Computational Visual Media
https://doi.org/10.1007/s41095-021-0234-8 Vol. 8, No. 1, March 2022, 3–31
Review Article

Deep image synthesis from intuitive user input: A review and perspectives

Yuan Xue[1], Yuan-Chen Guo[2], Han Zhang[3], Tao Xu[4], Song-Hai Zhang[2], and Xiaolei Huang[1] (✉)

Abstract In many applications of computer graphics, art, and design, it is desirable for a user to provide intuitive non-image input, such as text, sketch, stroke, graph, or layout, and have a computer system automatically generate photo-realistic images according to that input. While classically, works that allow such automatic image content generation have followed a framework of image retrieval and composition, recent advances in deep generative models such as generative adversarial networks (GANs), variational autoencoders (VAEs), and flow-based methods have enabled more powerful and versatile image generation approaches. This paper reviews recent works for image synthesis given intuitive user input, covering advances in input versatility, image generation methodology, benchmark datasets, and evaluation metrics. This motivates new perspectives on input representation and interactivity, cross fertilization between major image generation paradigms, and evaluation and comparison of generation methods.

Keywords image synthesis; intuitive user input; deep generative models; synthesized image quality evaluation

1 College of Information Sciences and Technology, the Pennsylvania State University, University Park, PA, USA. E-mail: Y. Xue, yuanxue@psu.edu; X. Huang, sharon.x.huang@psu.edu (✉).

2 Department of Computer Science and Technology, Tsinghua University, Beijing, China, and Beijing National Research Center for Information Science and Technology (BNRist), Tsinghua University, Beijing, China. E-mail: Y.-C. Guo, guoyc19@mails.tsinghua.edu.cn; S.-H. Zhang, shz@tsinghua.edu.cn.

3 Google Brain, Mountain View, CA, USA. E-mail: zhanghan@google.com.

4 Facebook, Menlo Park, CA, USA. E-mail: xutao@fb.com.

Manuscript received: 2021-01-25; accepted: 2021-03-27

1 Introduction

Machine learning and artificial intelligence have given computers the abilities to mimic or even defeat humans in tasks like playing games of chess and go, recognizing objects in images, and translating from one language to another. An interesting next pursuit would be to see if computers can mimic creative processes such as those used by painters in making pictures, or assisting artists or architects in making artistic or architectural designs. In fact, in the past decade, we have witnessed advances in systems that synthesize an image from a text description [1–4] or from a learned style of content [5], paint a picture given a sketch [6–9], render a photorealistic scene from a wireframe [10, 11], and create virtual reality content from images and videos [12], among others. A comprehensive review of such systems can explain the current state-of-the-art in such pursuits, reveal open challenges, and illuminate future directions. In this paper, we make an attempt at a comprehensive review of image synthesis and rendering techniques given simple, intuitive user input such as text, sketches or strokes, semantic label maps, poses, visual attributes, graphs, and layouts. We first present ideas on what makes a good paradigm for image synthesis from intuitive user input and review popular metrics for evaluating the quality of generated images. We then introduce several mainstream methodologies for image synthesis given user input, and review algorithms developed for application scenarios specific to different formats of user input. We also summarize major benchmark datasets used by current methods, and advances and trends in image synthesis methodology. Finally, we provide our perspective on future directions for developing image synthesis models capable of

generating complex images that are closely aligned with user input, have high visual realism, and adhere to constraints of the physical world.

2 What makes a good paradigm for image synthesis from intuitive user input?

2.1 What types of user input do we need?

For an image synthesis model to be user-friendly and useful in real-world applications, the user input should be intuitive, easily interactively edited, and commonly used in design and creative processes. We define an input modality to be intuitive if it has the following characteristics:

- *Accessibility.* The input should be easy to provide, especially for non-professionals. Taking sketching as an example, even people without any trained skills in drawing can express rough ideas through sketching.
- *Expressiveness.* The input should be expressive enough to allow someone to convey not only simple concepts but also complex ideas.
- *Interactivity.* The input should be interactive to some extent, so that users can interactively modify its content, to fine tune the synthesized output in an iterative fashion.

Taking painting as an example, a sketch is an intuitive input because it is what humans use to design the composition of a painting. On the other hand, being intuitive often means that the information provided by the input is limited, which makes the generation task more challenging. Moreover, for different types of applications, suitable forms of user input can be quite different.

For image synthesis with intuitive user input, the most relevant and well-investigated method is to use conditional image generation models. In other words, user inputs are treated as conditional input to the synthesis model to guide generation by conditional generative models. In this review, we mainly discuss mainstream conditional image generation applications including those using text descriptions, sketches or strokes, semantic maps, poses, visual attributes, or graphs as intuitive input. The processing and representation of the user input are usually application- and modality-dependent. When given text descriptions as input, pretrained text embeddings are often used to convert text into a vector-representation of the input words. Image-like inputs, such as sketches, semantic maps, and poses are often represented as images and processed accordingly. In particular, one-hot encoding can be used in semantic maps to represent different categories, and keypoint maps can be used to encode poses where each channel represents the position of a body keypoint; both result in multi-channel image-like tensors as input. Using visual attributes as input is most similar to general conditional generation tasks, where attributes can be provided in the form of class vectors. For graph-like user inputs, additional processing steps are required to extract relationship information represented in the graphs. For instance, graph convolutional networks (GCNs) [13] can be applied to extract node features from input graphs. More details of the processing and representation methods of various input types will be reviewed and discussed in Section 4.

2.2 How do we evaluate the output synthesized images?

The quality of an image synthesis method depends on how well its output adheres to user input, whether the output is photorealistic or structurally coherent, and whether it can generate a diverse pool of images that satisfy requirements. General metrics have been designed for evaluating the quality and sometimes diversity of synthesized images. Widely adopted metrics use different methods to extract features from images and then calculate different scores or distances. Such metrics include peak signal-to-noise ratio, Inception score, Fréchet Inception distance, structural similarity index measure, and learned perceptual image patch similarity.

Peak signal-to-noise ratio (PSNR) measures the physical quality of a signal by the ratio between the maximum possible power of the signal and the power of the noise affecting it. For images, PSNR can be represented as

$$\mathrm{PSNR} = \frac{1}{3}\sum_{k} 10\log_{10}\frac{\max \mathrm{DR}^2}{\frac{1}{m}\sum_{i,j}\left(t_{i,j,k}-y_{i,j,k}\right)^2} \quad (1)$$

where k is the number of channels, DR is the dynamic range of the image (255 for 8-bit images), m is the number of pixels, i, j are indices iterating over every pixel, and t and y are the reference image and synthesized image, respectively.

The Inception score (IS) [14] uses a pre-trained

Inception [15] network to compute the KL-divergence between the conditional class distribution and the marginal class distribution. The Inception score is defined as

$$\mathrm{IS} = \exp(\mathbb{E}_x \mathrm{KL}(P(y|x)||P(y))) \tag{2}$$

where x is an input image and y is the label predicted by an Inception model. A high Inception score indicates that the generated images are diverse and semantically meaningful.

Fréchet Inception distance (FID) [16] is a popular evaluation metric for image synthesis tasks, especially for generative adversarial network (GAN) based models. It computes the divergence between the synthetic data distribution and the real data distribution:

$$\mathrm{FID} = ||\hat{m} - m||_2^2 + \mathrm{tr}(\hat{C} + C - 2(C\hat{C})^{1/2}) \tag{3}$$

where m, C and $\hat{m}$, $\hat{C}$ represent the mean and covariance of the feature embeddings of the real and the synthetic distributions, respectively. The feature embedding is extracted from a pre-trained Inception-v3 [15] model.

Structural similarity index measure (SSIM) [17] or multi-scale structural similarity (MS-SSIM) metric [18] gives a score for relative similarity between an image and a reference image, unlike absolute measures such as PSNR. The SSIM is defined as

$$\mathrm{SSIM}(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{\left(\mu_x^2 + \mu_y^2 + c_1\right)\left(\sigma_x^2 + \sigma_y^2 + c_2\right)} \tag{4}$$

where μ and σ indicate the average and variance of two windows x and y respectively, and c_1 and c_2 are two variables to stabilize division by vanishing denominators. The SSIM measures perceived image quality considering structural information. It tests pair-wise similarity between generated images, where a lower score indicates higher diversity of generated images (i.e., fewer mode collapses).

Another metric based on features extracted from pre-trained CNN networks is the learned perceptual image patch similarity (LPIPS) score [19]. The distance is calculated as

$$d(x, x_0) = \sum_l \frac{1}{H_l W_l} \sum_{h,w} \left\| w_l \odot \left(\hat{y}_{hw}^l - \hat{y}_{0hw}^l\right)\right\|_2^2 \tag{5}$$

where $\hat{y}^l, \hat{y}_0^l \in \mathbb{R}^{H_l \times W_l \times C_l}$ are a unit-normalized feature stack from the l-th layer in a pre-trained CNN and w_l indicates channel-wise weights. LPIPS evaluates perceptual similarity between image patches using the learned deep features from trained neural networks.

For flow based models [20, 21] and autoregressive models [22–24], the average negative log-likelihood (i.e., bits per dimension) [22] is often used to evaluate the quality of generated images. It is interpretable as the number of bits that a compression scheme based on this model would need to compress every RGB color value [22].

Apart from metrics designed for general purposes, specific evaluation metrics have been proposed for different applications with various input types. For instance, using text descriptions as input, R-precision [25] evaluates whether a generated image is well conditioned on the given text description. It is measured by retrieving relevant text given an image query. For sketch-based image synthesis, classification accuracy is used to measure the realism of the synthesized objects [7, 8] and how well the identities of synthesized results match those of real images [26]. Also, similarity between input sketches and edges of synthesized images can be measured to evaluate the correspondence between the input and output [8]. In the scenario of pose-guided person image synthesis, "masked" versions of IS and SSIM, Mask-IS and Mask-SSIM are often used to ignore the effects of the background [27–31], since we want to focus on the synthesized human body. As in sketch-based synthesis, the detection score (DS) is used to evaluate how well the synthesized person can be detected [29, 31] and keypoint accuracy can be used to measure the level of correspondence between keypoints [31]. For semantic maps, a commonly used metric tries to restore the semantic-map input from generated images using a pre-trained segmentation network and then compares the restored semantic map with the original input using intersection over union (IoU) score or other segmentation accuracy measures. Similarly, using visual attributes as input, a pre-trained attribute classifier or regressor can be used to assess the attribute correctness of generated images.

3 Overview of mainstream conditional image synthesis paradigms

3.1 Fundamentals

Image synthesis models with intuitive user inputs often involve different types of generative models, particularly conditional generative models that treat

user input as the observed conditioning variable. Two major goals of the synthesis process are high realism of the synthesized images, and correct correspondences between input conditions and output images. Existing methods vary from more traditional retrieval and composition based methods to more recent deep learning based algorithms. In this section, we give an overview of the architectures and main components of different conditional image synthesis models.

3.2 Retrieval and composition

Traditional image synthesis techniques are mainly based on a retrieval and composition paradigm. In the retrieval stage, candidate images or image fragments are fetched from a large image collection, under some user-provided constraints, like text, sketches, or semantic label maps. Methods like edge extraction, saliency detection, object detection, and semantic segmentation are used to pre-process images in the collection according to different input modalities and generation purposes, after which retrieval can be performed using shallow image features like HoG and shape context [32]. The user may interact with the system to improve the quality of the retrieved candidates. In the composition stage, the selected images or fragments are combined by Poisson blending, alpha blending, or a hybrid of both [33], resulting in the final output image.

The biggest advantages of synthesizing images through retrieval and composition are controllability and interpretability. The user can simply intervene in the generation process at any stage, and easily find out whether the output image looks like it should. But it can not generate instances that do not appear in the collection, which restricts the range and diversity of the output.

3.3 Conditional generative adversarial networks (cGANs)

Generative adversarial networks (GANs) [34] have achieved tremendous success in various image generation tasks. A GAN model typically consists of two networks: a generator network that learns to generate realistic synthetic images and a discriminator network that learns to differentiate between real images and synthetic images generated by the generator. The two networks are optimized alternatively through adversarial training. Plain GAN models are designed for unconditional image generation, and implicitly model the distribution of images. To gain more control over the generation process, conditional GANs (cGANs) [35] synthesize images based on both a random noise vector and a condition vector provided by the user. The objective of training a cGAN as a minimax game is

$$\begin{aligned}\min_{\theta_G}\max_{\theta_D}\mathcal{L}_{\text{cGAN}} = \ & \mathbb{E}_{(x,y)\sim p_{\text{data}}(x,y)}[\log D(x,y)] \\ & + \mathbb{E}_{z\sim p(z),y\sim p_{\text{data}}(y)}[\log(1-D(G(z,y),y)]\end{aligned} \tag{6}$$

where x is the real image, y is the user input, and z is the random noise vector. There are different ways of incorporating user input in the discriminator, such as inserting it at the beginning [35], the middle [36], or the end of the discriminator [37].

3.4 Variational auto-encoders (VAEs)

Variational auto-encoders (VAEs) [38] extend the idea of an auto-encoder and introduce variational inference to approximate the latent representation z encoded from the input data x. The encoder converts x into z in a latent space where the decoder tries to reconstruct x from z. Like GANs which typically assume the input noise vector follows a Gaussian distribution, VAEs use variational inference to approximate the posterior $p(z|x)$ given that $p(z)$ follows a Gaussian distribution. After training the VAE, the decoder is used as a generator, like the generator in a GAN; it can draw samples from the latent space and generate new synthetic data. Based on a simple VAE, Sohn et al. proposed a conditional VAE (cVAE) [39–41] which is a conditional directed graphical model whose input observations modulate the latent variables that generate the outputs. Like cGANs, cVAEs allow user input to provide guidance to the image synthesis process. The training objective for a cVAE is

$$\begin{aligned}\max_{\theta,\phi}\mathcal{L}_{\text{cVAE}} = & \ \mathbb{E}_{z\sim Q_\phi}[\log P_\theta(x \mid z,y)] \\ & - D_{\text{KL}}[Q_\phi(z \mid x,y)\|p(z \mid y)]\end{aligned} \tag{7}$$

where x is the real image, y is the user input, z is the latent variable, and $p(z \mid x)$ is the prior distribution of the latent vectors, such as the Gaussian distribution. ϕ and θ are parameters of the encoder Q and decoder P networks, respectively. A cGAN and a cVAE are illustrated in Fig. 1.

3.5 Other learning-based methods

Other learning-based conditional image synthesis models include hybrid methods such as a combination

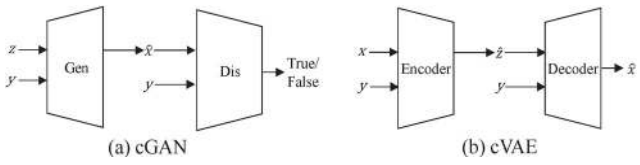

Fig. 1 Use of cGAN and cVAE for image synthesis from intuitive user input. During inferencing, the generator in the cGAN and the decoder in the cVAE generate new images $\hat{x}$ guided by user input y and noise vector or latent variable z.

of VAE and GAN models [42, 43], autoregressive models, and normalizing flow-based models. Among these methods, autoregressive models such as PixelRNN [22], PixelCNN [23], and PixelCNN++ [24] provide tractable likelihood over priors such as class conditions. The generation process is similar to an autoregression model: While classic autoregression models predict future information based on past observations, image autoregressive models synthesize subsequent image pixels based on previously generated or existing nearby pixels.

Flow-based models [20], or normalizing flow based methods, consist of a sequence of invertible transformations which can convert a simple distribution (e.g., a Gaussian) into a more complex one with the same dimension. While flow-based methods have not been widely applied to image synthesis with intuitive user inputs, a few works [21] show that they have great potential in visual attribute guided synthesis and may be applicable in broader scenarios.

Among the aforementioned mainstream paradigms, traditional retrieval and composition methods have the advantage of better controllability and interpretability, although the diversity of synthesized images and flexibility of the models are limited. In comparison, deep learning based methods generally have stronger feature representation capacity, with GANs having the potential to generate images of the highest quality. While having been successfully applied to various image synthesis tasks due to their flexibility, GAN models lack tractable and explicit likelihood estimation. On the contrary, autoregressive models admit tractable likelihood estimation, and can assign a probability to a single sample. VAEs with latent representation learning provide better feature representation power and can be more interpretable. Compared to VAEs and autoregressive models, normalizing flow methods provide both feature representation power and tractable likelihood estimation.

4 Methods specific to applications with various input types

4.1 Background

In this section, we review work that targets application scenarios with specific input types. We review methods for image synthesis from text descriptions, sketches and strokes, semantic label maps, poses, and other input modalities including visual attributes, graphs, and layouts. Among the different input types, text descriptions are flexible, expressive, and user-friendly, yet the comprehension of input content and responding to interactive editing can be challenging for generative models; example applications of text-to-image systems are computer generated art, image editing, computer-aided design, interactive story telling, and visual chat for education and language learning. Image-like inputs such as sketches and semantic maps contain richer information and can better guide the synthesis process, but may require more effort from users to provide adequate input; such inputs can be used in applications such as image and photo editing, computer-assisted painting and rendering. Other inputs such as visual attributes, graphs, and layouts allow appearance, structural, or other constraints to be given as conditional input and can help guide generation of images that preserve visual properties of objects and geometric relations between objects; they can be used in various computer-aided design applications for architecture, manufacturing, publishing, arts, and fashion.

4.2 Text description as input

4.2.1 Background

The task of text-to-image synthesis (Fig. 2) uses descriptive sentences as inputs to guide the generation of corresponding images. The generated image types vary from single-object images [44, 45] to multi-object images with complex backgrounds [46]. Descriptive sentences in a natural language offer a general and flexible way of describing visual concepts and objects. As text is one of the most intuitive types of user input, text-to-image synthesis has gained much attention from the research community and numerous efforts have been made towards developing better text-to-image synthesis models. In the following, we review

state-of-the-art text-to-image synthesis models and discuss recent advances.

4.2.2 *Learning correspondence between text and image representations*

One of the major challenges for the text-to-image synthesis task is that the input text and output image have different modalities, which requires learning of correspondences between text and image representations. This multimodal nature and the need to learn text-to-image correspondences motivated Reed et al. [47] to propose to solve the task using a GAN model. They generate images conditioned on the embedding of text descriptions, instead of class labels as in traditional cGANs [35]. To learn the text embedding from input sentences, a deep convolutional image encoder and a character level convolutional-recurrent text encoder are trained jointly so that the text encoder can learn a vector-representation of the input text descriptions. Adapting the DCGAN architecture [48], the learned text encoding is then concatenated with both the input noise vector in the generator and the image features in the discriminator along with the depth dimension. This method [47] generated encouraging results on both the Oxford-102 dataset [44] and the CUB dataset [45], but the generated images have low resolution (64×64). Other work proposed by Mansimov et al. [49] around the same time as DCGAN proposes a combination of a recurrent variational autoencoder with an attention model which iteratively draws patches on a canvas, while paying attention to the relevant words in the description. Input text descriptions are represented as a sequence of consecutive words and images are represented as a sequence of patches drawn on a canvas. Image generation samples from a Gaussian distribution, whose mean and variance depend on the previous hidden states of the generative LSTM. Experiments on the MS-COCO dataset show reasonable results that correspond well to text descriptions.

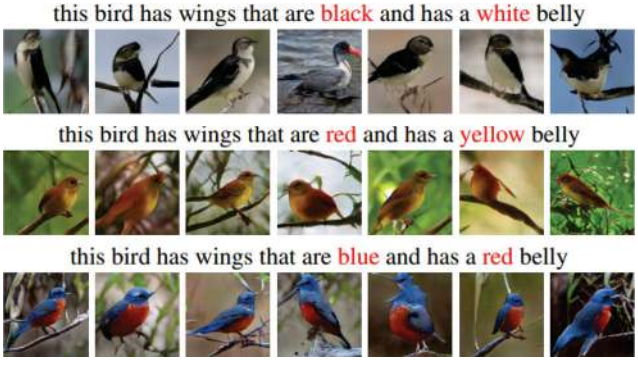

Fig. 2 Bird image synthesis results given text descriptions as input with an attention mechanism. Key words in the input sentences are correctly captured and represented in the generated images. Reproduced with permission from Ref. [25], © IEEE 2018.

To further improve the visual quality and realism of generated images given text descriptions, Zhang et al. proposed multi-stage GAN models, StackGAN [1] and StackGAN++ [50], to enable incremental refinement in the image generation process. Given text descriptions, StackGAN [1] decomposes the text-to-image generative process into two stages: In the first it captures basic object features and background layout, and then in the second it refines details of the objects and generates a higher resolution image. Unlike Ref. [47] which transforms high dimensional text encoding into low dimensional latent variables, StackGAN adopts conditioning augmentation, sampling the latent variables from an independent Gaussian distribution parameterized by the text encoding. Experiments on the Oxford-102 [44], CUB [45], and COCO [46] datasets show that StackGAN can generate compelling images with resolution up to 256×256. In StackGAN++ [50], the authors extended the original StackGAN to a more general and robust model which contains multiple generators and discriminators to handle images at different resolutions. Subsequently, Zhang et al. [51] extended the multi-stage generation idea by proposing an HDGAN model with a single-stream generator and multiple hierarchically-nested discriminators for high-resolution image synthesis. Hierarchically-nested discriminators distinguish outputs from intermediate layers of the generator to capture hierarchical visual features. HDGAN is trained by optimizing a pair loss [47] and a patch-level discriminator loss [52].

In addition to generation via multi-stage refinement [1, 50], the attention mechanism may be introduced to improve text to image synthesis at a finer-grained level. Xu et al. [25] introduced AttnGAN, an attention driven image synthesis model that generates images by focusing on different regions described by different words of the text input. A deep attentional multimodal similarity model (DAMSM) module is also proposed to match the learned embedding between image regions and text at the word level. To achieve better semantic

consistency between text and image, Qiao et al. [2] proposed MirrorGAN which guides image generation with both sentence- and word-level attention and further tries to reconstruct the original text input to guarantee the image–text consistency. The backbone of MirrorGAN uses a multi-scale generator as in Ref. [50]. The proposed text reconstruction model is pre-trained to stabilize the training of MirrorGAN. Zhu et al. [3] introduced a gating mechanism where a writing gate writes selected important textual features from the given sentence into a dynamic memory, and a response gate adaptively reads from the memory and the visual features from some initially generated images. The proposed DM-GAN relies less on the quality of the initial images and can refine poorly-generated initial images having wrong colors and rough shapes.

To learn expression variants in different text descriptions of the same image, Yin et al. [53] proposed SD-GAN to distill the shared semantics from texts that describe the same image. The authors propose a Siamese structure with a contrastive loss to minimize the distance between images generated from descriptions of the same image, and maximize the distance between those generated from the descriptions of different images. To retain semantic diversity for fine-grained image generation, semantically-conditioned batch normalization is also introduced for enhanced visual-semantic embedding.

4.2.3 Location and layout aware generation

With advances in correspondence learning between text and image, content described in the input text can already be well captured in the generated image. However, to achieve finer control of generated images such as object locations, additional inputs or intermediate steps are often required. For text-based, location-controllable synthesis, Reed et al. [54] proposed to generate images conditioned on both the text description and object locations. Built upon the similar idea of inferring scene structure for image generation, Hong et al. [55] introduced a novel hierarchical approach for text-to-image synthesis by inferring semantic layout from the text description. Bounding boxes are first generated from text input through an auto-regressive model, and then semantic layouts are refined from the generated bounding boxes using a convolutional recurrent neural network. Conditional on both the text and the semantic layouts, the authors adopt a combination of pix2pix [52] and the CRN [56] image-to-image translation model to generate the final images. With predicted semantic layouts, this work can potentially generate more realistic images containing complex objects such as those in the MS-COCO [46] dataset. Li et al. [57] extended the work in Ref. [55] and introduced Obj-GAN, which generates salient objects given a text description. Semantic layout is first generated as in Ref. [55] and then later converted into the synthetic image. A Fast R-CNN [58] based object-wise discriminator is developed to retain the matching between generated objects and the input text and layout. Experiments on the MS-COCO dataset show improved performance in generating complex scenes compared to previous methods.

Compared to Ref. [55], Johnson et al. [59] included another intermediate step which converts the input sentences into scene graphs before generating the semantic layouts. A graph convolutional network is developed to generate embedding vectors for each object. Bounding boxes and segmentation masks for each object, constituting the scene layout, are converted from the object embedding vectors. Final images are synthesized by a CRN model [56] from the noise vectors and scene layouts. In addition to text input, Ref. [59] also allows direct generation from input scene graphs. Experiments conducted on the Visual Genome [60] dataset and COCO-Stuff [61], an augmented subset of the MS-COCO [46] dataset, show better depiction of complex sentences with many objects than a previous method [1].

Without taking the complete semantic layout as additional input, Hinz et al. [62] introduced a model consisting of a global pathway and an object pathway for finer control of object location and size within an image. The global pathway is responsible for creating a general layout of the global scene, while the object pathway generates object features within the given bounding boxes. The outputs of the global and object pathways are then combined to generate the final synthetic image. When there is no text description available, Ref. [62] can take a noise vector and individual object bounding boxes as input.

Taking a different approach from GAN based methods, Tan et al. [63] proposed a Text2Scene model for text-to-scene generation, which learns to sequentially generate objects and their attributes such

as location, size, and appearance at every time step. With a convolutional recurrent module and attention module, Text2Scene can generate abstract scenes and object layouts directly from descriptive sentences. For image synthesis, Text2Scene retrieves patches from real images to generate the image composites.

4.2.4 Fusion of conditional and unconditional generation

While most existing text-to-image synthesis models are based on conditional image generation, Bodla et al. [64] proposed a FusedGAN which combines unconditional image generation and conditional image generation. An unconditional generator produces a structure prior independent of the condition, and the other conditional generator refines details and creates an image that matches the input condition. FusedGAN was evaluated on both the text-to-image generation task and the attribute-to-face generation task discussed later in Section 4.4.1.

4.2.5 Evaluation metrics for text to image synthesis

Widely used metrics for image synthesis such as IS [14] lack awareness of matching between the text and generated images. Recently, more effort has been focused on proposing more accurate evaluation metrics for text to image synthesis and for evaluating the correspondence between generated image content and input conditions. R-precision is proposed in Ref. [25] to evaluate whether a generated image is well conditioned on the given text description. Hinz et al. [65] proposed the semantic object accuracy (SOA) score which uses a pre-trained object detector to check whether the generated image contains the objects described in the caption, especially for the MS-COCO dataset. SOA shows better correlation with human perception than IS in the user study and provides better guidance for training text to image synthesis models.

4.2.6 Benchmark datasets

For text-guided image synthesis tasks, popular benchmark datasets include datasets with a single object category and datasets with multiple object categories. For the former, the Oxford-102 dataset [44] contains 102 different types of flowers common in the UK. The CUB dataset [45] contains photos of 200 bird species mostly from North America. Datasets with multiple object categories and complex relationships can be used to train models for more challenging image synthesis tasks. One such dataset is MS-COCO [46], which has a training set with 80k images and a validation set with 40k images. Each image in the COCO dataset has five text descriptions.

4.3 Image-like inputs

4.3.1 Commonality

In this section, we summarize image synthesis works based on three types of intuitive inputs, namely sketches, semantic maps, and pose. We call them image-like inputs because all of them can be, and have been, represented as rasterized images. Therefore, synthesizing images from these image-like inputs can be regarded as an image-to-image translation problem. Several works provide general solutions to this problem, like pix2pix [52] and pix2pixHD [66]. In this survey, we focus on works that deal with a specific type of input.

4.3.2 Sketches and strokes as input

Sketches, or line drawings, can be used to express users' intent in an intuitive way, even for those without professional drawing skills. With the widespread use of touch screens, it has become very easy to create sketches; and the research community is paying increasingly more attention to the understanding and processing of hand-drawn sketches, especially in applications such as sketch-based image retrieval and sketch-to-image generation. Generating realistic images from sketches is not a trivial task, since the synthesized images need to be aligned spatially with the given sketches, while maintaining semantic coherence.

(1) Retrieval-and-composition based approaches

Early approaches for generating images from sketches mainly took a retrieval-and-composition strategy as illustrated in Fig. 3. For each object in the user-given sketch, they searched for candidate images in a pre-built object-level image (fragment) database, using some similarity metric to evaluate how well the sketch matched the image. The final image is synthesized by composition of retrieved results, mainly by image blending algorithms. Chen et al. [33] presented a system called Sketch2Photo, which composes a realistic image from a simple free-hand sketch annotated with text labels. The authors proposed a contour-based filtering scheme to search for appropriate photographs according to the given sketch and text labels, and a novel hybrid blending algorithm combining alpha blending and Poisson

blending, to improve the synthesis quality. Eitz et al. [67] created Photosketcher, a system that finds semantically relevant regions from appropriate images in a large image collection and composes the regions automatically. Users can also interact with the system by drawing scribbles on the retrieved images to improve region segmentation quality, re-sketching to find better candidates, or choosing from different blending strategies. Hu et al. [68] introduced PatchNet, a hierarchical representation of image regions that summarizes a homogeneous image patch by a graph node and represents geometric relationships between regions by labeled graph edges. PatchNet was shown to be a compact representation that can be used efficiently for sketch-based, library-driven, interactive image editing. Wang et al. [69] proposed a sketch-based image synthesis method that compares sketches with contours of object regions via the GF-HoG descriptor; novel images are composited by GrabCut followed by Poisson blending or alpha blending. For generating images of a single object like an animal with user-specified pose and appearance, Turmukhambetov et al. [70] presented a sketch-based interactive system that generates the target image by composing patches of nearest neighbour images on the joint manifold of ellipses and contours for object parts.

(2) Deep learning based approaches

In recent years, deep convolutional neural networks (CNNs) have achieved significant progress in image-related tasks. CNNs have been used to map sketches to images with the benefit of being able to synthesize novel images different from those in pre-built databases. One challenge to using deep CNNs is that training such networks requires paired sketch–image data, which can be expensive to acquire. Hence, various techniques have been proposed to generate synthetic sketches from images, and then use the synthetic sketch and image pairs for training. Methods for synthetic sketch generation include boundary detection algorithms such as Canny or holistically-nested edge detection (HED) [71], and stylization algorithms for image-to-sketch conversion [72–76]. Post-processing steps are adopted for small stroke removal, spline fitting [77], and stroke simplification [78]. A few works utilize crowd-sourced free-hand sketches for training [8, 9]. They either construct pseudo-paired data by matching sketches and images [8], or propose a method that does not require paired data [9]. Another aspect of CNN training that has been investigated is the representation of sketches. In some works [79, 80], the input sketches are transformed into distance fields to obtain a dense representation, but no experimental comparisons have been done to demonstrate which form of input is more suitable for CNN processing. Next, we review specific works that utilize a deep-learning based approach for sketch to image generation.

Treating a sketch as an image-like input, several works use a fully convolutional neural network architecture to generate photorealistic images. Güçlütürk et al. [81] first attempted to use deep neural networks to tackle the problem of sketch-based synthesis. They developed three different models to generate face images from three different types of sketches: line sketches, grayscale sketches, and color sketches. An encoder–decoder fully convolutional

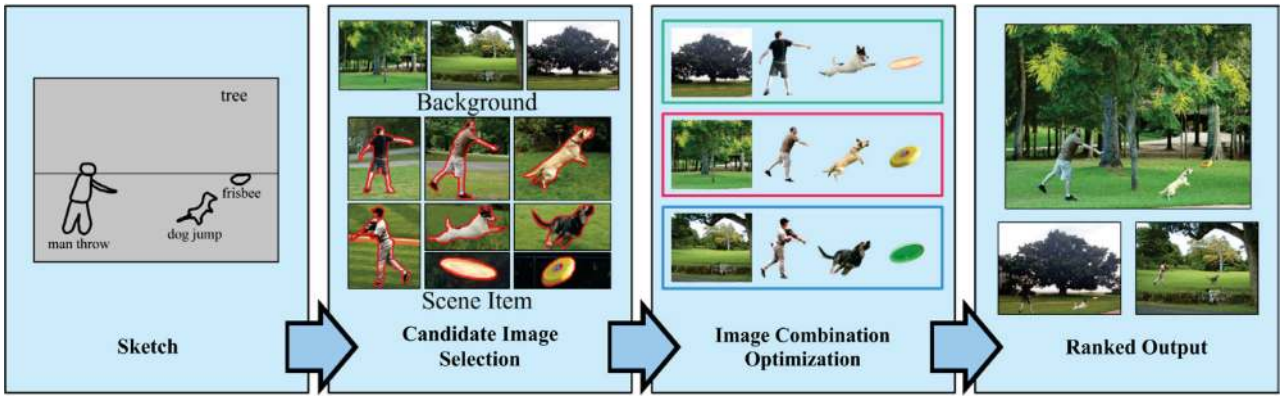

Fig. 3 A classical pipeline of retrieval-and-composition methods for synthesis. Candidate images are generated by composing image segments retrieved from a pre-built image database. Reproduced with permission from Ref. [33], © ACM 2009.

neural network is adopted and trained with various loss terms. A total variation loss is proposed to encourage smoothness. Sangkloy et al. [6] proposed Scribbler, a system that can generate realistic images from human sketches and color strokes. An XDoG filter is used for boundary detection to generate image–sketch pairs and color strokes are sampled to provide color constraints in training. The authors also use an encoder–decoder network architecture and adopt similar loss functions to those in Ref. [81]. Users can interact with the system in real time. The authors also provide applications for colorization of grayscale images.

Generative adversarial networks have also been used for sketch-to-image synthesis. Chen and Hays [79] proposed a novel GAN-based architecture with multi-scale inputs for the problem. The generator and discriminator both consist of several masked residual unit (MRU) blocks. An MRU takes in a feature map and an image, and outputs a new feature map, which can allow a network to repeatedly condition on an input image, like a recurrent network. They also adopt a novel data augmentation technique, which generates sketch–image pairs automatically through edge detection and post-processing steps including binarization, thinning, small component removal, erosion, and spur removal. To encourage diversity of generated images, the authors proposed a diversity loss, which maximizes the $L1$ distance between the outputs of two identical input sketches with different noise vectors. Lu et al. [26] considered the sketch-to-image synthesis problem as an image completion task and proposed a contextual GAN for the task. Unlike a traditional image completion task where only part of an object is masked, the entire real image is treated as the missing piece in a joint image that consists of both sketch and the corresponding photo. The advantage of using such a joint representation is that, instead of using the sketch as a hard constraint, the sketch part of the joint image serves as a weak contextual constraint. Furthermore, the same framework can also be used for image-to-sketch generation where the sketch would be the masked or missing piece to be completed. Ghosh et al. [7] presented an interactive GAN-based sketch-to-image translation system. As the user draws a sketch of a desired object type, the system automatically recommends completions and fills the shape with class-conditioned texture. The result changes as the user adds or removes strokes over time, which enables a feedback loop that the user can leverage for interactive editing. The system consists of a shape completion stage based on a non-image generation network [82], and a class-conditioned appearance translation stage based on the encoder–decoder model from MUNIT [83]. To perform class-conditioning more effectively, the authors propose a soft gating mechanism, instead of using simple concatenation of class codes and features.

Several works focus on sketch-based synthesis of human face images. Portenier et al. [84] developed an interactive system for face photo editing. The user can provide shape and color constraints by sketching on the original photo, to edit it. Editing is done by a CNN, which is trained on randomly masked face photos with sampled sketches and color strokes in an adversarial manner. Xia et al. [85] proposed a two-stage network for sketch-based portrait synthesis. The stroke calibration network is responsible for converting the input poorly-drawn sketch to a more detailed and calibrated one that resembles an edge map. Then the refined sketch is used in the image synthesis network to produce a photo-realistic portrait image. Li et al. [80] proposed a self-attention module to capture long-range connections of sketch structures, where a self-attention mechanism is adopted to aggregate features from all positions of the feature map using the calculated self-attention map. A multi-scale discriminator is used to distinguish patches of different receptive fields, to simultaneously ensure local and global realism. Chen et al. [86] introduced DeepFaceDrawing, a local-to-global approach for generating face images from sketches that uses input sketches as soft constraints and is able to produce high-quality face images even from rough or incomplete sketches. The key idea is to learn feature embeddings of key face components and then train a deep neural network to map the embedded component features to realistic images.

While most works in sketch-to-image synthesis with deep learning techniques have focused on synthesizing object-level images from sketches, Gao et al. [8] explored synthesis at the scene level by proposing a deep learning framework for scene-level image generation from freehand sketches. The framework first segments the sketch into individual objects,

recognizes their classes, and categories them into foreground/background objects. Then the foreground objects are generated by an EdgeGAN module that learns a common vector representation for images and sketches and maps the vector representation of an input sketch to an image. The background generation module is based on the pix2pix [52] architecture. The synthesized foregrounds along with background sketches are fed to a network to get the final generated scene. To train the network and evaluate their method, the authors constructed a composite dataset called SketchyCOCO based on the Sketchy database [87], Tuberlin dataset [88], QuickDraw dataset, and COCO Stuff [89].

As collecting paired training data can be labor intensive, learning from unpaired sketch-photo data in an unsupervised setting is an interesting direction to explore. Liu et al. [9] proposed an unsupervised solution by decomposing the synthesis process into a shape translation stage and a content enrichment stage. The shape translation network transforms an input sketch into a gray-scale image, trained using unpaired sketches and images, under the supervision of a cycle-consistency loss. In the content enrichment stage, a reference image can be provided as style guidance, whose information is injected into the synthesis process following the AdaIN framework [90].

(3) Benchmark datasets

For synthesis from sketches, various datasets covering multiple types of objects are used [45, 46, 87, 89, 92–98]. However, only a few [87, 92, 98] have paired image and sketch data. For the other datasets, edge maps or line strokes are extracted using edge extraction or style transfer techniques and used as ersatz sketch data for training and validation. SketchyCOCO [8] built a paired image–sketch dataset from existing image datasets [89] and sketch datasets [87, 88] by looking for the most similar sketch with the same class label for each foreground object in a natural image.

4.3.3 Semantic label maps as input

(1) Background

Synthesizing photorealistic images from semantic label maps is the inverse problem of semantic image segmentation. It has applications in controllable image synthesis and image editing. Existing methods either work with a traditional retrieval-and-composition approach [99, 100], a deep learning based method [101–106], or a hybrid of the two [107]. Different types of datasets are utilized to allow synthesis of images of various scenes or subjects, such as indoor and outdoor scenes, or human bodies.

(2) Retrieval-and-composition based methods

Non-parametric methods follow the traditional retrieval-and-composition strategy. Johnson et al. [99] first proposed synthesizing images from semantic concepts. Given an empty canvas, the user can paint regions with corresponding keywords at desired locations. The algorithm searches for candidate images in the stock and uses a graph-cut based seam optimization process to generate realistic photographs for each combination. The best combination with the minimum seam cost is chosen as the final result. Bansal et al. [100] proposed a non-parametric matching and hierarchical composition strategy to synthesize realistic images from semantic maps. The strategy has four stages: a global consistency stage to retrieve relevant samples based on indicator vectors of presented categories, a shape consistency stage to find candidate segments based on shape context similarity between the input label mask and the ones in the database, and a part consistency stage and a pixel consistency stage that re-synthesize patches and pixels based on best-matching areas as measured by Hamming distance. The proposed method outperforms state-of-the-art parametric methods like pix2pix [52] and pix2pixHD [66] both qualitatively and quantitatively.

(3) Deep learning based methods

Methods based on deep learning mainly vary in network architecture design and optimization objective. Chen and Koltun [101] proposed a regression approach for synthesizing realistic images from semantic maps, without the need for adversarial training. To improve synthesis quality, they proposed a cascaded refinement network (CRN), which progressively generates images from low resolution to high resolution (up to 1024×2048 pixels) through a cascade of refinement modules. To encourage diversity in generated images, the authors proposed a diversity loss, which lets the network output multiple images at once and optimizes diversity within the collection. Wang et al. [108] proposed a style-consistent GAN framework that generates images given a semantic label map input and an example image indicating style. A novel style-

consistency discriminator is designed to determine whether a pair of images have consistent style and an adaptive semantic consistency loss is optimized to ensure correspondence between the generated image and input semantic label map.

Having found that directly synthesizing images from semantic maps through a sequence of convolutions sometimes provides unsatisfactory results because of semantic information loss during forward propagation, some work seeks to better use the input semantic map and preserve semantic information in all stages of the synthesis network. Park et al. [103] proposed a spatially-adaptive normalization layer (SPADE) with learnable parameters that utilizes the original semantic map to help retain semantic information in the feature maps after traditional batch normalization. The authors incorporated their SPADE layers into the pix2pixHD architecture and produced state-of-the-art results on multiple datasets. Liu et al. [104] argued that a convolutional network should be sensitive to semantic layouts at different locations. Thus they proposed conditional convolution blocks (CC Block), where parameters for convolution kernels are predicted from semantic layouts. They also proposed a feature pyramid semantic-embedding (FPSE) discriminator, which predicts semantic alignment scores in addition to real versus fake scores. It explicitly forces the generated images to be better aligned semantically with the given semantic map. Zhu et al. [105] proposed a group decreasing network (GroupDNet). It utilizes group convolutions in the generator; the number of groups in the decoder decreases progressively. Inspired by SPADE, the authors also proposed a novel normalization layer to make better use of information in the input semantic map. Experiments show that the GroupDNet architecture is more suitable for multi-modal image synthesis, and can produce plausible results.

Observing that results from existing methods often lack detailed local texture, resulting from large objects dominating the training, Tang et al. [106] aimed to better synthesize small objects in the image. In their design, each class has its own class-level generation network that is trained with feedback from a classification loss; all classes share an image-level global generator. The class-level generator generates parts of the image that correspond to each class, from masked feature maps. All the class-specific image parts are then combined and fused with the image-level generation result. In other work, to provide more fine-grained interactivity, Zhu et al. [91] proposed semantic region-adaptive normalization (SEAN), which allows manipulation of each semantic region individually, to improve image quality. A qualitative comparison of different deep learning based methods is shown in Fig. 4.

(4) Integrative methods

While deep learning based generative methods are

Fig. 4 Image synthesis from semantic label maps. Reproduced with permission from Ref. [91], © IEEE 2020.

better able to synthesize novel images, traditional retrieval-and-composition methods generate images with more reliable texture and fewer artifacts. To combine the advantages of both parametric and non-parametric methods, Qi et al. [107] presented a semi-parametric approach. They built a memory bank offline, containing segments of different classes of objects. Given an input semantic map, segments are first retrieved using a similarity metric defined by IoU score of the masks. The retrieved segments are fed to a spatial transformer network where they are aligned, and further put onto a canvas by an ordering network. The canvas is refined by a synthesis network to give the final result. This combination of retrieval-and-composition and deep-learning based methods allows high-fidelity image generation, but it takes more time during inferencing and the framework is not end-to-end trainable.

(5) Benchmark datasets

For synthesis from semantic label maps, experiments are mainly conducted on datasets of the human body [109–111] or face [112], or indoor [113–115] or outdoor scenes [116]. Lassner et al. [102] augmented the Chictopia10K [109, 110] dataset by adding 2D keypoint locations and fitted SMPL body models, and the augmented dataset was used by Bem et al. [117]. Park et al. [103] and Zhu et al. [91] collected images from the Internet and applied state-of-the-art semantic segmentation models [118, 119] to build paired datasets.

4.3.4 Poses as input

(1) Background

Given a reference person image, its corresponding pose, and a novel pose, pose-based image synthesis methods can generate an image of the person in that novel pose. Unlike synthesizing images from sketches or semantic maps, pose-guided synthesis requires novel views to be generated, which cannot be done by a retrieval and composition pipeline. Thus we focus on reviewing deep learning-based methods [27–31, 117, 120–123]. In these methods, a pose is often represented as a set of well-defined body keypoints. Each keypoint can be modeled as an isotropic Gaussian that is centered at the ground-truth joint location with a small standard deviation, giving rise to a heatmap. The concatenation of the joint-centered heatmaps then can be used as input to the image synthesis network. Heatmaps of rigid parts and the whole body can also be utilized [117].

(2) Supervised deep learning methods

In a supervised setting, ground-truth target images in target poses are required for training. Thus, datasets with the same person in multiple poses are needed. Ma et al. [27] proposed a pose guided person generation network for generating person images in given poses. It adopts a GAN-like architecture and generates images in a coarse-to-fine manner. In the coarse stage, an image of a person along with a novel pose are fed into the U-Net based generator, where the pose is represented as heatmaps of body keypoints. The coarse output is then concatenated again with the person image, and a refinement network is trained to learn a difference map that can be added to the coarse output to give the final refined result. The discriminator is trained to distinguish synthetic outputs from real images. Besides the GAN loss, an $L1$ loss is used to measure dissimilarity between the generated output and the target image. Since the target image may have different background from the input condition image, the $L1$ loss is modified to give higher weight to the human body, utilizing a pose mask derived from the pose skeleton.

Although GANs have achieved great success in image synthesis, there are still difficulties when it comes to pose-based synthesis, one of which is the deformation problem. The given novel pose can be drastically different from the original pose, resulting in large deformations in both shape and texture in the synthesized image, making it hard to directly train a network that can generate images without artifacts. Existing work mainly adopts transformation strategies to overcome this problem, because transformations make explicit which body part moves to which place, given original and target poses. These methods usually transform body parts of the original image [120], the human parsing map [122], or the feature map [29, 31, 122]. Balakrishnan et al. [120] explicitly separated the human body from the background and synthesized person images of unseen poses and background in separate steps. Their method consists of four modules: a segmentation module that produces masks of the whole body and each body part based on the source image and pose, a transformation module that calculates and applies an affine transformation to each body part and corresponding feature maps, a

background generation module that applies inpainting to fill the body-removed foreground region, and a final integration module that uses the transformed feature maps and the target pose to produce the synthesized foreground, which is then combined with the inpainted background to give the final result. To train the network, VGG-19 perceptual loss is used along with GAN loss. Siarohin et al. [29] noted that it is hard for the generator to directly capture large body movements because of the restricted receptive field, and introduced deformable GANs to tackle the problem. The method decomposes the body joints into several semantic parts, and calculates an affine transform from the source to the target pose for each part. The affine transforms are used to align the feature maps of the source image with the target pose. The transformed feature maps are then concatenated with the target pose features and decoded to synthesize the output image. The authors also proposed a novel nearest-neighbor loss based on feature maps, instead of using $L1$ or $L2$ loss. Their method is more robust to large pose changes and produces higher quality images than Ref. [27]. Dong et al. [122] utilized parsing results as a proxy to achieve better synthesis results. They first estimate parsing results for the target pose, and then fit a thin plate spline (TPS) transformation between the original and estimated parsing maps. The TPS transformation is further applied to warp the feature maps for feature alignment and a soft-gated warping block is used to provide controllability to the transformation. The final image is synthesized using the transformed feature maps. Zhu et al. [31] proposed to divide large deformations into a sequence of small deformations, which are more amenable to network training. In this way, the original pose can be transformed progressively, through many intermediate poses. They proposed a pose-attentional transfer block (PATB), which transforms the feature maps under the guidance of an attention mask. By stacking multiple PATBs, the feature maps undergo several transformations and the transformed maps are used to synthesize the final result.

While most deep learning based methods for synthesis from poses adopt an adversarial training paradigm, Bem et al. [117] proposed a conditional-VAEGAN architecture that combines a conditional-VAE framework and a GAN discriminator module to generate realistic natural images of people in a unified probabilistic framework, where the body pose and appearance are kept as separate interpretable variables, allowing the sampling of people with independent variations of pose and appearance. The loss function used includes both conditional-VAE and GAN losses including $L1$ reconstruction loss, closed-form KL-divergence loss between recognition and prior distributions, and discriminator cross-entropy loss.

(3) Unsupervised deep learning methods

The aforementioned pose-to-image synthesis methods require ground-truth images in target poses for training because of their use of $L1$, $L2$ or perceptual losses. To eliminate the need for target images, some works consider an unsupervised setting of this problem [30, 121], where the training process does not require ground-truth images of the target pose. The basic idea is to ensure cycle consistency. After the forward pass, the synthesized results along with the target pose are treated as the reference, and used to synthesize the image in the original reference pose. This synthesized image should be consistent with the original reference image. Pumarola et al. [121] further utilized a pose estimator, to ensure pose consistency. Song et al. [30] used parsing maps as supervision instead of poses. They predict parsing maps under new target poses and use them to synthesize the corresponding images. Since the parsing maps in the target poses are not available due to operating in an unsupervised setting, the authors proposed a pseudo-label selection technique to provide ersatz parsing maps by searching for ones with the same type of clothes and minimum transformation energy.

(4) Benchmark datasets

For synthesis from poses, the DeepFashion [111] and Market-1501 [124] datasets are most widely used. The former is built for clothes recognition but has also been used for pose-based image synthesis because of its rich annotation, such as clothing landmarks, as well as images with corresponding foreground but diverse backgrounds. The Market-1501 dataset was initially introduced for the purpose of person re-identification, and contains a large number of person images produced using a pedestrian detector, with annotated bounding boxes; also, each identity has multiple images from different camera views.

4.4 Other input modalities

Apart from text descriptions and image-like inputs, other intuitive user inputs exist, such as class labels, attribute vectors, and graph-like inputs.

4.4.1 Visual attributes as input

In this subsection, we mainly focus on works that use one of the fine-grained class conditional labels or vectors, i.e., visual attributes, as inputs. Visual attributes provide a simple and accurate way of describing major features in images, e.g., describing attributes of a certain category of birds or details of a person's face. Current methods either take a discrete one-hot vector as attribute labels, or a continuous vector as visual attribute input.

Yan et al. [125] proposed a disentangling CVAE (disCVAE) for conditioned image generation from visual attributes. A conditional variational auto-encoder (cVAE) [39] generates images from a posterior conditioned on both the conditions and random vectors, while disCVAE interprets an image as a composite of a foreground layer and a background layer. The foreground layer is conditioned on visual attributes and the whole image is generated through gated integration. Attribute-conditioned experiments are often conducted on the LFW [126] and CUB [45] datasets.

An application of face generation with visual attribute inputs is to manipulate existing face images. AttGAN [127] applies an attribute classification constraint and reconstruction learning to guarantee changes in desired attributes while maintaining other details. Zhang et al. [128] proposed using spatial attention which can localize attribute-specific regions to perform desired attribute manipulation while keeping the rest unchanged. Unlike other work utilizing attribute input, Qian et al. [129] explored face manipulation via conditional structure input. Given a structure prior as conditional input to a cVAE, AF-VAE [129] can arbitrarily modify facial expressions and head poses using geometry-guided feature disentanglement and an additive Gaussian mixture prior for appearance representation. Most such face image manipulation work performs experiments on commonly used face image datasets such as the CelebA [96] dataset.

For controllable person image synthesis, Men et al. [130] introduced an attribute-decomposed GAN, where visual attributes, including clothes, are extracted from reference images and combined with target poses to generate target images with desired attributes. The separation and decomposition of attributes from existing images provide a new way of synthesizing person images without attribute annotations.

Another interesting application of taking visual attributes as input is fashion design. Lee and Lee [131] proposed a GAN model with an attentional discriminator for attribute-to-fashion generation. For multiple-attribute inputs, multiple independent Gaussian distributions are derived by mapping each attribute vector to the mean vector and diagonal covariance matrix. The prior distribution for attribute combination is the product of all independent Gaussians. Experiments were conducted on a dataset consisting of dress images collected from a popular fashion site.

In terms of image generation methodology using visual attributes as inputs, the Glow model introduced in Ref. [21], a generative flow model using an invertible 1×1 convolution, shows great potential. Compared with VAEs and GANs, flow models have merits including reversible generation, meaningful latent space, and memory efficiency. Glow consists of a series of steps of flow, where each step consists of activation normalization followed by an invertible 1×1 convolution, followed by a coupling layer. On the Cifar10 dataset, Glow achieves better negative log likelihood than RealNVP [132]. On the CelebA-HQ dataset, Glow generates high fidelity face images and also allows meaningful visual attribute manipulation.

For attribute-guided synthesis tasks, major benchmark datasets include the Visual Genome, CelebA(-HQ), and Labeled Faces in the Wild. The Visual Genome [60] contains over 100k images in which each image has an average of 21 objects, 18 attributes, and 18 pairwise relationships between objects. The CelebA [96] dataset has a 40 dimensional binary attribute vector annotated for each face image. The CelebA-HQ dataset [97] consists of 30,000 high resolution images from the CelebA dataset. The Labeled Faces in the Wild (LFW) dataset contains face images that are segmented and labeled with semantically meaningful region labels (e.g., hair, skin).

4.4.2 *Graphs and layouts as input*

Another interesting type of intuitive user input is graphs (see Fig. 5). Graphs can encode multiple relationships in a concise way and have distinctive characteristics such as sparse representation. An example application of graph-based inputs is architectural design using scene graphs, layouts, and other similar modalities.

Johnson et al. [59], as mentioned earlier in Section 4.2.3, can take a scene graph and generate a corresponding layout. The final image is then synthesized by a CRN model [56] from a noise vector and the layout. Figure 5 demonstrates some results.

To generate images that exhibit complex relationships between multiple objects, Zhao et al. [133] proposed a Layout2Im model that uses layout as input to generate images. The layout is specified by multiple bounding boxes of objects with category labels. Training of the model is done by taking ground-truth images with their layouts, and testing is done by sampling object latent codes from a normal distribution. An object composer takes the word embedding of input text, object latent code, and bounding box locations to produce object feature maps. The object feature maps are then composed using convolutional LSTM into a hidden feature map and decoded into the final image.

Also containing the idea of converting layout to image, LayoutGAN [10] uses a differentiable wireframe rendering layer with an image-based discriminator that can generate layout from graphical element inputs. Semantic and spatial relations between elements are learned via a stacked relation module with self attention; experiments on various datasets show promising results in generating meaningful layouts which can be also rasterized.

Luo et al. [134] proposed a variational generative model which generates 3D scene layouts given input scene graphs. cVAE is combined with a GCN [13] for layout synthesis. The authors also present a rendering model which first instantiates a 3D model by retrieving object meshes, and then utilizes a differentiable renderer to render the corresponding semantic image and depth image. Their experiments on the SUNCG dataset [135] show that the method can generate accurate and diverse 3D scene layouts,

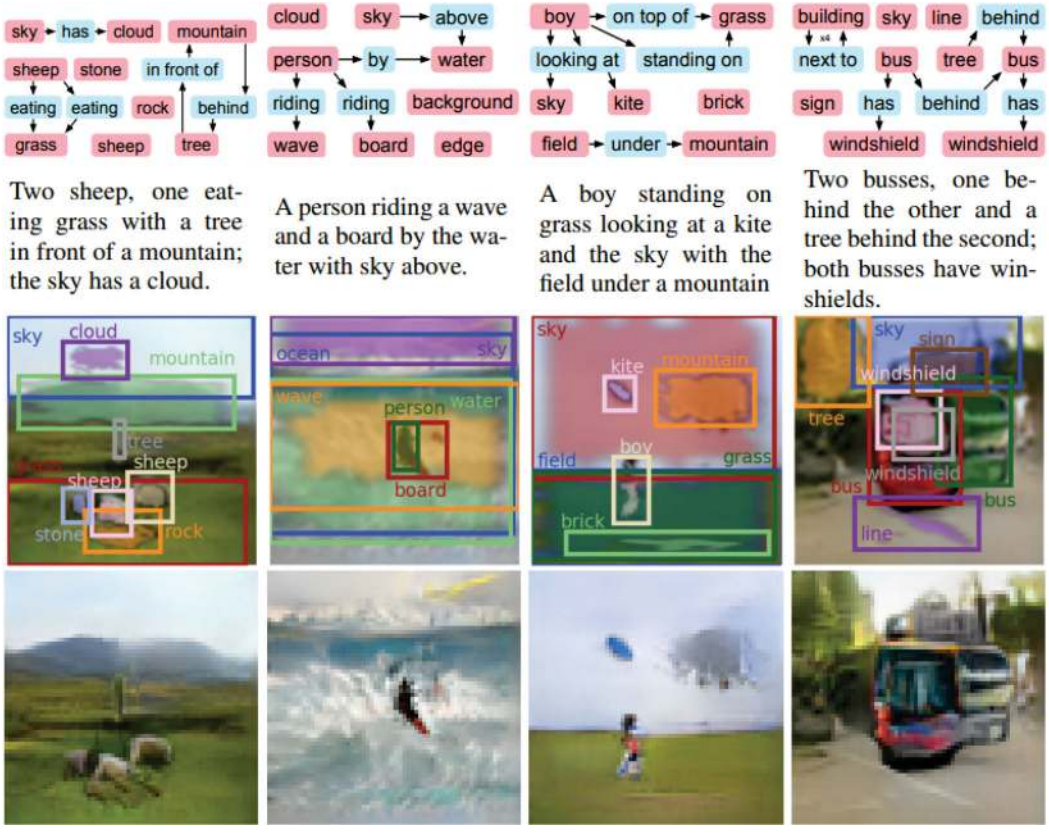

Fig. 5 Scene graph to image synthesis results. Scene graphs are often extracted from text descriptions. Correct object relationships embedded in input scene graphs are reflected in the generated images. Reproduced with permission from Ref. [59], © IEEE 2018.

and has potential for various downstream scene layout and image synthesis tasks.

5 Summary and trends

5.1 Advances in model architecture design and training strategy

Among different attempts to improve the quality of the synthesized image and the correspondence between user input and generated image, several successful designs have been incorporated into multiple conditional generative models and have proven their effectiveness in various tasks. For instance, a hierarchical generation architecture has been widely used by different models, including GANs [50, 66, 136] and VAEs [137], in order to generate high-resolution, high-quality images in a multi-stage, progressive fashion. Attention-based mechanisms have been incorporated in various work [25, 138] to give finer-grained control over regions within generated images. To ensure correspondence between user input and generated images, various designs have been proposed for generative neural networks: relatively straightforward methods combine user input and other input (e.g., a latent vector) as input to the generative model, other methods take the user input as part of the supervision signal to measure the correspondence between input and output, and more advanced methods, which may also be more effective, combine transformed inputs, e.g., in a projection discriminator [36] and spatially-adaptive normalization [103].

While most current successful models are based on GANs, it is well-known that training GANs is difficult and can be unstable. As for general purpose GANs, works focusing on image synthesis with intuitive user inputs adopt different design and training strategies to ease and stabilize GAN training. Commonly used normalisations include conditional batch normalization [139] and spectral normalization [140]; commonly used adversarial losses include WGAN loss with various regularizations [141, 142], LS-GAN loss [143], and hinge loss [144]. To balance training of the generator and the discriminator, imbalanced training strategies such as two time-scale update rule (TTUR) [16] have also been adopted for better convergence.

General losses employed in different models heavily depend on the methodological framework. Retrieval and composition methods typically do not need to be trained, so no loss is used. For GAN-like models, an adversarial loss is essential in a majority of the models, combining a loss for the generator and a loss for the discriminator in order to push the generator toward generating fake samples that match the distribution of real examples. Widely used adversarial losses include the minimax loss introduced in the original GAN paper [34] and the Wasserstein loss introduced in the WGAN paper [141]. VAE models are typically trained by minimizing a reconstruction error between the encoder–decoded data and the initial data, with some regularization of the latent space [38]. To evaluate the visual quality of generated images and help provide better image quality, perceptual loss [145] or adversarial feature matching loss [14] has been adopted by many existing works, especially when a paired supervision signal is available.

Alongside general losses, auxiliary losses are often incorporated in models to better handle different tasks. Task-specific losses, as well as evaluation metrics, are natural choices to evaluate and improve task-specific performance. Depending on the output modality, one commonly used loss or metric considers recovering the input condition from the synthesized images. For instance, image captioning losses can be included in text-to-image synthesis models [2], and pose prediction losses can complement general losses in pose-to-image synthesis tasks [30, 121].

5.2 Summary of methods using specific input types

Recent advances in text-to-image synthesis have been mainly based on deep learning methods, especially GANs. Two major challenges in text-to-image synthesis are learning the correspondence between text descriptions and generated images, and ensuring the quality of generated images. The text–image correspondence problem has been addressed in recent years with advanced embedding techniques for text descriptions and special designs such as attention mechanisms used to match words and image regions. High quality generated images, however, are still limited to narrow categories of objects. For general scenes where multiple objects co-exist with complex relationships, the realism and diversity of the generated images are unsatisfactory and need improvement. To reduce the difficulty of synthesizing

complex scenes, current models may benefit from leveraging different methods such as combining retrieval-and-composition with deep learning, and relationship learning which uses relation graphs as auxiliary input or intermediate steps.

For image-like inputs, one can use a traditional retrieval-and-composition strategy or adopt recent deep learning based methods. The former has several advantages. First, its outputs contain fewer artifacts because objects are retrieved rather than synthesized. Second, it allows user intervention at all stages of the workflow, bringing controllability and customizability. Third, it can be directly applied to a new dataset, without the need for time-consuming training or adaptation. In comparison, deep learning based methods are less interpretable and user intervention at all stages of the synthesis process is more difficult. Although attempts have been made to combine the advantages of both approaches [107], deep learning based methods still dominate for their versatility and ability to generate completely novel images. In these deep learning based methods, inputs are usually represented using grid structures like rasterized images (e.g., for sketches) or multi-channel tensors (e.g., for poses, semantic maps), to simplify utilizing convolution based neural networks. Methods for different input types also have their own idiosyncrasies. Sketch-based synthesis work has attempted to bridge the gap between synthetic sketches and real free-hand sketches, because the latter are hard to collect; synthetic sketches can be used to satisfy the needs of training large networks. For synthesis using semantic maps as input, most progress is in the design of network architectures to better utilize information in the input. For pose-based synthesis, various proposals address problems caused by large deformations between source and target poses, including performing explicit transformations, learning pixel-level correspondence, and synthesis through a sequence of small deformations. Efforts have also been made to alleviate the need for ground-truth data in supervised learning. For example, in pose-based synthesis, the supervised setting requires multiple images of the same person in different poses with the same background; however, often we have an image collection with only one image per person. Some methods [9, 30, 121] work in an unsupervised setting, where no ground-truth of the synthesized result is needed; they mainly work by constraining cycle consistency, with extra supervision for intermediate outputs.

For image synthesis from visual attributes, applications are mainly in face synthesis, person synthesis, and fashion design. Since attributes are an intuitive type of user input suitable for interactive synthesis, we believe that more applications could be explored and more advanced models proposed. One bottleneck for current visual attribute based synthesis tasks is that attribute-level annotation is often required for supervised training. For datasets with no attribute-level annotations, unsupervised attribute disentanglement or attribute-related prior knowledge need to be incorporated into the model design to guarantee that the generated images have the correct attributes.

Image synthesis with graphs as input can better encode relationships between objects than other intuitive user inputs. Current work often relies on graph neural networks [13, 146] to learn graph and node features. In addition to graph input, current methods also try to generate scene graphs as intermediate output from other input modalities such as text descriptions. Applications of graphs as intuitive input include architectural design and scene synthesis requiring the preservation of specific object relationships. While few works have consider image synthesis with graphs, we believe it has great potential for generating scenes with multiple objects, complex relationships, and structural constraints.

5.3 Summary of benchmark datasets

To facilitate finding datasets for particular tasks or particular types of input, we summarize datasets popularly used for various image synthesis tasks with intuitive user inputs in Table 1. State-of-the-art image synthesis methods have achieved high-quality results using datasets containing single object categories such as cars [93], birds [45], and human faces [96–98, 112]. When synthesizing images containing multiple object categories and complex scene structure, there is still room for improvement; datasets such as MS-COCO [46] provide a suitable basis. Future work could also focus more on synthesis with intuitive and interactive user inputs, as well as applications of synthesis methods to real-world scenarios.

Table 1 Commonly used datasets in image synthesis tasks with intuitive user inputs. Annotation types include **L**abel, **A**ttribute, **P**air, **K**ey**P**oint, **B**ounding **B**ox, **S**emantic map, **R**elationship, **T**ext, **V**isual **Q**uestion **A**nswers, **D**epth map, 3D **SC**an. Tasks values are **TE**xt, **P**ose, **SK**etch, **SE**mantic map, **AT**tributes, **S**cene **G**raph, **LA**yout

Dataset name	# images	Categories	Annotations	Tasks	Used in
Shoe V2 [92]	8,648[a]	shoe	P	SK	[9]
Stanford's Cars [93]	16,185	car	L,BB	SK	[26]
UT Zappos50K [94, 95]	50,025	shoe	L,P	SK	[7]
Caltech-UCSD Birds 200 [45]	6,033	bird	L,A,BB,S	TE, SK	[1–3, 25, 26, 47, 50, 51, 53, 64, 125]
Oxford-102 [44]	8,189	flower	L	TE	[1–3, 25, 47, 50, 51]
Labeled Faces in the Wild [126]	13,233	face	L,S	AT	[125, 128]
CelebA [96]	202,599	face	L,A,KP	SK, AT	[26, 64, 127–129]
CelebA-HQ [97]	30,000	face	L,A,KP	SK, AT	[21, 80, 84]
Sketchy [87]	87,971[b]	objects	L,P	SK	[79]
CUHK Face Sketch [98]	1,212[c]	face	P	SK	[6, 81, 85]
COCO [46]	330,000	objects	BB,S,KP,T	TE,SK,SE	[1–3, 25, 49–51, 53, 55, 57, 63, 65, 69, 100]
COCO-Stuff [89]	164,000	objects	S,C	SK,SE,SG,LA	[8, 59, 103, 104, 133]
CelebAMask-HQ [112]	30,000	face	S	SE	[91]
Cityscapes [116]	25,000	outdoor scene	S	SE	[91, 101, 103–107]
ADE20K [113, 114]	22,210	indoor scene	S	SE	[91, 103–107]
NYU Depth [115]	1,449	indoor scene	S,D	SE	[101, 107]
Chictopia10K [109, 110]	17,706	human	S	SE	[102]
DeepFashion [111]	52,712	human	L,A,P,KP	SE,P,AT	[27–31, 105, 121–123, 130]
Market-1501 [124]	32,668	human	L,A	P	[27–31, 122, 123]
Human3.6M [147]	3,600,000	human	KP,BB,S,SC	P	[117]
Visual Genome [60]	108,077	objects	BB,A,R,T,VQA	SG,LA	[59, 133]

[a] 2000 real images and 6648 sketches.
[b] 12,500 real images and 75,471 sketches.
[c] 606 pairs of face photo and corresponding sketch.

6 Perspectives

Having reviewed recent work on image synthesis given intuitive inputs, we discuss in this section perspectives on future research, related to input versatility, generation methodology, benchmark datasets, and evaluation metrics.

6.1 Input versatility

6.1.1 Text to image

While current methods for text-to-image synthesis mainly take text inputs that describe the visual content of an image, more natural inputs often contain affective words such as happy, pleasing, scary, or frightful. To handle such inputs, it is necessary for models to consider the emotional effects during input text comprehension. Further, generating images that express or evoke a certain sentiment will require learning the mapping between visual content and emotional dimensions such as valence (i.e., positive or negative affectivity) and arousal (how calming or exciting the information is), as well as understanding how differently composing the same objects in an image can lead to different sentiments.

For particular application domains, input text descriptions may be more versatile. For instance, in medical image synthesis, a given input might be a clinical report containing one or several paragraphs of text description. Such domain-specific inputs also require prior knowledge for input text comprehension and text-to-image mapping. Other under-explored applications include taking paragraphs or multiple sentences as input to generate a sequence of images for story telling [148], or text-based video synthesis and editing [149–151].

For conditional synthesis, most current works perform one-to-many generation and try to improve the diversity of images generated from the same text input. One interesting work on text-to-image synthesis by Yin et al. proposes SD-GAN [53] which investigates the variability between different inputs intended for the same target image. New applications may be discovered that need methods for many-to-one synthesis using similar pipelines.

6.2 Images from other inputs

Existing methods using sketches and poses as user input treat them as rasterized images, and perform image-to-image translation as the synthesis method. As sketches and poses both contain geometric information and relationships between different points on the geometry are important, we believe it is worth investigating representing such inputs as sparse vectorized representations such as graphs, instead of rasterized representations. Using vectorized inputs will greatly reduce the size of the input and will also enable the use of existing graph understanding techniques such as graph neural networks. Using sketches as input, another interesting task is generating videos from sketch-based storyboards, with numerous applications in animation and visualization.

For graphic inputs that represent architectural structures such as layouts and wireframes, an important consideration is that the synthesized images should preserve structural constraints such as junctions, parallel lines, and planar surfaces [11] or relations between graphical elements [10]. In these scenarios, incorporating prior knowledge about the physical world could help enhance the photorealism of generated images and improve the structural coherence of generated designs.

It will also be interesting to further investigate image and video generation from other forms of input. Audio, for instance, is another intuitive, interactive, and expressive type of input. Generating photo-realistic video portraits that follow input audio streams [152–154] has many applications such as assisting the hearing impaired with speech comprehension, privacy-preserving video calls, and VR/AR for training professionals.

6.3 Linking paradigms

In conditional image synthesis, deep learning based methods have dominated, showing promising results. However, they still have limitations including the requirement for large training datasets and high computational cost for training. Since retrieval-and-composition methods are often light-weight and require little training, they can be complementary to deep learning based methods. Existing works on image synthesis from semantic maps have explored the strategy of combining retrieval-and-composition and learning-based models [107]. One approach to combination could be to use retrieval-and-composition to generate a draft image and then refine it to provide better visual quality and diversity using a learning-based approach.

Besides the quality of generated images, the controllability of the output and the interpretability of the model also play essential roles in synthesis. Although GAN models generally achieve better image quality than other methods, it is often more difficult to interact to control GAN methods than other learning based methods. Hybrid models combining GANs and VAEs [42, 43, 117, 155] have shown promising synthesis results as well as better feature disentanglement properties. Future works in image synthesis from intuitive user input can explore other hybrid models combining the advantages of GANs and VAEs, as in Ref. [117], as well as using normalizing flow based methods [20, 21] which allow both feature learning and tractable marginal likelihood estimation.

Overall, we believe cross pollination between major image generation paradigms will continue to be an important direction to produce new models that improve upon existing image synthesis paradigms by combining their merits and overcoming their limitations.

6.4 Evaluation of generation methods

6.4.1 Evaluation metrics

While a range of quantitative metrics for measuring the realism and diversity of generated images have been proposed, including the widely used IS [14], FID [16], and SSIM [17], they still lack consistency with human perception, which is why many works still rely on qualitative human assessment of the quality of images synthesized by different methods. Recently, some metrics, such as R-precision [25] and SOA score [65] in text-to-image synthesis, have been proposed to evaluate whether a generated image is well conditioned on the given input, in an attempt to achieve better consistency with human perception. Further work on automatic metrics that agree with human evaluation will continue to be important.

For a specific task or application, evaluation should be based on not just the final image quality but how well the generated images match the conditional input and serve the purpose of the intended application or task. If the synthesized images are used in down-stream tasks such as data augmentation for classification,

evaluation based on down-stream tasks also provides valuable information.

While it is difficult to compare methods across input types due to differences in input modality and interactivity, it is feasible to establish standard processes for synthesis from a particular kind of input, thus making fair benchmark comparison possible between methods given the same type of input.

6.4.2 Datasets

As shown in Section 5.3, large-scale datasets of natural images and annotations have been collected for specific object categories such as human bodies, faces, birds, and cars, and for scenes that contain multiple object categories such as those in COCO [46] and CityScapes [116]. In future, in order to enable applications in particular domains that benefit from image synthesis, e.g., medical image synthesis for data augmentation and movie video generation, domain-specific datasets with appropriate annotations will need to be created.

6.4.3 Evaluation of input choices

Existing image generation methods have been evaluated and compared mainly based on their output, i.e., the generated images. We believe that in image generation tasks conditioned on intuitive inputs, it is equally important to compare methods based on the choice of input. In Section 2.1, we introduced several characteristics that can be used to compare and evaluate inputs such as their accessibility, expressiveness, and interactivity. It will be interesting to study other important characteristics of inputs as well as criteria for evaluating how well an input type meets the needs of an application, how well the input supports interactive editing, how regularized the learned latent space is, and how well the synthesized image matches the input condition.

7 Conclusions

This review has covered main approaches for image synthesis and rendering given intuitive user inputs. First, we examined what makes a good paradigm for image synthesis from intuitive user input, from the perspective of user input characteristics and output image quality. We then provided an overview of the main generation paradigms: retrieval and composition, cGAN, cVAE, and hybrid models, autoregressive models, and normalizing flow based methods. Their relative strengths and weaknesses were discussed in the hope of inspiring ideas that draw connections between the main approaches, to produce models and methods that take advantage of the relative strengths of each paradigm. After the overview, we delved into details of specific algorithms for different input types and examined their ideas and contributions. In particular, we conducted a comprehensive survey of approaches for generating images from text, sketches, strokes, semantic label maps, poses, visual attributes, graphs, and layouts. Then, we summarized these existing methods in terms of benchmark datasets used and identified trends related to advances in model architecture design and training strategy, and strategies for handling specific input types. Last but not least, we provided our perspective on future directions related to input versatility, generation methodology, benchmark datasets, and method evaluation and comparison.

Acknowledgements

The co-authors Y.-C. Guo and S.-H. Zhang were supported by the National Natural Science Foundation of China (Project Nos. 61521002 and 61772298), a Research Grant of Beijing Higher Institution Engineering Research Center, and the Tsinghua–Tencent Joint Laboratory for Internet Innovation Technology.

References

[1] Zhang, H.; Xu, T.; Li, H. S.; Zhang, S. T.; Wang, X. G.; Huang, X. L.; Metaxas, D. StackGAN: Text to photo-realistic image synthesis with stacked generative adversarial networks. In: Proceedings of the IEEE International Conference on Computer Vision, 5908–5916, 2017.

[2] Qiao, T. T.; Zhang, J.; Xu, D. Q.; Tao, D. C. MirrorGAN: Learning text-to-image generation by redescription. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 1505–1514, 2019.

[3] Zhu, M. F.; Pan, P. B.; Chen, W.; Yang, Y. DM-GAN: Dynamic memory generative adversarial networks for text-to-image synthesis. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 5795–5803, 2019.

[4] Zhang, H.; Koh, J. Y.; Baldridge, J.; Lee, H.; Yang, Y. F. Cross-modal contrastive learning for text-to-image generation. *arXiv preprint* arXiv:2101.04702, 2021.

[5] Karras, T.; Laine, S.; Aila, T. M. A style-based generator architecture for generative adversarial networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 4396–4405, 2019.

[6] Sangkloy, P.; Lu, J. W.; Fang, C.; Yu, F.; Hays, J. Scribbler: Controlling deep image synthesis with sketch and color. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 6836–6845, 2017.

[7] Ghosh, A.; Zhang, R.; Dokania, P.; Wang, O.; Efros, A.; Torr, P.; Shechtman, E. Interactive sketch & fill: Multiclass sketch-to-image translation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, 1171–1180, 2019.

[8] Gao, C. Y.; Liu, Q.; Xu, Q.; Wang, L. M.; Liu, J. Z.; Zou, C. Q. SketchyCOCO: Image generation from freehand scene sketches. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 5173–5182, 2020.

[9] Liu, R.; Yu, Q.; Yu, S. Unsupervised sketch-to-photo synthesis. *arXiv preprint* arXiv:1909.08313, 2019.

[10] Li, J. N.; Yang, J. M.; Hertzmann, A.; Zhang, J. M.; Xu, T. F. LayoutGAN: Generating graphic layouts with wireframe discriminators *arXiv preprint* arXiv:1901.06767, 2019.

[11] Xue, Y.; Zhou, Z. H.; Huang, X. L. Neural wireframe renderer: Learning wireframe to image translations. In: *Computer Vision – ECCV 2020. Lecture Notes in Computer Science, Vol. 12371.* Vedaldi, A.; Bischof, H.; Brox, T.; Frahm, J. M. Eds. Springer Cham, 279–295, 2020.

[12] Wang, M.; Lyu, X. Q.; Li, Y. J.; Zhang, F. L. VR content creation and exploration with deep learning: A survey. *Computational Visual Media* Vol. 6, No. 1, 3–28, 2020.

[13] Kipf, T. N.; Welling, M. Semi-supervised classification with graph convolutional networks. *arXiv preprint* arXiv:1609.02907, 2016.

[14] Salimans, T.; Goodfellow, I.; Zaremba, W.; Cheung, V.; Radford, A.; Chen, X. Improved techniques for training GANs. In: Proceedings of the 29th International Conference on Neural Information Processing Systems, 2234–2242, 2016.

[15] Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2818–2826, 2016.

[16] Heusel, M.; Ramsauer, H.; Unterthiner, T.; Nessler, B.; Hochreiter, S. GANs trained by a two time-scale update rule converge to a local Nash equilibrium. In: Proceedings of the 31st International Conference on Neural Information Processing Systems, 6629–6640, 2017.

[17] Wang, Z.; Bovik, A. C.; Sheikh, H. R.; Simoncelli, E. P. Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing* Vol. 13, No. 4, 600–612, 2004.

[18] Wang, Z.; Simoncelli, E. P.; Bovik, A. C. Multiscale structural similarity for image quality assessment. In: Proceedings of the 37th Asilomar Conference on Signals, Systems & Computers, 1398–1402, 2003.

[19] Zhang, R.; Isola, P.; Efros, A. A.; Shechtman, E.; Wang, O. The unreasonable effectiveness of deep features as a perceptual metric. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 586–595, 2018.

[20] Rezende, D. J.; Mohamed, S. Variational inference with normalizing flows. In: Proceedings of the International Conference on Machine Learning, 1530–1538, 2015.

[21] Kingma, D. P.; Dhariwal, P. Glow: Generative flow with invertible 1×1 convolutions. In: Proceedings of the 31st International Conference on Neural Information Processing Systems, 10215–10224, 2018.

[22] Oord, A. V. D.; Kalchbrenner, N.; Kavukcuoglu, K. Pixel recurrent neural networks. In: Proceedings of the International Conference on Machine Learning, 1747–1756, 2016.

[23] Oord, A. V. D.; Kalchbrenner, N.; Espeholt, L.; Kavukcuoglu, K.; Vinyals, O.; Graves, A. Conditional image generation with pixelCNN decoders. In: Proceedings of the 29th International Conference on Neural Information Processing Systems, 4790–4798, 2016.

[24] Salimans, T.; Karpathy, A.; Chen, X.; Kingma, D. P. PixelCNN++: Improving the pixelCNN with discretized logistic mixture likelihood and other modifications. *arXiv preprint* arXiv:1701.05517, 2017.

[25] Xu, T.; Zhang, P. C.; Huang, Q. Y.; Zhang, H.; Gan, Z.; Huang, X. L.; He, X. D. AttnGAN: Fine-grained text to image generation with attentional generative adversarial networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 1316–1324, 2018.

[26] Lu, Y. Y.; Wu, S. Z.; Tai, Y. W.; Tang, C. K. Image generation from sketch constraint using contextual GAN. In: *Computer Vision – ECCV 2018. Lecture Notes in Computer Science, Vol. 11220.* Ferrari, V.; Hebert, M.; Sminchisescu, C.; Weiss, Y. Eds. Springer Cham, 213–228, 2018.

[27] Ma, L.; Jia, X.; Sun, Q.; Schiele, B.; Tuytelaars, T.; Van Gool, L. Pose guided person image generation. In: Proceedings of the 30th International Conference on Neural Information Processing Systems, 406–416, 2017.

[28] Ma, L. Q.; Sun, Q. R.; Georgoulis, S.; Van Gool, L.; Schiele, B.; Fritz, M. Disentangled person image generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 99–108, 2018.

[29] Siarohin, A.; Sangineto, E.; Lathuilière, S.; Sebe, N. Deformable GANs for pose-based human image generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 3408–3416, 2018.

[30] Song, S. J.; Zhang, W.; Liu, J. Y.; Mei, T. Unsupervised person image generation with semantic parsing transformation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2352–2361, 2019.

[31] Zhu, Z.; Huang, T. T.; Shi, B. G.; Yu, M.; Wang, B. F.; Bai, X. Progressive pose attention transfer for person image generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2342–2351, 2019.

[32] Belongie, S.; Malik, J.; Puzicha, J. Shape context: A new descriptor for shape matching and object recognition. In: Proceedings of the International Conference on Neural Information Processing Systems, 831–837, 2000.

[33] Chen, T.; Cheng, M. M.; Tan, P.; Shamir, A.; Hu, S. M. Sketch2Photo: Internet image montage. *ACM Transactions on Graphics* Vol. 28, No. 5, Article No. 124, 2009.

[34] Goodfellow, I. J.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio Y. Generative adversarial nets. In: Proceedings of the 27th International Conference on Neural Information Processing Systems, 2672–2680, 2014.

[35] Mirza, M.; Osindero, S. Conditional generative adversarial nets. *arXiv preprint* arXiv:1411.1784, 2014.

[36] Miyato, T.; Koyama, M. cGANs with projection discriminator. In: Proceedings of the International Conference on Learning Representations, 2018.

[37] Odena, A.; Olah, C.; Shlens, J. Conditional image synthesis with auxiliary classifier GANs. In: Proceedings of the International Conference on Machine Learning, 2642–2651, 2017.

[38] Kingma, D. P.; Welling, M. Auto-encoding variational bayes. *arXiv preprint* arXiv:1312.6114, 2013.

[39] Sohn, K.; Lee, H.; Yan, X. Learning structured output representation using deep conditional generative models. In: Proceedings of the 28th International Conference on Neural Information Processing Systems, 3483–3491, 2015.

[40] Klys, J.; Snell, J.; Zemel, R. Learning latent subspaces in variational autoencoders. In: Proceedings of the 31st International Conference on Neural Information Processing Systems, 6444–6454, 2018.

[41] Ivanov, O.; Figurnov, M.; Vetrov, D. Variational autoencoder with arbitrary conditioning. In: Proceedings of the International Conference on Learning Representations, 2018.

[42] Larsen, A. B. L.; Sønderby, S. K.; Larochelle, H.; Winther, O. Auto encoding beyond pixels using a learned similarity metric. In: Proceedings of the International Conference on Machine Learning, 1558–1566, 2016.

[43] Bao, J. M.; Chen, D.; Wen, F.; Li, H. Q.; Hua, G. C. VAE-GAN: Fine-grained image generation through asymmetric training. In: Proceedings of the IEEE International Conference on Computer Vision, 2764–2773, 2017.

[44] Nilsback, M. E.; Zisserman, A. Automated flower classification over a large number of classes. In: Proceedings of the 6th Indian Conference on Computer Vision, Graphics & Image Processing, 722–729, 2008.

[45] Welinder, P.; Branson, S.; Mita, T.; Wah, C.; Schrofi, F.; Belongie, S.; Perona, P. Caltech-UCSD Birds200. Technical Report CNS-TR-2010-001. California Institute of Technology, 2010.

[46] Lin, T. Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollfiar, P.; Zitnick, C. L. Microsoft COCO: Common objects in context. In: *Computer Vision–ECCV 2014. Lecture Notes in Computer Science, Vol. 8693*. Fleet, D.; Pajdla, T.; Schiele, B.; Tuytelaars, T. Eds. Springer Cham, 740–755, 2014.

[47] Reed, S.; Akata, Z.; Yan, X.; Logeswaran, L.; Schiele, B.; Lee, H. Generative adversarial text to image synthesis. In: Proceedings of the International Conference on Machine Learning, 1060–1069, 2016.

[48] Radford, A.; Metz, L.; Chintala, S. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint* arXiv:1511.06434, 2015.

[49] Mansimov, E.; Parisotto, E.; LeiBa, J.; Salakhutdinov, R. Generating images from captions with attention. *arXiv preprint* arXiv:1511.02793, 2015.

[50] Zhang, H.; Xu, T.; Li, H. S.; Zhang, S. T.; Wang, X. G.; Huang, X. L.; Metaxas, D. N. StackGAN++:

Realistic image synthesis with stacked generative adversarial networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 41, No. 8, 1947–1962, 2019.

[51] Zhang, Z. Z.; Xie, Y. P.; Yang, L. Photographic text-to-image synthesis with a hierarchically-nested adversarial network. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 6199–6208, 2018.

[52] Isola, P.; Zhu, J. Y.; Zhou, T. H.; Efros, A. A. Image-to-image translation with conditional adversarial networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 5967–5976, 2017.

[53] Yin, G. J.; Liu, B.; Sheng, L.; Yu, N. H.; Wang, X. G.; Shao, J. Semantics disentangling for text-to-image generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2322–2331, 2019.

[54] Reed, S. E.; Akata, Z.; Mohan, S.; Tenka, S.; Schiele, B.; Lee, H. Learning what and where to draw. In: Proceedings of the 29th International Conference on Neural Information Processing Systems, 217–225, 2016.

[55] Hong, S.; Yang, D. D.; Choi, J.; Lee, H. Inferring semantic layout for hierarchical text-to-image synthesis. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 7986–7994, 2018.

[56] Chen, Q. F.; Koltun, V. Photographic image synthesis with cascaded refinement networks. In: Proceedings of the IEEE International Conference on Computer Vision, 1520–1529, 2017.

[57] Li, W. B.; Zhang, P. C.; Zhang, L.; Huang, Q. Y.; He, X. D.; Lyu, S. W.; Gao, J. F. Object-driven text-to-image synthesis via adversarial training. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 12166–12174, 2019.

[58] Girshick, R. Fast R-CNN. In: Proceedings of the IEEE International Conference on Computer Vision, 1440–1448, 2015.

[59] Johnson, J.; Gupta, A.; Li, F. F. Image generation from scene graphs. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 1219–1228, 2018.

[60] Krishna, R.; Zhu, Y. K.; Groth, O.; Johnson, J.; Hata, K. J.; Kravitz, J.; Chen, S.; KAlantidis, Y.; Li, L.-J.; Shamma, D. A. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision* Vol. 123, No. 1, 32–73, 2017.

[61] Caesar, H.; Uijlings, J.; Ferrari, V. COCO-stuff: Thing and stuff classes in context. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 1209–1218, 2018.

[62] Hinz, T.; Heinrich, S.; Wermter, S. Generating multiple objects at spatially distinct locations. *arXiv preprint* arXiv:1901.00686, 2019.

[63] Tan, F. W.; Feng, S.; Ordonez, V. Text2Scene: Generating compositional scenes from textual descriptions. *arXiv preprint* arXiv:1809.01110, 2018.

[64] Bodla, N.; Hua, G.; Chellappa, R. Semi-supervised FusedGAN for conditional image generation. In: *Computer Vision–ECCV 2018. Lecture Notes in Computer Science, Vol. 11209*. Ferrari, V.; Hebert, M.; Sminchisescu, C.; Weiss, Y. Eds. Springer Cham, 689–704, 2018.

[65] Hinz, T.; Heinrich, S.; Wermter, S. Semantic object accuracy for generative text-to-image synthesis. *arXiv preprint* arXiv:1910.13321, 2019.

[66] Wang, T. C.; Liu, M. Y.; Zhu, J. Y.; Tao, A.; Kautz, J.; Catanzaro, B. High-resolution image synthesis and semantic manipulation with conditional GANs. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 8798–8807, 2018.

[67] Eitz, M.; Richter, R.; Hildebrand, K.; Boubekeur, T.; Alexa, M. Photosketcher: Interactive sketch-based image synthesis. *IEEE Computer Graphics and Applications* Vol. 31, No. 6, 56–66, 2011.

[68] Hu, S.-M.; Zhang, F.-L.; Wang, M.; Martin, R. R.; Wang, J. PatchNet: A patch-based image representation for interactive library-driven image editing. *ACM Transactions on Graphics* Vol. 32, No. 6, Article No. 196, 2013.

[69] Wang, J. Y.; Zhao, Y.; Qi, Q.; Huo, Q. M.; Zou, J.; Ge, C.; Liao, J. MindCamera: Interactive sketch-based image retrieval and synthesis. *IEEE Access* Vol. 6, 3765–3773, 2018.

[70] Turmukhambetov, D.; Campbell, N. D. F.; Goldman, D. B.; Kautz, J. Interactive sketch-driven image synthesis. *Computer Graphics Forum* Vol. 34, No. 8, 130–142, 2015.

[71] Xie, S. N.; Tu, Z. W. Holistically-nested edge detection. In: Proceedings of the IEEE International Conference on Computer Vision, 1395–1403, 2015.

[72] Winnemöller, H.; Kyprianidis, J. E.; Olsen, S. C. XDoG: An eXtended difference-of-Gaussians compendium including advanced image stylization. *Computers & Graphics* Vol. 36, No. 6, 740–753, 2012.

[73] Kang, H.; Lee, S.; Chui, C. K. Coherent line drawing. In: Proceedings of the 5th International Symposium on Non-photorealistic Animation and Rendering, 43–50, 2007.

[74] Li, Y. J.; Fang, C.; Hertzmann, A.; Shechtman, E.; Yang, M. H. Im2Pencil: Controllable pencil illustration from photographs. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 1525–1534, 2019.

[75] Li, M. T.; Lin, Z.; Mech, R.; Yumer, E.; Ramanan, D. Photo-sketching: Inferring contour drawings from images. In: Proceedings of the IEEE Winter Conference on Applications of Computer Vision, 1403–1412, 2019.

[76] Gastal, E. S. L.; Oliveira, M. M. Domain transform for edge-aware image and video processing. *ACM Transactions on Graphics* Vol. 30, No. 4, Article No. 69, 2011.

[77] Hahn-Powell, G. V.; Archangeli, D. AutoTrace: An automatic system for tracing tongue contours. *The Journal of the Acoustical Society of America* Vol. 136, No. 4, 2104, 2014.

[78] Simo-Serra, E.; Iizuka, S.; Sasaki, K.; Ishikawa, H. Learning to simplify. *ACM Transactions on Graphics* Vol. 35, No. 4, Article No. 121, 2016.

[79] Chen, W. L.; Hays, J. SketchyGAN: Towards diverse and realistic sketch to image synthesis. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 9416–9425, 2018.

[80] Li, Y. H.; Chen, X. J.; Wu, F.; Zha, Z. J. LinesToFacePhoto: Face photo generation from lines with conditional self-attention generative adversarial networks. In: Proceedings of the 27th ACM International Conference on Multimedia, 2323–2331, 2019.

[81] Güçlütürk, Y.; Güçlü, U.; van Lier, R.; van Gerven, M. A. J. Convolutional sketch inversion. In: *Computer Vision–ECCV 2016 Workshops. Lecture Notes in Computer Science, Vol. 9913.* Hua, G.; Jégou, H. Eds. Springer Cham, 810–824, 2016.

[82] Mescheder, L.; Geiger, A.; Nowozin, S. Which training methods for GANs do actually converge? *arXiv preprint* arXiv:1801.04406, 2018.

[83] Huang, X.; Liu, M. Y.; Belongie, S.; Kautz, J. Multimodal unsupervised image-to-image translation. In: *Computer Vision–ECCV 2018. Lecture Notes in Computer Science, Vol. 11207.* Ferrari, V.; Hebert, M.; Sminchisescu, C.; Weiss, Y. Eds. Springer Cham, 179–196, 2018.

[84] Portenier, T.; Hu, Q.; Szabó, A.; Bigdeli, S. A.; Favaro, P.; Zwicker, M. Faceshop: Deep sketch-based face image editing. *arXiv preprint* arXiv:1804.08972, 2018.

[85] Xia, W.; Yang, Y.; Xue, J.-H. Calisketch: Stroke calibration and completion for high quality face image generation from poorly-drawn sketches. *arXiv preprint* arXiv:1911.00426, 2019.

[86] Chen, S.-Y.; Su, W.; Gao, L.; Xia, S.; Fu, H. DeepFaceDrawing: Deep generation of face images from sketches. *ACM Transactions on Graphics* Vol. 39, No. 4, Article No. 72, 2020.

[87] Sangkloy, P.; Burnell, N.; Ham, C.; Hays, J. The sketchy database. *ACM Transactions on Graphics* Vol. 35, No. 4, Article No. 119, 2016.

[88] Eitz, M.; Hays, J.; Alexa, M. How do humans sketch objects? *ACM Transactions on Graphics* Vol. 31, No. 4, Article No. 44, 2012.

[89] Caesar, H.; Uijlings, J.; Ferrari, V. COCO-stuff: Thing and stuff classes in context. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 1209–1218, 2018.

[90] Huang, X.; Belongie, S. Arbitrary style transfer in real-time with adaptive instance normalization. In: Proceedings of the IEEE International Conference on Computer Vision, 1510–1519, 2017.

[91] Zhu, P. H.; Abdal, R.; Qin, Y. P.; Wonka, P. SEAN: Image synthesis with semantic region-adaptive normalization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 5103–5112, 2020.

[92] Yu, Q.; Liu, F.; Song, Y. Z.; Xiang, T.; Hospedales, T. M.; Loy, C. C. Sketch me that shoe. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 799–807, 2016.

[93] Krause, J.; Stark, M.; Jia, D.; Li, F. F. 3D object representations for fine-grained categorization. In: Proceedings of the IEEE International Conference on Computer Vision Workshops, 554–561, 2013.

[94] Yu, A.; Grauman, K. Fine-grained visual comparisons with local learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 192–199, 2014.

[95] Yu, A.; Grauman, K. Semantic jitter: Dense supervision for visual comparisons via synthetic images. In: Proceedings of the IEEE International Conference on Computer Vision, 5571–5580, 2017.

[96] Liu, Z. W.; Luo, P.; Wang, X. G.; Tang, X. O. Deep learning face attributes in the wild. In: Proceedings of the IEEE International Conference on Computer Vision, 3730–3738, 2015.

[97] Karras, T.; Aila, T.; Laine, S.; Lehtinen, J. Progressive growing of GANs for improved quality, stability, and variation. *arXiv preprint* arXiv:1710.10196, 2017.

[98] Wang, X. G.; Tang, X. O. Face photo-sketch synthesis and recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 31, No. 11, 1955–1967, 2009.

[99] Johnson, M.; Brostow, G. J.; Shotton, J.; Arandjelovic, O.; Kwatra, V.; Cipolla, R. Semantic photo synthesis. *Computer Graphics Forum* Vol. 25, No. 3, 407–413, 2006.

[100] Bansal, A.; Sheikh, Y.; Ramanan, D. Shapes and context: In-the-wild image synthesis & manipulation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2312–2321, 2019.

[101] Chen, Q. F.; Koltun, V. Photographic image synthesis with cascaded refinement networks. In: Proceedings of the IEEE International Conference on Computer Vision, 1520–1529, 2017.

[102] Lassner, C.; Pons-Moll, G.; Gehler, P. V. A generative model of people in clothing. In: Proceedings of the IEEE International Conference on Computer Vision, 853–862, 2017.

[103] Park, T.; Liu, M. Y.; Wang, T. C.; Zhu, J. Y. Semantic image synthesis with spatially-adaptive normalization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2332–2341, 2019.

[104] Liu, X.; Yin, G.; Shao, J.; Wang, X.; Li, H. Learning to predict layout-to-image conditional convolutions for semantic image synthesis. In: Proceedings of the 33rd Conference on Neural Information Processing Systems, 570–580, 2019.

[105] Zhu, Z.; Xu, Z. L.; You, A. S.; Bai, X. Semantically multi-modal image synthesis. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 5466–5475, 2020.

[106] Tang, H.; Xu, D.; Yan, Y.; Torr, P. H. S.; Sebe, N. Local class-specific and global image-level generative adversarial networks for semantic-guided scene generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 7867–7876, 2020.

[107] Qi, X. J.; Chen, Q. F.; Jia, J. Y.; Koltun, V. Semi-parametric image synthesis. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 8808–8816, 2018.

[108] Wang, M.; Yang, G. Y.; Li, R. L.; Liang, R. Z.; Zhang, S. H.; Hall, P. M.; Hu, S.-M. Example-guided style-consistent image synthesis from semantic labeling. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 1495–1504, 2019.

[109] Liang, X. D.; Liu, S.; Shen, X. H.; Yang, J. C.; Liu, L. Q.; Dong, J.; Lin, L.; Yan, S. C. Deep human parsing with active template regression. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 37, No. 12, 2402–2414, 2015.

[110] Liang, X. D.; Xu, C. Y.; Shen, X. H.; Yang, J. C.; Liu, S.; Tang, J. H.; Lin, L.; Yan, S. C. Human parsing with contextualized convolutional neural network. In: Proceedings of the IEEE International Conference on Computer Vision, 1386–1394, 2015.

[111] Liu, Z. W.; Luo, P.; Qiu, S.; Wang, X. G.; Tang, X. O. DeepFashion: Powering robust clothes recognition and retrieval with rich annotations. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1096–1104, 2016.

[112] Lee, C. H.; Liu, Z. W.; Wu, L. Y.; Luo, P. MaskGAN: Towards diverse and interactive facial image manipulation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 5548–5557, 2020.

[113] Zhou, B. L.; Zhao, H.; Puig, X.; Fidler, S.; Barriuso, A.; Torralba, A. Semantic understanding of scenes through the ADE20K dataset. *arXiv preprint* arXiv:1608.05442, 2016.

[114] Zhou, B. L.; Zhao, H.; Puig, X.; Fidler, S.; Barriuso, A.; Torralba, A. Scene parsing through ADE20K dataset. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 5122–5130, 2017.

[115] Silberman, N.; Hoiem, D.; Kohli, P.; Fergus, R. Indoor segmentation and support inference from RGBD images. In: *Computer Vision – ECCV 2012. Lecture Notes in Computer Science, Vol. 7576*. Fitzgibbon, A.; Lazebnik, S.; Perona, P.; Sato, Y.; Schmid, C. Eds. Springer Berlin Heidelberg, 746–760, 2012.

[116] Cordts, M.; Omran, M.; Ramos, S.; Rehfeld, T.; Enzweiler, M.; Benenson, R.; Franke, U.; Roth, S.; Schiele B. The cityscapes dataset for semantic urban scene understanding. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 3213–3223, 2016.

[117] Bem, R. D.; Ghosh, A.; Boukhayma, A.; Ajanthan, T.; Siddharth, N.; Torr, P. A conditional deep generative model of people in natural images. In: Proceedings of the IEEE Winter Conference on Applications of Computer Vision, 1449–1458, 2019.

[118] Chen, L. C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A. L. DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 40, No. 4, 834–848, 2018.

[119] Chen, L. C.; Zhu, Y. K.; Papandreou, G.; Schroff, F.; Adam, H. Encoder-decoder with atrous separable convolution for semantic image segmentation. In: *Computer Vision – ECCV 2018. Lecture Notes in Computer Science, Vol. 11211*. Ferrari, V.; Hebert, M.; Sminchisescu, C.; Weiss, Y. Eds. Springer Cham, 833–851, 2018.

[120] Balakrishnan, G.; Zhao, A.; Dalca, A. V.; Durand, F.; Guttag, J. Synthesizing images of humans in unseen poses. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 8340–8348, 2018.

[121] Pumarola, A.; Agudo, A.; Sanfeliu, A.; Moreno-Noguer, F. Unsupervised person image synthesis in arbitrary poses. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 8620–8628, 2018.

[122] Dong, H.; Liang, X.; Gong, K.; Lai, H.; Zhu, J.; Yin, J. Soft-gated warping-GAN for pose-guided person image synthesis. In: Proceedings of the 32nd Conference on Neural Information Processing Systems, 474–484, 2018.

[123] Li, Y. N.; Huang, C.; Loy, C. C. Dense intrinsic appearance flow for human pose transfer. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 3688–3697, 2019.

[124] Zheng, L.; Shen, L. Y.; Tian, L.; Wang, S. J.; Wang, J. D.; Tian, Q. Scalable person re-identification: A benchmark. In: Proceedings of the IEEE International Conference on Computer Vision, 1116–1124, 2015.

[125] Yan, X. C.; Yang, J. M.; Sohn, K.; Lee, H. Attribute2Image: Conditional image generation from visual attributes. In: *Computer Vision – ECCV 2016. Lecture Notes in Computer Science, Vol. 9908*. Leibe, B.; Matas, J.; Sebe, N.; Welling, M. Eds. Springer Cham, 776–791, 2016.

[126] Huang, G. B.; Ramesh, M.; Berg, T.; Learned-Miller, E. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49. University of Massachusetts, 2007.

[127] He, Z. L.; Zuo, W. M.; Kan, M. N.; Shan, S. G.; Chen, X. L. AttGAN: Facial attribute editing by only changing what you want. *IEEE Transactions on Image Processing* Vol. 28, No. 11, 5464–5478, 2019.

[128] Zhang, G.; Kan, M. N.; Shan, S. G.; Chen, X. L. Generative adversarial network with spatial attention for face attribute editing. In: *Computer Vision – ECCV 2018. Lecture Notes in Computer Science, Vol. 11210*. Ferrari, V.; Hebert, M.; Sminchisescu, C.; Weiss, Y. Eds. Springer Cham, 422–437, 2018.

[129] Qian, S. J.; Lin, K. Y.; Wu, W.; Liu, Y.; Wang, Q.; Shen, F. M.; Qian, C.; He, R. Make a face: Towards arbitrary high fidelity face manipulation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, 10032–10041, 2019.

[130] Men, Y. F.; Mao, Y. M.; Jiang, Y. N.; Ma, W. Y.; Lian, Z. H. Controllable person image synthesis with attribute-decomposed GAN. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 5083–5092, 2020.

[131] Lee, H.; Lee, S. G. Fashion attributes-to-image synthesis using attention-based generative adversarial network. In: Proceedings of the IEEE Winter Conference on Applications of Computer Vision, 462–470, 2019.

[132] Dinh, L.; Sohl-Dickstein, J.; Bengio, S. Density estimation using real NVP. *arXiv preprint* arXiv: 1605.08803, 2016.

[133] Zhao, B.; Meng, L. L.; Yin, W. D.; Sigal, L. Image generation from layout. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 8576–8585, 2019.

[134] Luo, A.; Zhang, Z. T.; Wu, J. J.; Tenenbaum, J. B. End-to-end optimization of scene layout. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 3753–3762, 2020.

[135] Song, S. R.; Yu, F.; Zeng, A.; Chang, A. X.; Savva, M.; Funkhouser, T. Semantic scene completion from a single depth image. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 190–198, 2017.

[136] Choi, Y.; Choi, M.; Kim, M.; Ha, J. W.; Kim, S.; Choo, J. StarGAN: Unified generative adversarial networks for multi-domain image-to-image translation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 8789–8797, 2018.

[137] Vahdat, A.; Kautz, J. NVAE: A deep hierarchical variational autoencoder. In: Proceedings of the 34th Conference on Neural Information Processing Systems, 2020.

[138] Zhang, H.; Goodfellow, I. J.; Metaxas, D.; Odena, A. Self-attention generative adversarial networks. In: Proceedings of the International Conference on Machine Learning, 7354–7363, 2019.

[139] De Vries, H.; Strub, F.; Mary, J.; Larochelle, H.; Pietquin, O.; Courville A. Modulating early visual processing by language. In: Proceedings of the 30th Conference on Neural Information Processing Systems 6594–6604, 2017.

[140] Miyato, T.; Kataoka, T.; Koyama, M.; Yoshida, Y. Spectral normalization for generative adversarial networks. In: Proceedings of the International Conference on Learning Representations, 2018.

[141] Arjovsky, M.; Chintala, S.; Bottou, L. Wasserstein generative adversarial networks. In: Proceedings of the 34th International Conference on Machine Learning, Vol. 70, 214–223, 2017.

[142] Gulrajani, I.; Ahmed, F.; Arjovsky, M.; Dumoulin, V.; Courville, A. C. Improved training of Wasserstein GANs. In: Proceedings of the 30th Conference on Neural Information Processing Systems, 5767–5777, 2017.

[143] Mao, X. D.; Li, Q.; Xie, H. R.; Lau, R. Y. K.; Wang, Z.; Smolley, S. P. Least squares generative adversarial networks. In: Proceedings of the IEEE International Conference on Computer Vision, 2813–2821, 2017.

[144] Lim, J. H.; Ye, J. C. Geometric GAN. *arXiv preprint* arXiv:1705.02894, 2017.

[145] Johnson, J.; Alahi, A.; Li, F. F. Perceptual losses for real-time style transfer and super-resolution. In: *Computer Vision – ECCV 2016. Lecture Notes in Computer Science, Vol. 9906.* Leibem, B.; Matas, J.; Sebe, N.; Welling, M. Eds. Springer Cham, 694–711, 2016.

[146] Velifickovific, P.; Cucurull, G.; Casanova, A.; Romero, A.; Lifio, P.; Bengio, Y. Graph attention networks. In: Proceedings of the International Conference on Learning Representations, 2018.

[147] Ionescu, C.; Papava, D.; Olaru, V.; Sminchisescu, C. Human3.6M: Large scale datasets and predictive methods for 3D human sensing in natural environments. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 36, No. 7, 1325–1339, 2014.

[148] Li, Y. T.; Gan, Z.; Shen, Y. L.; Liu, J. J.; Cheng, Y.; Wu, Y. X.; Carin, L.; Carlson, D.; Gao, J. F. StoryGAN: A sequential conditional GAN for story visualization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 6322–6331, 2019.

[149] Pan, Y. W.; Qiu, Z. F.; Yao, T.; Li, H. Q.; Mei, T. To create what you tell: Generating videos from captions. In: Proceedings of the 25th ACM international Conference on Multimedia, 1789–1798, 2017.

[150] Li, Y.; Min, M. R.; Shen, D.; Carlson, D.; Carin, L. Video generation from text. In: Proceedings of the AAAI Conference on Artificial Intelligence, 2018.

[151] Wang, M.; Yang, G.-W.; Hu, S.-M.; Yau, S.-T.; Shamir, A. Write-a-video: Computational video montage from themed text. *ACM Transactions on Graphics* Vol. 38, No. 6, Article No. 177, 2019.

[152] Chen, L. L.; Maddox, R. K.; Duan, Z. Y.; Xu, C. L. Hierarchical cross-modal talking face generation with dynamic pixel-wise loss. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 7824–7833, 2019.

[153] Zhou, H.; Liu, Y.; Liu, Z. W.; Luo, P.; Wang, X. G. Talking face generation by adversarially disentangled audio-visual representation. *Proceedings of the AAAI Conference on Artificial Intelligence* Vol. 33, 9299–9306, 2019.

[154] Wen, X.; Wang, M.; Richardt, C.; Chen, Z. Y.; Hu, S. M. Photorealistic audio-driven video portraits. *IEEE Transactions on Visualization and Computer Graphics* Vol. 26, No. 12, 3457–3466, 2020.

[155] Mescheder, L.; Nowozin, S.; Geiger, A. Adversarial variational bayes: Unifying variational autoencoders and generative adversarial networks. In: Proceedings of the 34th International Conference on Machine Learning, 2391–2400, 2017.

Yuan Xue received his bachelor degree in electrical engineering from Huazhong University of Science and Technology, Wuhan, China, in 2015, and his master degree in computer science from Lehigh University, Bethlehem, USA. He is now working towards his Ph.D. degree in the College of Information Sciences and Technology at the Pennsylvania State University, USA. His current research interests include computer vision, machine learning, and biomedical image analysis.

Yuan-Chen Guo received his bachelor degree from Tsinghua University in 2019, where he is currently pursuing his Ph.D. degree in the Department of Computer Science and Technology. His research interests include computer graphics and computer vision.

Han Zhang is currently a research scientist in Google Brain, USA. He received his Ph.D. degree in computer science from Rutgers University, USA in 2018. His research interests include generative modeling, semi-supervised learning, and vision-language interaction.

Tao Xu is currently a research scientist in Facebook, USA. She received her Ph.D. degree in computer Science from Lehigh University in 2018, her M.S. degree in computer science from the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China, in 2013, and her B.E. degree from China Agricultural University, Beijing, China, in 2010. Her research interests include deep learning and computer vision.

Song-Hai Zhang received his Ph.D. degree in computer science and technology from Tsinghua University, in 2007. He is currently an associate professor in the Department of Computer Science and Technology at Tsinghua University. His research interests include image and video analysis and processing as well as computer graphics.

Xiaolei Huang is an associate professor in the College of Information Sciences and Technology at the Pennsylvania State University. Her research interests lie at the intersection of computer vision, machine learning, and biomedical image analysis, focusing on methods for image segmentation, image synthesis, object recognition, computer-assisted diagnosis, among others. She has over 150 publications and 7 patents in these areas. She is an associate editor for the *Computer Vision and Image Understanding* journal. She received her bachelor degree in computer science from Tsinghua University, and her master and doctoral degrees in computer science from Rutgers University.