MirrorGAN (Qiao et al. 2019)		X				X									
SegAttnGAN (Gou et al. 2020)		X	X												
AGAN-CL (Wang et al. 2020)		X	X			X									
ACGAN (Li et al. 2020a)		X	X												
LD-CGAN (Gao et al. 2021)		X	X												
XMC-GAN (Zhang et al. 2021a)						X									
T2M-M2I (Baraheem and Nguyen 2020a)							X								

Table 3 (continued)

	MNIST (Deng 2012)	CUB-Birds-200–2011 (Wah et al. 2011)	Oxford 102 Flower (Nilsback and Zisserman 2008)	CIFAR (Krizhevsky 2009)	CelebA (Liu et al. 2014)	MS COCO (Lin et al. 2014)	COCO-Stuff (Caesar et al. 2018)	ImageNet (Deng et al. 2009)	CelebA-HQ (Karras et al. 2017)	Sketchy (Sangkloy et al. 2016a)	Sketchy-COCO (Gao et al. 2020)	ADE20K (Zhou et al. 2016)	Cityscapes (Cordts et al. 2016)	Places (Zhou et al. 2018)	SUB-URMP (Hao et al. 2018)
Aesthetic-Aware T2M-M2I (Baraheem and Nguyen 2020b)							X								
Sangkloy et al. (Sangkloy et al. 2016a)										X					
SketchyGAN (Chen and Hays 2018)										X					
Liu et al. (Liu et al. 2020)									X						
CoCosNet (Zhang et al. 2020)									X			X			
Gao et al. (Gao et al. 2020)											X				
Lu et al. (Lu et al. 2018)		X			X										
Li et al. (Li et al. 2021)										X					
Pix2pix (Isola et al. 2016)													X		
CycleGAN (Zhu et al. 2017)													X		
MUNIT (Huang et al. 2018a)													X		

Table 3 (continued)

	MNIST (Deng 2012)	CUB-Birds-200–2011 (Wah et al. 2011)	Oxford 102 Flower (Nilsback and Zisserman 2008)	CIFAR (Krizhevsky 2009)	CelebA (Liu et al. 2014)	MS COCO (Lin et al. 2014)	COCO-Stuff (Caesar et al. 2018)	ImageNet (Deng et al. 2009)	CelebA-HQ (Karras et al. 2017)	Sketchy (Sangkloy et al. 2016a)	Sketchy-COCO (Gao et al. 2020)	ADE20K (Zhou et al. 2016)	Cityscapes (Cordts et al. 2016)	Places (Zhou et al. 2018)	SubURMP (Hao et al. 2018)
SPADE (Park et al. 2019a)							X					X	X		
RelGAN (Lin et al. 2019)					X				X						
OASIS (Sushko and E. Sch¨onfeld, D. Zhang, J. Gall, B. Schiele, and A. Khoreva. 2020)							X					X	X		
SEAN (Zhu et al. 2020)												X	X		
Chen et al. (Chen et al. 2017b)															X
CMCGAN (Hao et al. 2018)															X
Li et al. (Li et al. 2020b)		X	X												
S2IGAN (Wang et al. 2021)		X	X											X	

supervised machine learning tasks. ImageNet is organized into 21,841 subclasses, where on average over 500 images were assigned to each subclass. This dataset was manually annotated with not only the labels in terms of the presence or absence of an object, but also annotated with bounding boxes along with class labels. The number of images with bounding boxes annotations is 1,034,908. Therefore, it is commonly used in object localization, detection, and classification tasks. Due to the use of Scale-Invariant Feature Transform SIFT (Lowe 2004) which aids in detecting local features in images, ImageNet is classified into 1,000 sub-classes with SIFT local features for about 1.2 million images.

CelebA-HQ Dataset (Karras et al. 2017) is a high-quality version of CelebA dataset (Liu et al. 2014), where it contains 30,000 large-scale face images of celebrities of size 1024×1024. Like CelebA (Liu et al. 2014), each image is annotated with 40 facial attributes. Unline CelebA (Liu et al. 2014), the dataset is centered on the facial region and has fixed size 1024×1024.

3.2 Sketch datasets

Sketchy Dataset (Sangkloy et al. 2016a) is a large-scale fine-grained dataset of sketch-image pairs categorized into 125 classes. While 100 classes of Sketchy dataset (Sangkloy et al. 2016a) are already exist in the Eitz 2012 dataset (Eitz et al. 2012), ImageNet dataset (Deng et al. 2009) classes were used. However, multiple related classes in ImageNet (Deng et al. 2009) were combined into one class to increase the diversity such as dog breeds. The images were collected first and filtered by many steps ranging from maintaining only images of one bounding boxes to eliminating degraded and ambiguous contents. After collecting images, crowd workers were asked to sketch the images of objects without using directly the images but sketching by their imagination similar sketches to the images. Therefore, the dataset consists of 75,471 human sketches and 12,500 objects.

SketchyCOCO Dataset (Gao et al. 2020) comprises two parts: object-level data and scene-level data. The object-level data consists of 20,198 images spanning 14 classes, where 18,869 and 1,329 for training and validation, respectively. It contains triplets of foreground sketch, foreground image, and foreground edge map. However, pairs of background sketch-background image cover only 3 classes, where the number of total images is 27,683 (22,171 for training and 5,512 for validation). The scene-level data consists of 14,081 images, where 11,265 and 2,816 for training and validation, respectively. This part contains pairs of foreground image with background sketch and scene image. Additionally, it contains pairs of scene sketch and scene image with size of 14,081 images (11,265 for training and 2,816 for validation). Moreover, it provides the segmentation ground truth for the scene sketches of 14,081 images in total, where 11,265 for training set and 2,816 for validation set. For some scene images in validation set, the images were taken from COCO-Stuff dataset (Caesar et al. 2018) and specifically from the training set to increase the number of validation images.

3.3 Scene datasets

ADE20K Dataset (Zhou et al. 2016) is a semantic segmentation dataset that is annotated with pixel-level objects and object parts labels covering both indoor and outdoor images. It contains images, object segmentations, and parts segmentations. Some categories could be objects and parts at the same time, *i.e.*, a door can be an object in an indoor image or can be a part of a car object. In total, it is composed of about 27,274 scene-centric

images spanning 365 various scenes, where the scene classes from the Scene UNderstanding (SUN) (Xiao et al. 2010) and Places (Zhou et al. 2018) datasets. 25,574 for the training set with full annotation for objects and many of the parts as well. 2,000 for the validation set with full annotation for objects and parts. While 707,868 distinct objects from 3,688 classes were provided along with their WordNet (Miller et al. 1990) hierarchy and definition, 193,238 object parts and parts of parts were annotated. It contains not only objects, such as person, bed, and car but also stuff like grass, road, and sky. Furthermore, polygon annotations were offered for scene segmentation.

Cityscapes Dataset (Cordts et al. 2016) is a large-scale dataset concentrating on urban street scenes' semantic understanding. It provides semantic, object-level, and dense pixel-level annotations spanning 30 classes grouped into 8 superclasses (nature, sky, flat surfaces, constructions, objects, humans, vehicles, and void). It includes approximately 5,000 fine annotated images and about 20,000 coarse annotated images. Images were captured in 50 different cities during the daytimes and good or at least medium weather conditions over several months (spring, summer, and fall). In addition, some researchers augment the images with other weather conditions, such as fog and rain along with specifying the bounding boxes of people in the annotation as an extension. In the beginning, this dataset was captured as video, but then the frames were manually selected so that they vary in terms of background and layout.

Places Dataset (Zhou et al. 2018) is a scene-centric dataset of visual environments around us in the real world. It is composed of about 10 million scene images. Each image is annotated with a category label, which includes 476 scene semantic classes and attributes, and for each class, more than 5,000 images were assigned. The scene semantic classes were inherited from the scene classes list's SUN dataset (Xiao et al. 2010) including places, scenes, and environments. Images were collected from image search engines such as Google Images, Bing Images, and Flickr through a query word based on the scene classes from the SUN dataset (Xiao et al. 2010). The images are color images of at least 200×200 resolution. It is widely used for object and scene recognition.

3.4 Sound datasets

Sub-URMP Dataset (Hao et al. 2018) is a subset of the University of Rochester Musical Performance dataset (Li et al. 2016). This subset dataset comprises of paired sounds and images for 13 music instrument classes. Each class contains various music pieces that are played by 1 to 5 individuals. Sounds are extracted from 107 videos of a single-instrument musical performance including 13 different types of instruments in the University of Rochester Musical Performance. In total, there are 17,555 images, and for each image, a half-second sound clip is paired.

4 Evaluation Metrics

To assess the success and performance of image synthesis models, many evaluation measures are utilized. These evaluation measures are categorized into two major types: qualitative and quantitative. Qualitative measures are subjective measures, where the model is evaluated based on user observations and preferences without any metrics or statistics. Qualitative evaluation of image generation concentrates on the quality of the generated image and/or the consistency between the input and output generated image through

human perception. Contrary to qualitative measures, in quantitative measures, the model is evaluated by metrics or statistics. Therefore, the model is measured using numbers, leading to more robust and reliable measures. In image generation, qualitative evaluation is often accomplished through human rank (HR). However, since qualitative measures are subjective, meaning that the evaluation might vary from one person to another due to the differences in human perception and human preferences. Furthermore, while sometimes it is difficult to find appropriate participants, a user study is usually time-consuming.

On the other hand, in image generation, there are several quantitative measures. One quantitative evaluation metric is the inception score (IS) (Salimans et al. 2016) which is used to measure the quality and diversity of the generated images. Thus, a good model should not only generate reasonable images but also diverse images. IS (Salimans et al. 2016) is a well-known metric for GANs assessment, defined as:

$$IS = \exp(E_x \mathrm{DKL}(\mathrm{p(y|x)||p(y)})) \tag{1}$$

Given a pre-trained image classifier, IS (Salimans et al. 2016) computes KL-divergence between the conditional distribution p(y|x) and the marginal distribution p(y), where x is one generated sample from the generator, and y is the label predicted via the inception model. Indeed, the higher IS, the better the model in terms of quality and diversity of images. IS correlates with human judgments in terms of image quality, but it has some drawbacks. First, due to considering only the generated images and not incorporating the real ones, IS fails to determine the generator's efficiency in GANs models. Moreover, it cannot determine if the generated images are well-aligned with the given input or not.

In addition, it is less informative because it easily overfits.

Another quantitative measure used to evaluate the image synthesis models based on the image quality is Fréchet Inception Distance (FID) (Heusel et al. 2017). FID considers not only the generated images but also the real ones. Thus, it computes the distance between the generated distribution $p_g(x)$ and the real distribution $p_{real}(x)$ based on the extracted visual features. In fact, it calculates multivariate Gaussian (mean m and covariance c) of the generated and real images as illustrated in Eq. (2).

$$\begin{aligned} FID\left(p_{real},\, p_g\right) &= d^2\left(\left(m_{real},\, m_g\right),\, \left(m_g,\, c_g\right)\right) \\ &= \left|\left|m_{real},\, -\, m_g\right|\right|^2 \,+\, T_r\left(c_{real} + c_g \,-\, 2\left(c_{real}\, c_g\right)\right)^{1/2} \end{aligned} \tag{2}$$

where mean and covariance of real and generated images are m_{real}, c_{real}, m_g, and c_g, respectively. In FID (Davenport et al. 1973), the lower FID value means the closer the distance between generated and real distributions, leading to better model performance and better generated images. However, since the distance between generated and real images depends on the extracted features which might be affected by artifacts, the result might be impacted even with a small artifact in the feature space.

The third quantitative measure is R-precision (Xu et al. 2017) that is used to assess the semantic consistency/similarity between the generated images and their corresponding inputs (text, sketch, another image, or audio); thus, determining how well the generated images are conditioned on the given inputs. R-precision (Xu et al. 2017) is a score that results from pre-training a convolutional neural network (CNN) and input encoder in order to make the embeddings of real images similar to the embedding of the corresponding inputs. Then, a sample of the generated images is taken along with their corresponding inputs to compute the cosine similarities between the extracted visual features from the generated images and the extracted features from the inputs. For each pair of generated

images and its corresponding input, 99 randomly sampled wrong inputs are given. The higher the value means the embedding of the generated image is most similar to the correct input; thus, the higher semantic consistency/similarity between the generated images and the inputs. One concern with R-precision is that models may already overfit to R-precision evaluation measure during training.

Multi-Scale Structural SIMilarity (MS-SSIM) (Wang et al. 2003) is another quantitative measure used to assess the quality of generated images, where it takes both the generated and real images and finds the similarity between them. In fact, the idea behind MS-SSIM is that the human visual system is extremely capable for eliciting structural information from the world around us; thus, computing the structural similarity between two images can be a good measure to perceive the quality of an image. MS-SSIM is an advance variant form of Structural Similarity Index Measure (SSIM) (Wang et al. 2004), where the similarity between two images is measured over multiple scales via multiple sub-sampling stages to incorporate details of images at various resolutions. The process starts by computing the contrast c and the structure s comparisons. This process is iteratively done through multiple sub-sampling stages, where it successively applies a low-pass filter, and then, it down-samples the image after applying the filter by a factor of 2. Subsequently, at the highest scale, the luminance comparison l is calculated. MS-SSIM defines as follows.

$$SSIM(x, y) = \left[l_M(x, y)\right]^{\alpha M} \prod_{j=1}^{M} \left[cj(x, y)\right]^{\beta j} \left[sj(x, y)\right]^{\gamma j} \tag{3}$$

where c_j and sj are contrast and structure comparisons at j-th scale. l_M is the luminance comparison at scale M, and x and y are the two images. The MS-SSIM value ranges between 0.0 and 1.0, where the higher the value the most perceptually similar image. Although MS-SSIM metric is a good method that follows the human visual system in images quality assessment, it fails sometimes to consider human perception nuances. Another perceptual similarity metric is Learned Perceptual Image Patch Similarity (LPIPS) metric (Zhang et al. 2018) that mimics the human judgment process on the similarity of two images. It measures the differences between the generated output image and its corresponding real image. In particular, it computes the distances between these two images in terms of the extracted visual features from pre-trained CNN. The highest LPIPS is the most similar image to the corresponding real image and vice versa.

As can be seen each evaluation metric has its success and failure aspects; therefore, the research community is still attempting to find a better metric. While Table 4 summarizes the reviewed evaluation measures, and the image generation models that leverage them, Table 5 shows Inception Score (IS) (Salimans et al. 2016) and Fréchet Inception Distance (FID) (Heusel et al. 2017) results of various image generation models against different datasets.

5 Challenges and future outlook

Based on the introduced image synthesis methods and their achieved results, it is clear that image synthesis task has produced promising results, but it is still challenging, and optimal ultimate solution is still uncovered. Image generation is not a trivial task because of numerous challenges that lie in multi-domain translations' challenges. Mapping from one domain

Table 4 Summary of the discussed evaluation measures along with the image generation models that use them

	IS (Salimans et al. 2016)	FID (Heusel et al. 2017)	R-precision (Xu et al. 2017)	MS-SSIM (Wang et al. 2003)	LPIPS (Zhang et al. 2018)	HR
TTP (Zhu et al. 2007)						X
AC-GAN (Odena et al. 2016)				X		
StackGAN (Zhang et al. 2016)	X					X
ChatPainter (Sharma et al. 2018)	X					
StackGAN++(Zhang et al. 2017)	X	X		X		X
Progressive GAN (Karras et al. 2017)	X			X		
AttnGAN (Xu et al. 2017)	X		X			
MirrorGAN (Qiao et al. 2019)	X		X			X
SegAttnGAN (Gou et al. 2020)	X		X			
AGAN-CL (Wang et al. 2020)	X		X			X
ACGAN (Li et al. 2020a)	X					X
LD-CGAN (Gao et al. 2021)	X					
XMC-GAN (Zhang et al. 2021a)	X	X	X			X
T2M-M2I (Baraheem and Nguyen 2020a)		X			X	X
Aesthetic-Aware T2M-M2I (Baraheem and Nguyen 2020b)					X	X
Sketch2Photo (Chen et al. 2009)						X
Auto-painter (Failed 2017)						X
SketchyGAN (Chen and Hays 2018)	X					X
Liu et al. (Liu et al. 2019)		X			X	X
Liu et al. (Liu et al. 2020)		X			X	
CoCosNet (Zhang et al. 2020)		X				X
Gao et al. (Gao et al. 2020)		X		X		X
Osahor et al. (Osahor et al. 2020)	X	X		X		
Li et al. (Li et al. 2021)	X	X				
Pix2pix (Isola et al. 2016)						X

Table 4 (continued)

	IS (Salimans et al. 2016)	FID (Heusel et al. 2017)	R-precision (Xu et al. 2017)	MS-SSIM (Wang et al. 2003)	LPIPS (Zhang et al. 2018)	HR
CycleGAN (Zhu et al. 2017)						X
MUNIT (Huang et al. 2018a)	X				X	X
SPADE (Park et al. 2019a)		X				X
TransGaGa (Wu et al. 2019)		X			X	X
RelGAN (Lin et al. 2019)		X		X		X
SEAN (Zhu et al. 2020)		X		X		
PISE (Zhang et al. 2021b)		X			X	
Chen et al. (Chen et al. 2017b)						X
Li et al. (Li et al. 2020b)	X	X				
S2IGAN (Wang et al. 2021)	X	X				

Table 5 Inception Score (IS) (Salimans et al. 2016) and Fréchet Inception Distance (FID) (Heusel et al. 2017) results of various image generation models against different datasets

Dataset	Method	Input	IS (Salimans et al. 2016)	FID (Heusel et al. 2017)
Caltech-UCSD Birds-200–2011	StackGAN++(Zhang et al. 2017)	Text	4.02 ± 0.03	20.94
	AttnGAN (Xu et al. 2017)	Text	4.36 ± 0.03	–
	MirrorGAN (Qiao et al. 2019)	Text	4.56 ± 0.05	–
	ACGAN (Li et al. 2020a)	Text	4.48 ± 0.05	–
	SegAttnGAN (Gou et al. 2020)	Text	4.82 ± 0.05	–
	Li et al. (Li et al. 2020b)	Speech	4.09 ± 0.04	18.37
	S2IGAN (Wang et al. 2021)	speech	4.29 ± 0.04	14.50
Oxford 102 Flower	StackGAN++(Zhang et al. 2017)	Text	3.35 ± 0.07	50.38
	ACGAN (Li et al. 2020a)	Text	3.98 ± 0.05	–
	SegAttnGAN (Gou et al. 2020)	Text	3.52 ± 0.09	–
	Li et al. (Li et al. 2020b)	Speech	3.23 ± 0.05	54.76
	S2IGAN (Wang et al. 2021)	Speech	3.55 ± 0.04	48.64
CelebA	AttnGAN (Xu et al. 2017)	Text	–	10.74
	RelGAN (Lin et al. 2019)	Image	–	4.68
CelebA-HQ	MUNIT (Huang et al. 2018a)	Image	–	56.8
	SPADE (Park et al. 2019a)	Image	–	31.5
	CoCosNet (Zhang et al. 2020)	Sketch	–	14.3
	Liu et al. (Liu et al. 2020)	sketch	–	13.6
Places	AttnGAN (Xu et al. 2017)	Text	4.59 ± 0.51	35.59
	Li et al. (Li et al. 2020b)	Speech	–	83.06
	S2IGAN (Wang et al. 2021)	Speech	4.04 ± 0.25	42.09
COCO-Stuff	SPADE (Park et al. 2019a)	Image	–	22.6
	Pix2pix (Isola et al. 2016)	Image/Sketch	–	111.5
	OASIS (Sushko and E. Sch¨onfeld, D. Zhang, J. Gall, B. Schiele, and A. Khoreva. 2020)	Image	–	17.0

Table 5 (continued)

Dataset	Method	Input	IS (Salimans et al. 2016)	FID (Heusel et al. 2017)
ADE20K	SPADE (Park et al. 2019a)	Image	–	33.9
	Pix2pix (Isola et al. 2016)	Image/Sketch	–	81.8
	MUNIT (Huang et al. 2018a)	Image	–	129.3
	CoCosNet (Zhang et al. 2020)	Sketch	–	26.4
	OASIS (Sushko and E. Sch¨onfeld, D. Zhang, J. Gall, B. Schiele, and A. Khoreva. 2020)	Image	–	28.3
	SEAN (Zhu et al. 2020)	Image	–	24.84
Cityscapes	SPADE (Park et al. 2019a)	Image	–	71.8
	Pix2pix (Isola et al. 2016)	Image/Sketch	–	95.0
	OASIS (Sushko and E. Sch¨onfeld, D. Zhang, J. Gall, B. Schiele, and A. Khoreva. 2020)	Image	–	47.7
	SEAN (Zhu et al. 2020)	Image	–	50.38
Sketchy	Pix2pix (Isola et al. 2016)	Sketch	3.94	–
	SketchyGAN (Chen and Hays 2018)	Sketch	7.90	–
	Li et al. (Li et al. 2021)	Sketch	5.1	231.01
SketchyCOCO	SketchyGAN (Chen and Hays 2018)	Sketch	–	137.9
	Gao et al. (Gao et al. 2020)	Sketch	–	87.6

to another different domain suffers from several difficulties. Therefore, these considerable challenges limit the ability to generate images that are highly photorealistic, naturalistic, and semantically consistent to the inputs.

One major challenge in the image synthesis is the significant differences between input domain (text, sketch, speech, or image) and output domain (corresponding image). The huge gap between two distinct domains needs to be addressed through intermediate components. These intermediate components make the image synthesis task not trivial. Some image generation models attempt to map the input to the corresponding image output directly. This approach often fails, especially with complex inputs. Therefore, the community research has shifted to map the input to its corresponding output via intermediate component(s), such as mapping an input text to a mask map which in turns is mapped into an image. Another example is translating a speech to text which in turns is mapped into an image through an intermediate component.

Another challenge is handling the semantic consistency between input and its corresponding generated output. Some image synthesis models suffer from this challenge, where the generated image output is not semantically consistent with its input. The generated output image should align and match the corresponding input by conveying similar meaning in order to succeed.

Furthermore, the realism aspect is very substantial in image generation. Not only that the smoothness and the quality of the generated images are essential, but also lacking artifact and blurriness are very important. Moreover, layout, spatial, and configuration information, *i.e.*, location, relation, size, shape, orientation and other important information are critical. This challenge often occurs with text-to-image and speech-to-image synthesis. Many text-to-image and speech-to-image generation models struggle with this aspect, leading to unnaturalistic images.

One main problem in image generation is that the model should be able to generate output images for unseen or unheard input which makes the problem even harder. Even though generative adversarial networks (GANs) help in ease this challenge to some extent, some image synthesis models still sometimes experience this problem.

Furthermore, the complexity of the inputs play an important role in the success of image generation models. With regards to simple input, such as a short text/speech with a small number of objects or coarse sketch/image, most of the image generation models work fine in generating the corresponding images. However, the problem always emerges with complex input, such as long texts or speech and fine-grained details of sketches or images from different domain.

In addition, the computation cost should be minimum while the stability of the network should be maximum. Therefore, image synthesis task is still an active open area of research, thus, the aim of this paper is to provide a comprehensive review that presents image generation models conditioned on various input domains while attempting to solve the aforementioned challenges in image synthesis.

Recently, a type of generative model has shown great success in producing good high-quality images (Ho et al. 2020; Song and Ermon 2020; Jolicoeur-Martineau and R´emi Pich´e-Taillefer 2020; Nichol and Dhariwal 2021; Sasaki et al. 2021; Muzaffer O¨ zbey, Onat Dalmaz, Salman U. H. Dar, Hasan A. Bedel, S¸aban O¨ zturk, Alper Gu¨ng¨or, and Tolga C¸ ukur. 2022; Alper Gu¨ng¨or, Salman U. H. Dar, S¸aban O¨ ztu¨rk, Yilmaz Korkmaz, Gok- berk Elmas, Muzaffer O¨ zbey, and Tolga C¸ ukur. 2022) and audio (Nanxin Chen et al. 2020; Kong et al. 2020). It is called diffusion models (Sohl-Dickstein et al. 2015) that can be another possible direction for future research in the image synthesis domain. Diffusion models (Sohl-Dickstein et al. 2015) are latent variable models

that work by slowly adding random noise to the inputs through forward diffusion steps and then learning to reverse the diffusion process to recreate the input samples from the noise. **Nichol et al.** (Nichol and Dhariwal 2021) use the Denoising Diffusion Probabilistic Models (DDPM) (Ho et al. 2020) and apply some modifications to achieve competitive log- likelihoods with high-quality images and in less forward passes. During forward noising process, **DDPM** (Ho et al. 2020) adds Gaussian noise at every timestep with fixed noising process, fixed variance. Instead of a fixed variance to either a lower or upper bound (Ho et al. 2020), Nichol et al. (Nichol and Dhariwal 2021) propose to learn a model that interpolates between the two bounds in the log domain to predict the variance. Furthermore, rather than incorporating a linear noise schedule as in Ho et al. (2020), Nichol et al. (Nichol and Dhariwal 2021) uses a different noise schedule, cosine schedule. Moreover, a small loss term is added to the objective function to improve the variational lower-bound (VLB). Thus, the log-likelihood is improved even on high-diversity datasets, such as ImageNet (Krizhevsky et al. 2017). **UNIT-DDPM** (Sasaki et al. 2021) is introduced for unpaired image-to-image translation leveraging DDPM. It learns both domains of images (source and target domains) through DDPM and concatenates images of both domains to infer the joint probability as a Markov chain. The DSM objective function conditioned on the other domain, *i.e.*, to transfer the source to target and the target to source, respectively, is minimized. To generate the target domain image based on the source domain image, the Gaussian noise along with the noisy source image which is perturbed by forward diffusion process are used to gradually synthesize the target image. **SynDiff** (Muzaffer O¨ zbey, Onat Dalmaz, Salman U. H. Dar, Hasan A. Bedel, S¸aban O¨ zturk, Alper Gu¨ng¨or, and Tolga C¸ ukur. 2022) uses a conditional diffusion process for medical image synthesis. This diffusion process progressively translates images from the source domain with noise into the target domain images. To generate accurate and high-quality efficient samples, an adversarial projector is leveraged. The adversarial projector captures reverse mapping probabilities through considerable step sizes to speed up the process and for effectiveness. To allow unsupervised learning for unpaired datasets, a cycle-consistent architecture is designed. This architecture is built with diffusive and non-diffusive processes which bilaterally map between two domains. **AdaD-iff** (Alper Gu¨ng¨or, Salman U. H. Dar, S¸aban O¨ ztu¨rk, Yilmaz Korkmaz, Gok- berk Elmas, Muzaffer O¨ zbey, and Tolga C¸ ukur. 2022) adopts an adaptive diffusion prior to reconstruct MRI images. During inference, it uses an effective diffusion prior that is trained and learned through a rapid diffusion model based upon an adversarial translation over considerable reverse diffusion steps for effective sampling. Given a trained diffusion prior, during inference, two stages are employed for MRI reconstruction. The first stage is a rapid-diffusion stage that initially reconstruct an image based on the trained prior. The second stage is an adaptation stage, which refines the output generated by the first stage through updating the prior. This update enhances and minimizes the reconstruction loss.

Recently, many studies have been conducted in the image synthesis field about backbone architectures. There has been an ongoing battle between CNNs and transformers again as a backbone architecture of the image synthesis method. In computer vision, attention is incorporated with CNN or utilized to replace specific aspects of CNN while maintaining the whole composition intact. However, a standard transformer can be applied to sequences of image patches to substitute that CNN backbone architecture. Vision transformer (ViT) (Dosovitskiy et al. 2020) works by splitting an image into fixed-size patches. Then, each patch is linearly embedded while adding position embedding to each of them. Following this, the resultant sequence of vectors is fed into a standard transformer encoder (Vaswani et al. 2017). On one hand, CNNs work on feature maps at the high spatial resolution to

enhance sensitivity for local features (He et al. 2015). On the other hand, vision transformers work on feature maps at the lower spatial resolution to improve sensitivity for global contextual features (Dosovitskiy et al. 2020). In recent years, vision transformer (Dosovitskiy et al. 2020; Dalmaz et al. 2022) is leveraged in the image synthesis domain. **Kamran et al.** (Kamran et al. 2021) propose a GAN model that is trained in a semi-supervised manner with a vision transformer architecture incorporating several weighted losses. VTGAN is used for retinal image synthesis and disease prediction. Specifically, it generates the retinal vascular structure, in particular, Fluorescein Angiography images from fundus photographs. Additionally, it can distinguish between normal and abnormal retinas. This model consists of multi-scale generators to capture coarse and fine details features to generate realistic and reasonable vascular images. The discriminator is a vision transformer. ViT (Dosovitskiy et al. 2020) architecture is used to preserve the cohesiveness of fine and coarse features by using additional information, particularly, the position embedding of each patch. CNN can be used for obtaining multi-scale features when a large receptive field is incorporated. However, an overfitting problem might emerge during the training process. Consequently, to address this issue, ViT (Dosovitskiy et al. 2020) is utilized as a discriminator architecture. In another study, **Dalmaz et al.** (Dalmaz et al. 2022) propose ResViT that uses a hybrid architecture of CNN and a vision transformer to learn both local structural and global contextual representations, respectively. ResViT is an adversarial model with a transformer-based generator to map between multi-modal imaging data. The generator is composed of an encoder-decoder along with a central information bottleneck. The information bottleneck consists of aggregated residual trans- former (ART) blocks that combine residual convolutional and vision transformer models. While the vision transformer is sensitive to the global context, the residual convolutional is leveraged to use local precision of convolution operators (He et al. 2015). Thus, ART blocks maintain local and global contexts for medical image synthesis. The discriminator consists of convolutional operators.

The future research directions of image synthesis look promising. Currently the input is from text, sketch, speech or image. In the future, we believe that the input may come from other sources, such as brain signals to support people with special. Furthermore, we expect that the synthesized image may be generated from multiple sources. For instance, the input is a combination of text and speech. In addition, there should be studies in domain adaptation and transfer learning for image synthesis. There is also a need to compare different models in terms of accuracy, the training and testing time. Since the generated results are getting better and better, there should be research on accurately identify the "fake but look real" results for the security purposes. Last but not least, the advancement of image synthesis definitely paves way to video synthesis in the near future.

6 Conclusion

This survey paper provides a comprehensive review of many image synthesis models based on supervised and unsupervised learning, where different forms of inputs are considered to generate an image, such as text, sketch, another image, or speech. Following this, a brief details of benchmarked image generation datasets is discussed since dataset plays a significant role in the success of image synthesis models. Quality, diversity, and the size of the dataset are important factors to consider when training image generation models. Moreover, concise details about several evaluation measures to assess the success of the image

generation models is introduced. Evaluation measures are divided into two main categories: qualitative and quantitative. While qualitative measures are subjective, quantitative measures are objective based on metrics and statistics. Both evaluation types are important to determine the performance of image synthesis models. Finally, discussion about image synthesis challenges and future outlook are introduced.

Acknowledgements The first author would like to thank Umm Al-Qura University, in Saudi Arabia, for the continuous support. This work has been supported in part by the University of Dayton Office for Graduate Academic Affairs through the Graduate Student Summer Fellowship Program. This work was also supported in part by the National Science Foundation under Grant NSF 2025234.

Data availability The datasets generated during and/or analyzed during the current study are available from the corresponding author on reasonable request.

Declarations

Conflicts of interest The authors have no conflicts of interest to declare that are relevant to the content of this article.

References

Adiban M, Safari A, Salvi G (2020) Step-gan: A step-by-step training for multi generator gans with application to cyber security in power systems. arXiv [eess.SP].

. Arjovsky M, Chintala S, Bottou L (2017) Wasserstein gan. arXiv [stat.ML]

Baraheem SS, Nguyen TV (2020b) Aesthetic-aware text to image synthesis. In 2020b 54th Annual Conference on Information Sciences and Systems (CISS), p 1–6

Baraheem SS, Nguyen TV (2020) Text-to-image via mask anchor points. Pattern Recognition Lett 133:25–32

Caesar H, Uijlings J, Ferrari V (2018) Coco-stuff: Thing and stuff classes in context. In 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, p 1209–1218

Cai L, Gao H, Ji S (2017) Multi-stage variational auto-encoders for coarse- to-fine image generation. arXiv [cs.CV]

Chalechale A, Mertins A, Naghdy G (2004) Edge image description us- ing angular radial partitioning. IEE Proc - Vis. Image Signal Process 151(2):93

Chen T, Cheng M-M, Tan P, Shamir A, Hu S-M (2009) Sketch2photo: internet image montage. ACM Trans Graph 28(5):1–10

Chen W, Hays J (2018) Sketchygan: towards diverse and realistic sketch to image synthesis. arXiv [cs.CV]

Chen H, Jiang L (2019) Efficient gan-based method for cyber-intrusion detection. arXiv [cs.LG

Chen X, Kingma DP, Salimans T, Duan Y, Dhariwal P, Schulman J, Sutskever I, Abbeel P (2016) Abbeel. Variational lossy autoencoder. arXiv [cs.LG]

Chen J, Shen Y, Gao J, Liu J, Liu X (2017a) Language-based image editing with recurrent attentive models. arXiv [cs.CV]

Chen L, Srivastava S, Duan Z, Xu C (2017b) Deep cross-modal audio- visual generation arXiv [cs.CV].

Chicco D (2021) Siamese neural networks: An overview. Methods Mol Biol 2190:73–94

Cho K, Van Merriënboer B, Gulcehre C, Bahdanau D, Bougares F, Schwenk H, Bengio Y (2014) Learning phrase representations using rnn encoder–decoder for statistical machine translation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP).

Comaniciu D, Meer P (2002) obust analysis of feature spaces: color image segmentation. In Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition.

"Common problems," Google Developers. https://developers.google.com/machine-learning/gan/problems (accessed Jan. 10, 2023)

Cordts M, Omran M, Ramos S, Rehfeld T, Enzweiler M, Benenson R, Franke U, Roth S, Schiele B (2016) The cityscapes dataset for semantic ur- ban scene understanding. In 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)

Dalal N, Triggs B (2005) Histograms of oriented gradients for human de- tection. In 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), vol 1, p 886–893
Dalmaz O, Yurt M, Cukur T (2022) Resvit: Residual vision transformers for multimodal medical image synthesis. IEEE Trans Med Imaging 41(10):2598–2614
Das A, Kottur S, Moura JM, Lee S, Batra D (2017) Visual dialog. In 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
Davenport RK, Rogers CM, Russell IS (1973) Cross modal perception in apes. Neuropsychologia 11(1):21–28
Deng L (2012) The mnist database of handwritten digit images for ma- chine learning research [best of the web. IEEE Signal Process Mag 29(6):141–142
Deng J, Dong W, Socher R, Li LJ, Li K, Fei-Fei L (2009) Imagenet: A large-scale hierarchical image database. In 2009 IEEE Conference on Computer Vision and Pattern Recognition, p 248–255
Denton EL, Chintala S, Fergus R 2015 Deep generative image models using a laplacian pyramid of adversarial networks. arXiv [cs.CV]
Devlin J, Chang MW, Lee K, Toutanova K (2018) Bert: Pre-training of deep bidirectional transformers for language under- standing. arXiv [cs.CL]
Dokmanic I, Parhizkar R, Ranieri J, Vetterli M (2015) Euclidean distance matrices: essential theory, algorithms and applications. IEEE Signal Process Mag 32(6):12–30
Dosovitskiy A, Springenberg JT, Brox T (2015) Learning to generate chairs with convolutional neural networks. In 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), p 1538–1546
Dosovitskiy A, Beyer L, Kolesnikov A, Weissenborn D, Zhai X, Unterthiner T, Dehghani M, Minderer M, Heigold G, Gelly S, Uszkoreit J (2020) An image is worth 16x16 words: Transformers for image recognition at scale. arXiv [cs.CV]
Dumitrescu B (2017) Gram matrix representation. Signals and Communication Technology. Springer International Publishing, Cham, pp 23–69
Dumoulin V, Shlens J, Kudlur M (2016) A learned representation for artistic style. arXiv [cs.CV]
Eitz M, Richter R, Hildebrand K, Boubekeur T, Alexa M (2011) Photo- sketcher: interactive sketch-based image synthesis. IEEE Comput Graph Appl 31(6):56–66
Eitz M, Hays J, Alexa M (2012) How do humans sketch objects? ACM Trans Graph 31(4):1–10
Eitz M, Hildebrand K, Boubekeur T, Alexa M (2009) A descriptor for large scale image retrieval based on sketched feature lines. In Proceed- ings of the 6th Eurographics Symposium on Sketch-Based Interfaces and Modeling - SBIM
Elgammal A, Liu B, Elhoseiny M, Mazzone M (2017) Can: creative adversarial networks, generating 'art' by learning about styles and deviating from style norms. arXiv [cs.AI]
Mikolov T, Sutskever I, Chen K, Corrado GS, Dean J (2013) Distributed representations of words and phrases and their compo- sitionality. arXiv [cs.CL]
He K, Zhang X, Ren S, Sun J. (2015) Deep residual learning for image recognition. arXiv [cs.CV]
Perarnau G, Weijer J, Raducanu B, A´lvarez JM (2016) Invertible conditional gans for image editing. arXiv [cs.CV]
Liu Y, Qin Z, Luo Z, Wang H (2017) Auto-painter: cartoon image generation from sketch by using conditional generative adversarial networks. arXiv [cs.CV]
Feng F, Li R, Wang X (2014) Cross-modal retrieval with correspondence autoencoder. In Proceedings of the ACM International Conference on Multimedia, vol MM 14
Feng Z, Xu C, Tao D (2019) Self-supervised representation learning by rotation feature decoupling. In 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), p 10364–10374
Finlayson SG, Lee H, Kohane IS, Oakden-Rayner L (2018) Towards generative adversarial networks as a new paradigm for radiology education. arXiv [cs.CV]
Gadde R, Karlapalem K (2011) Aesthetic guideline driven photography by robots. Proceedings of the Twenty-Second international joint conference on Artificial Intelligence, p 2060–2065
Gao L, Chen D, Zhao Z, Shao J, Shen HT (2021) Lightweight dynamic conditional gan with pyramid attention for text-to-image synthesis. Pattern Recogn 110(107384):107384
Gao C, Liu Q, Xu Q, Wang L, Liu J, Zou C (2020) Sketchycoco: Image generation from freehand scene sketches,. arXiv [cs.CV].
Goodfellow IJ, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y (2014) Generative adversarial nets. In Proceedings of the 27th International Conference on Neural Information Processing Systems vol 2, p 2672–2680
Gou Y, Wu Q, Li M, Gong B, Han M (2020) Segattngan: Text to image generation with segmentation attention. arXiv [cs.CV]
. Gregor K, Danihelka I, Graves A, Rezende DJ, Wierstra D (2015) Draw: a recurrent neural network for image generation. arXiv [cs.CV]

Grother P (1995) Nist special database 19 handprinted forms and characters database.
Gulrajani I, Kumar K, Ahmed F, Taiga AA, Visin F, Vazquez D, Courville A (2016) Pixelvae: A latent variable model for natural images. arXiv [cs.LG]
Gulrajani I, Ahmed F, Arjovsky M, Dumoulin V, Courville AC (2017) Improved training of wasserstein gans. arXiv [cs.LG]
Güngör A, Dar SU, Öztürk Ş, Korkmaz Y, Elmas G, Özbey M, Güngör A, Çukur T (2022) Adaptive diffusion priors for accelerated mri reconstruction. arXiv [eess.IV]
Hao W, Zhang Z, Guan H (2018) Cmcgan: a uniform framework for cross-modal visual-audio mutual generation. Proc. Conf. AAAI Artif. Intell, 32(1)
Harris Zellig S (1981) Distributional Structure. Springer Netherlands, Dordrecht, pp 3–22
He K, Zhang X, Ren S, Sun J (2015) Deep residual learning for image recognition. arXiv [cs.CV]
Heusel M, Ramsauer H, Unterthiner T, Nessler B, Hochreiter S (2017) Gans trained by a two time-scale update rule converge to a local nash equilibrium. arXiv [cs.LG].
Ho J, Jain A, Abbeel P (2020) Denoising diffusion probabilistic models. arXiv [cs.LG]
Huang X, Belongie S (2017) Arbitrary style transfer in real-time with adaptive instance normalization. arXiv [cs.CV]
Huang GB, Mattar M, Berg T, Learned-Miller E (2008) Labeled faces in the wild: A database for studying face recognition in un- constrained environments. In Workshop on Faces in "Real-Life" Images: Detection, Alignment, and Recognition
Huang X, Liu M-Y, Belongie S, Kautz J (2018a) Multimodal unsupervised image-to-image translation, arXiv [cs.CV]
Huang H, Yu PS, Wang C (2018b) An introduction to image synthesis with generative adversarial nets, 2018b. arXiv [cs.CV]
Huiskes MJ, Lew MS (2008) Lew. The mir flickr retrieval evaluation. In Proceed- ing of the 1st ACM international conference on Multimedia information retrieval - MIR
Ioffe S, Szegedy C (2015) Batch normalization: Accelerating deep network training by reducing internal covariate shift. arXiv [cs.LG]
Isola P, Zhu J-Y, Zhou T, Efros AA (2016) Image-to-image translation with conditional adversarial networks. arXiv [cs.CV]
Jia Y, Shelhamer E, Donahue J, Karayev S, Long J, Girshick R, Guadarrama S, Darrell T (2014) Caffe: Convolutional architecture for fast feature embedding. arXiv [cs.CV]
Jinzhen Mu, Chen C, Zhu W, Li S, Zhou Y (2022) Taming mode collapse in generative adversarial networks using cooperative realness discriminators. IET Image Proc 16(8):2240–2262
Johnson J, Alahi A, Li Fei-Fei (2016) Perceptual Losses for Real-time Style Transfer and Super-resolution. Springer International Publishing, Cham
Jolicoeur-Martineau A, Piché-Taillefer R, Combes RT, Mitliagkas I (2020) Adversarial score matching and im- proved sampling for image generation. arXiv [cs.LG]
Amit Kamran S, Fariha Hossain K, Tavakkoli A, Zuckerbrod SL, Baker SA (2021) Vtgan: semi-supervised retinal image synthesis and disease prediction using vision transformers. arXiv [eess.IV]
Karras T, Aila T, Laine S, Lehtinen J (2017) Progressive growing of gans for improved quality, stability, and variation. arXiv [cs.NE]
Kingma DP, Welling M (2013) Auto-encoding variational bayes. arXiv [stat.ML]
Kingma DP, Salimans T, Jozefowicz R, Chen X, Sutskever I, Welling M (2016) Welling. Improving variational inference with inverse autoregressive flow. arXiv [cs.LG].
Kiros R, Zhu Y, Salakhutdinov RR, Zemel R, Urtasun R, Torralba A, Fidler S (2015) Skip-thought vectors. arXiv [cs.CL]
Kolesnikov A, Zhai X, Beyer L (2019) Beyer. Revisiting self-supervised visual representation learning. arXiv [cs.CV]
Kong Z, Ping W, Huang J, Zhao K, Catanzaro B (2020) Diffwave: a versatile diffusion model for audio synthesis. arXiv [eess.AS]
Kramer MA (1991) Nonlinear principal component analysis using autoassocia- tive neural networks. AIChE J 37(2):233–243
Krizhevsky A (2009) Learning multiple layers of features from tiny images. Technical report, Journal of Software Engineering and Applications.
Krizhevsky A, Sutskever I, Hinton GE (2017) Imagenet classification with deep convolutional neural networks. Commun ACM 60(6):84–90
Kruskal JB (1964) Nonmetric multidimensional scaling: A numerical method. Psychometrika 29(2):115–129
Kumar N, Berg AC, Belhumeur PN, Nayar SK (2009) Attribute and simile classifiers for face verification. In 2009 IEEE 12th International Conference on Computer Vision. IEEE p 365–372

Li C, Wand M (2016) Precomputed Real-time Texture Synthesis with Markovian Generative Adversarial Networks. Springer International Publishing, Cham
Li B, Liu X, Dinesh K, Duan Z, Sharma G (2016) Creating a multi- track classical musical performance dataset for multimodal music analysis: challenges, insights, and applications. EEE Trans Multimedia 21(2):522–535
Li L, Sun Y, Hu F, Zhou T, Xi X, Ren J (2020) Text to realistic image generation with attentional concatenation generative adversarial networks. Discrete Dyn Nat Soc 2020(1):10
Li JG, Zhang XF, Jia CM, Xu JZ, Zhang L, Wang Y, Ma SW, Gao W (2020) Direct speech-to-image translation. IEEE Journal of Selected Topics in Signal Processing 14(3):517–529
Li Z, Deng C, Yang E, Tao D (2021) Staged sketch-to-image synthesis via semi-supervised generative adversarial networks. IEEE Trans Multi- Media 23:2694–2705
Lin TY, Maire M, Belongie S, Hays J, Perona P, Ramanan D, Dollár P, Zitnick CL (2014) Microsoft coco: Common objects in context. arXiv [cs.CV]
Lin TY, Dollár P, Girshick R, He K, Hariharan B, Belongie S (2016) Feature pyramid networks for object detection. arXiv [cs.CV]
Lin YJ, Wu PW, Chang CH, Chang EY, Liao SW (2019) Liao. Relgan: Multi-domain image-to-image translation via relative attributes. In 2019 IEEE/CVF International Conference on Computer Vision (ICCV)
Liu L, Chen R, Wolf L, Cohen-Or D (2010) Optimizing photo composition. Comput. Graph. Forum 29(2):469–478
Liu Y, Dellaert F (2002) A classification based similarity metric for 3D image retrieval, in Proceedings. 1998 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No.98CB36231), p. 800–805
Liu R, Yu Q, Yu S (2019) Unsupervised sketch-to-photo synthesis. arXiv [cs.CV]
Liu B, Zhu Y, Song K, Elgammal A (2020) Self-supervised sketch-to- image synthesis. arXiv [cs.CV]
Lowe DG (2004) Distinctive image features from scale-invariant keypoints. Int J Comput vis 60(2):91–110
Lu Y, Wu S, Tai Y-W, C.-K. (2018) Tang. Image generation from sketch constraint using contextual gan. Computer Vision – ECCV 2018. Springer International Publishing, Cham, pp 213–228
Manjunath BS, Salembier P, Sikora T (2002) Introduction to mpeg-7: Multimedia content description interface. In: Manjunath BS, Salembier P, Sikora T (eds) Introduction to mpeg-7: Multimedia content description interface. John Wiley and Sons, Chichester
Mansimov E, Parisotto E, Ba J.L, Salakhutdinov R (2015) Generating images from captions with attention. arXiv [cs.LG]
Mao X, Li Q, Xie H, Lau RY, Wang Z, Paul Smolley S (2016) Least squares generative adversarial networks, 2016. arXiv [cs.CV]
Miller GA, Beckwith R, Fellbaum C, Gross D, Miller KJ (1990) In- troduction to wordnet: An on-line lexical database. Int j Lexicogr 3(4):235–244
Mirza M, Osindero S (2014) Conditional generative adversarial nets. arXiv [cs.LG].
Özbey M, Dar SU, Bedel HA, Dalmaz O, Özturk Ş, Güngör A, Çukur T (2022) Unsupervised medical image translation with adversarial diffusion models. arXiv [eess.IV]
Chen N, Zhang Y, Zen H, Ron Weiss J, Norouzi M, Chan W (2020) Wavegrad: Estimating gradients for waveform gener- ation. arXiv [eess.AS]
Nazeri K, Ng E, Joseph T, Qureshi FZ, Ebrahimi M (2019) Edgeconnect: Generative image inpainting with adversarial edge learning. arXiv [cs.CV]
Nichol AQ, Dhariwal P (2021) Improved denoising diffusion probabilistic models, p 18–24. arXiv [cs. LG]
Nilsback ME, Zisserman A (2008) Automated flower classification over a large number of classes. In 2008 Sixth Indian Conference on Computer Vision. Graphics and Image Processing.
Odena A, Olah C, Shlens J (2016) Conditional image synthesis with auxiliary classifier gans. arXiv [stat. ML]
Ojala T, Pietikäinen M, Harwood D (1996) A comparative study of tex- ture measures with classification based on featured distributions. Pattern Recognit 29(1):51–59
Osahor U, Kazemi H, Dabouei A, Nasrabadi N (2020) Quality guided sketch-to-photo image synthesis. In 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW).
Otsu N (1979) A threshold selection method from gray-level histograms. IEEE Trans Syst Man Cybern 9(1):62–66
Park H, Yoo Y, Kwak N (2018) Mc-gan: Multi-conditional generative adversarial network for image synthesis. arXiv [cs.CV]

Park T, Liu M-Y, Wang T-C, Zhu J-Y (2019a) Semantic image synthesis with spatially-adaptive normalization. arXiv [cs.CV]
Park T, Liu MY, Wang TC, Zhu JY (2019b) Semantic image synthesis with spatially-adaptive normalization. In 2019b IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), p 2337–2346
Pereira JC, Coviello E, Doyle G, Rasiwasia N, Lanckriet G, Levy R, Vasconcelos N (2014) On the role of correlation and abstraction in cross- modal multimedia retrieval. IEEE Trans Pattern Anal Mach Intell 36(3):521–535
Qiao T, Zhang J, Xu D, Tao D (2019) Mirrorgan: Learning text-to-image generation by redescription. arXiv [cs.CL]
Radford A, Metz L, Chintala S (2015) Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv [cs.LG]
Rajput GG, Prashantha (2019) Sketch based image retrieval using grid approach on large scale database. Procedia Comput. Sci 165:216–223
Rasmussen CE (1999) The infinite gaussian mixture model. In Proceedings of the 12th International Conference on Neural Information Processing Systems, p 554–560
Reed S, Akata Z, Yan X, Logeswaran L, Schiele B, Lee H (2016a) Generative adversarial text to image synthesis. arXiv [cs.NE]
Reed S, Akata Z, Lee H, Schiele B (2016a) Learning deep representations of fine-grained visual descriptions. arXiv [cs.CV]
Ronneberger O, Fischer P, Brox T (2015) U-net: Convolutional networks for biomedical image segmentation. Springer International Publishing, Cham
Rother C, Kolmogorov V, Blake A (2004) Grabcut: Interactive foreground extraction using iterated graph cuts. ACM Trans Graph 23(3):309–314
Salimans T, Goodfellow I, Zaremba W, Cheung V, Radford A, Chen (2016) Improved techniques for training gans. In Proceedings of the 30th International Conference on Neural Information Processing Systems, p 2234–2242.
Sangkloy P, Burnell N, Ham C, Hays J (2016) The sketchy database: learning to retrieve badly drawn bunnies. ACM Trans Graph 35(4):1–12
Sangkloy P, Lu J, Fang C, Yu F, Hays J (2016b) Scribbler: controlling deep image synthesis with sketch and color. arXiv [cs.CV]
Sasaki H, Willcocks CG, Breckon TP (2021) Unit-ddpm: Unpaired image translation with denoising diffusion probabilistic models. arXiv [cs.CV]
Schuster M, Paliwal KK (1997) Bidirectional recurrent neural networks. IEEE Transactions on Signal Processing: a Publication of the IEEE Signal Processing Society 45(11):2673–2681
Sharma S, Suhubdy D, Michalski V, Kahou SE, Bengio Y (2018) Chat- painter: Improving text to image generation using dialogue. arXiv [cs.CV]
Sohl-Dickstein J, Weiss EA, Maheswaranathan N, Ganguli S (2015) Deep unsupervised learning using nonequilibrium thermodynam- ics. arXiv [cs.LG].
Song Y, Ermon S (2020) Improved techniques for training score- based generative models. Adv Neural Inf Process Syst 33(12438):12448
Souza DM, Wehrmann J, Ruiz DD (2020) Efficient neural architecture for text-to-image synthesis. arXiv [cs.LG]
Springenberg JT, Dosovitskiy A, Brox T, Riedmiller M (2014) Striving for simplicity: The all convolutional net. arXiv [cs.LG]
Stein BE, Meredith MA (1993) The merging of the senses. The MIT Press, Cambridge
Sushko V, Schönfeld E, Zhang D, Gall J, Schiele B, Khoreva A (2020) You only need adversarial supervision for semantic image synthesis. arXiv [cs.CV]
Szanto B, Pozsegovics P, Vamossy Z, Sergyan S (2011) Sketch4match — content-based image retrieval system using sketches. In 2011 IEEE 9th In- ternational Symposium on Applied Machine Intelligence and Informatics (SAMI), p 183–188
Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A (2015) Going deeper with convolutions. In 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), p 1–9
Taigman Y, Yang M, Ranzato MA, Wolf L (2014) Deepface: closing the gap to human-level performance in face verification. In 2014 IEEE Conference on Computer Vision and Pattern Recognition, p 1701–1708
Tanveer MI, Liu J, Hoque ME (2015) Unsupervised extraction of human- interpretable nonverbal behavioral cues in a public speaking scenario. In Proceedings of the 23rd ACM international conference on Multimedia - MM 15

Thaung L (2020) Advanced data augmentation: With generative adversarial networks and computer-aided design.
Torralba A, Fergus R, Freeman WT (2008) 80 million tiny images: a large data set for nonparametric object and scene recognition. IEEE Trans Pattern Anal Mach Intell 30(11):1958–1970
Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I (2017) Attention is all you need. arXiv [cs.CL]
Vincent P, Larochelle H, Lajoie I, Bengio Y, Manzagol P-A (2010) Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. J Mach Learn Res 11:3371–3408
Vroomen J, Gelder B (2000) Sound enhances visual perception: cross-modal effects of auditory organization on vision. J Exp Psychol Hum Percept Perform 26(5):1583–1590
Wah C, Branson S, Welinder P, Perona P, Belongie S (2011) The caltech-ucsd birds-200–2011 dataset.
Wah C, Branson S, Welinder P, Perona P, Belongie S (2011) The caltech-ucsd birds200–2011 dataset. Advances in Water Rerces - ADV WATER RESOUR.
Wang Z, Simoncelli EP, Bovik A (2003) Multi-scale structural similarity for image quality assessment. Ieee, New York
Wang Z, Bovik AC, Sheikh HR, Simoncelli EP (2004) Image quality assessment: from error measurement to structural similarity. IEEE Trans Image Processing 13(4):600–612
Wang X, Qiao T, Zhu J, Hanjalic A, Scharenborg O (2021) Generating images from spoken descriptions. IEEE ACM Trans Audio Speech Lang Process 29:850–865
Wang TC, Liu MY, Zhu JY, Tao A, Kautz J, Catanzaro B (2017) Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. ArXiv [Cs.CV]
Wang M,. Lang C,. Liang L,. Lyu G,. Feng S, and. Wang T (2020) Attentive generative adversarial network to bridge multi-domain gap for image syn- thesis. In 2020 IEEE International Conference on Multimedia and Expo (ICME,).
Welinder P, Branson S, Mita T, Wah C, Schroff F, Belongie S, Perona P (2010a) Caltech-ucsd birds 200". Technical report cns-tr-2010a-001, California Institute of Technology.
Welinder P, Branson S, Perona P (2010b) The multidimensional wisdom of crowds. NIPS.
Wu W, Cao K, Li C, Qian C, Loy CC (2019) Transgaga: Geometry- aware unsupervised image-to-image translation. In 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), p 8012–8021
Xian W, Sangkloy P, Agrawal V, Raj A, Lu J, Fang C, Yu F, Hays J (2018) Texturegan: Controlling deep image synthesis with texture patches. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), p 8456–8465
Xiao J, Hays J, Ehinger KA, Oliva A, Torralba A (2010) Sun database: Large-scale scene recognition from abbey to zoo. In 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, p 3485–3492
Xie S, Tu Z (2015) Holistically-nested edge detection. In 2015 IEEE Inter- national Conference on Computer Vision (ICCV), p 1395–1403
Xu T, Zhang P, Huang Q, Zhang H, Gan Z, Huang X, He X (2017) At- tngan: Fine-grained text to image generation with attentional generative adversarial networks. arXiv [cs.CV]
Yan Z, Zhang H, Wang B, Paris S, Yu Y (2014) Automatic photo adjustment using deep neural networks. arXiv [cs.CV]
Yan X, Yang J, Sohn K, Lee H (2015) Attribute2image: conditional image generation from visual attributes. arXiv [cs.LG]
Yu Q, Yang Y, Song YZ, Xiang T (2015) Hospedales. Sketch-a-net that beats humans. arXiv [cs.CV]
Yu Q, Liu F, Song Y-Z, Xiang T, Hospedales TM, Loy CC (2016) Sketch me that shoe. In 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), p 799–807
Yu J, Lin Z, Yang J, Shen X, Lu X, Huang TS (2018) Generative image inpainting with contextual attention. arXiv [cs.CV]
Zhang Z, Luo P, Loy CC, Tang X (2014) Deep learning face attributes in the wild. arXiv [cs.CV]
Zhang H, Xu T, Li H, Zhang S, Wang X, Huang X, Metaxas D (2016) Stackgan: text to photo-realistic image synthesis with stacked generative adversarial networks. arXiv [cs.CV]
Zhang H, Xu T, Li H,. Zhang S, Wang X, Huang X, Metaxas D (2017) Stackgan++: Realistic image synthesis with stacked generative adversar- ial networks,. arXiv [cs.CV]
Zhang R, Isola P, Efros AA, Shechtman E, Wang O (2018) The unreasonable effectiveness of deep features as a perceptual metric. arXiv [cs.CV]
Zhang P, Zhang B, Chen D, Yuan L, Wen F (2020) Cross-domain cor- respondence learning for exemplar-based image translation. arXiv [cs.CV]

Zhang H, Koh JY, Baldridge J, Lee H, Yang Y (2021a) Cross-modal contrastive learning for text-to-image generation. arXiv [cs.CV]
Zhang J, Li K, Lai YK, Yang J (2021b) Pise: Person image synthesis and editing with decoupled gan. In 2021b IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) p 7978–7986
Zhao T, Chen C, Liu Y, Zhu X (2021) Guigan: Learning to generate gui designs using generative adversarial networks. arXiv [cs.HC]
Zhou B, Zhao H, Puig X, Fidler S, Barriuso A, Torralba A (2016) Se- mantic understanding of scenes through the ade20k dataset. Int J Comput Vision 127:302–321
Zhou B, Lapedriza A, Khosla A, Oliva A, Torralba A (2018) Places: A 10 million image database for scene recognition. IEEE Trans Pattern Anal Mach Intell 40(6):1452–1464
Zhu X, Goldberg AB, Eldawy M, Dyer CR, Strock B (2007) A text-to- picture synthesis system for augmenting communication. In Proceedings of the 22nd national conference on Artificial intelligence, vol 2, p 1590–1595
Zhu J-Y, Park T, Isola P, Efros AA (2017) Unpaired image-to- image translation using cycle-consistent adversarial networks. arXiv [cs.CV]
Zhu P, Abdal R, Qin Y, Wonka (2020) Wonka. Sean: Image synthesis with semantic region-adaptive normalization. In 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)
Zou C, Mo H, Gao C, Du R, Fu H (2019) Language-based colorization of scene sketches. ACM Trans Graph 38(6):1–16