Wang YJ, Chen XL, Chen BQ. SinGRAV: Learning a generative radiance volume from a single natural scene. JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY 39(2): 305-319 Mar. 2024. DOI: 10.1007/s11390-023-3596-9

SinGRAV: Learning a Generative Radiance Volume from a Single Natural Scene

Yu-Jie Wang[1, 2, 3] (王玉洁), Xue-Lin Chen[4] (陈学霖), and Bao-Quan Chen[2, 3, *] (陈宝权), *Fellow, CCF, IEEE*

[1] *School of Computer Science and Technology, Shandong University, Qingdao 266237, China*

[2] *State Key Laboratory of General Artificial Intelligence, Beijing 100871, China*

[3] *School of Intelligence Science and Technology, Peking University, Beijing 100871, China*

[4] *Tencent AI Lab, Tencent Holdings Limited, Shenzhen 518057, China*

E-mail: yujiew@mail.sdu.edu.cn; xuelinchen@tencent.com; baoquan@pku.edu.cn

Received July 14, 2023; accepted January 12, 2024.

Abstract We present SinGRAV, an attempt to learn a generative radiance volume from multi-view observations of a single natural scene, in stark contrast to existing category-level 3D generative models that learn from images of many object-centric scenes. Inspired by SinGAN, we also learn the internal distribution of the input scene, which necessitates our key designs w.r.t. the scene representation and network architecture. Unlike popular multi-layer perceptrons (MLP)-based architectures, we particularly employ convolutional generators and discriminators, which inherently possess spatial locality bias, to operate over voxelized volumes for learning the internal distribution over a plethora of overlapping regions. On the other hand, localizing the adversarial generators and discriminators over confined areas with limited receptive fields easily leads to highly implausible geometric structures in the spatial. Our remedy is to use spatial inductive bias and joint discrimination on geometric clues in the form of 2D depth maps. This strategy is effective in improving spatial arrangement while incurring negligible additional computational cost. Experimental results demonstrate the ability of SinGRAV in generating plausible and diverse variations from a single scene, the merits of SinGRAV over state-of-the-art generative neural scene models, and the versatility of SinGRAV by its use in a variety of applications. Code and data will be released to facilitate further research.

Keywords generative model, neural radiance field, 3D scene generation

1 Introduction

3D generative modeling has made great strides via gravitating towards neural scene representations, which boast unprecedented photo-realism. Generative models[1–5] can now draw class-specific scenes (e.g., cars and portraits), offering a glimpse into the boundless universe in the virtual. Yet an obvious question is how we can go beyond class-specific scenes, and replicate the success with general natural scenes①, creating at scale diverse scenes of more sorts. This work presents an attempt towards answering this question.

Another key that boosted the field is differentiable projection techniques that enable training on only 2D images, bypassing the explicit need for collecting 3D models. However, collecting tons of homogeneous images for each scene type ad hoc is cumbersome, and would become prohibitive when the scene type varies dynamically. Herein, our key observation is that general natural scenes often contain many sim-

Regular Paper

Special Section of CGI 2023

This work was (partially) supported by the International (Regional) Cooperation and Exchange Program of National Natural Science Foundation of China under Grant No. 62161146002, and the Shenzhen Collaborative Innovation Program under Grant No. CJGJZD2021048092601003.

*Corresponding Author

①Analogous to the concept in single image generation[6], a general natural scene contains sufficiently rich information, such as complex structures and textures, for learning an internal distribution.

ilar constituents whose geometry, appearance, and spatial arrangements follow some clear patterns, while exhibiting rich variations over different regions. Therefore we propose to train on a single general natural scene, which builds upon recent success with differentiable rendering, particularly to learn the 3D internal distribution from its observation images.

To this end, we present SinGRAV, for a generative radiance volume learned to synthesize variations from images of a single general natural scene. Training with a single scene necessitates learning the internal statistics, which triggers a design choice of the scene representation with locality modeling in SinGRAV. Besides, different from object-centric scenes as in [1, 3, 4] or image generation[6, 7], plausible geometric arrangements are vital to 3D scene generation. Therefore, key designs are dedicated to improving the spatial arrangement plausibility, without significantly increasing the computational overhead.

Specifically, the core of SinGRAV is to learn from local regions via localizing the training. As our supervision comes from purely 2D observations, learning internal distributions consequently grounds SinGRAV on the assumption that multi-view observations share a consistent internal distribution for learning. This can be simply realized by capturing images with cameras at roughly uniform distances to the scene, e.g., with drones for outdoor scenes. On the other hand, multi-layer perceptron (MLP)-based representations tend to synthesize holistically and perform better at modeling global patterns over local ones[8], as also revealed in our ablation studies. Hence, we resort to convolutional operations, which generate discrete radiance volumes from noise volumes with limited receptive fields, for learning local properties over a confined spatial extent, granting better out-of-distribution generation in terms of global configuration. Moreover, we adopt a multi-scale architecture containing a pyramid of convolutional GANs to capture the internal distribution at various scales, alleviating the notorious mode-collapse issue. This is similar in spirit to [6]; however, important designs must be incorporated to efficiently and effectively improve the plausibility of the spatial arrangement of the generated 3D scene. Specifically, we found that coarser scales produce highly implausible geometric structures, which cannot be easily distinguished by discriminators operating on the renderings with limited receptive fields. Our remedy is to use a combination of 1) the spatial inductive bias injected at the coarsest scale, and 2) the joint discrimination on the geometric depth map, which is a byproduct from reconstructing the input scene and also the volume rendering technique.

To validate the proposed framework, we collect a dataset containing various example scenes, and conduct comprehensive investigations. We demonstrate that SinGRAV enables us to easily generate plausible variations of an input scene in large quantities and varieties, which is exemplified by a subset of the results showcased in Fig.1. To evaluate the plausibility of generated scenes, we compare the observed multi-view images from the generated scenes against those from the given exemplar scene. And performance comparisons are made to state-of-the-art generative neural scene models. We also extensively investigate each key design choice for inspiring more future research. Finally, we show the versatility of SinGRAV through its use in a series of applications, spanning 3D scene editing, composition, and animation.

2 Related Work

Neural Scene Representation and Rendering. In recent years, neural scene representations have been the de facto infrastructure in several tasks, including representing shapes[9–13], novel view synthesis[14–16], and 3D generative modeling[1–5, 17, 18]. Paired with differentiable projection functions, the geometry and appearance of the underlying scene can be optimized based on the error derived from the downstream supervision signals. [9–12, 19] adopt neural implicit fields to represent 3D shapes and attain highly detailed geometries. On the other hand, [16, 20, 21] work on discrete grids, UV maps, and point clouds, respectively, with attached learnable neural features that can produce pleasing novel view imagery. More recently, the Neural Radiance Field (NeRF) technique[15] has revolutionized several research fields with a trained MLP-based radiance and opacity field, achieving unprecedented success in producing photo-realistic imagery. An explosion of NeRF techniques occurred in the research community since then that improves the NeRF in various aspects of the problem[22–30]. A research direction drawing increasing interest, which we discuss in the following, is to incorporate such neural representations to learn a 3D generative model possessing photo-realistic viewing effects.

Generative Neural Scene Generation. Recently,

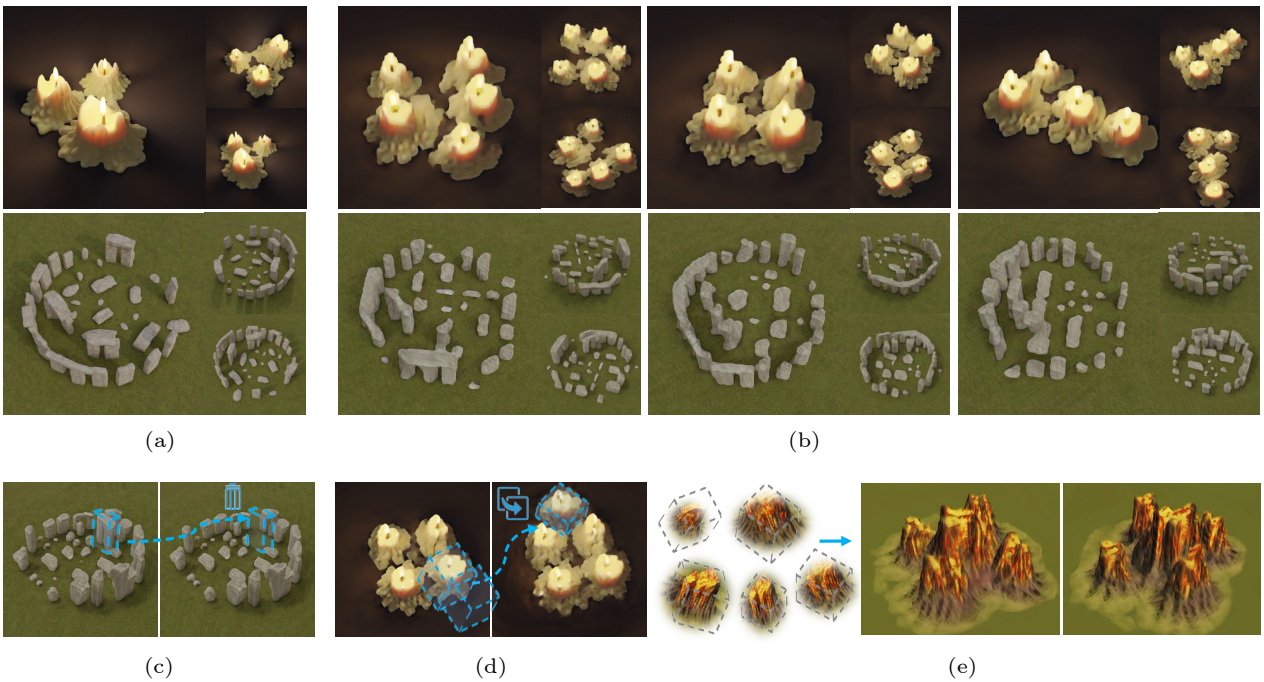


Fig.1. Scene generation results and application results. (a) Three views of the training scenes. (b) Randomly generated scenes from the proposed framework SinGRAV. (c) Results for object removal. (d) Results for duplicating an object. (e) Results for scene composition. Note how the global and object configurations vary in generated scenes shown in (b) yet still resembles the original training scene.

3D-aware generative models have attracted much attention and achieved appealing results. The heart of these models is a 3D neural scene representation, paired with the differentiable volume rendering, enabling the supervision imposed in the image domain. [1, 4] integrate a neural radiance field into generative models, and directly produce the final images via volume rendering, yielding consistent multi-view renderings of generated scenes. To overcome the low query efficiency and high memory cost issues, [2, 3, 5] propose to adopt 2D neural renderers to achieve high-resolution renderings. More often than not, these methods are demonstrated on single-object scenes. [18] utilizes a grid of locally conditioned radiance sub-fields to model indoor scenes. All these studies focus on category-specific models, requiring training on sufficient volumes of image data collected from many homogeneous scenes. In this work, we target general natural scenes, which in general possess intricate and exclusive characteristics, suggesting difficulties in collecting necessary volumes of training data and rendering these data-consuming learning setups intractable. Moreover, as aforementioned, our task necessitates localizing the training over local regions, which is lacking in MLP-based representations, leading us to use voxel grids.

Concurrently, [31] also explores the learning of a 3D generative model from single-scene images, employing a two-stage framework. The method[31] first constructs a 3D volume representation from the input multi-view images and subsequently employs a 3D discriminator to enhance spatial plausibility in the generated scenes. In contrast, we adopt a one-stage training approach and our proposed framework is significantly different from [31] in its strategy for improving 3D spatial plausibility. Specifically, we enhance the generator by incorporating Cartesian Spatial Grid positional encoding[32], and we make the 2D discriminator jointly discriminate the depth rendered from generated scenes to guarantee spatial plausibility. In comparison to the approach presented in [31], which includes an additional 3D volume discriminator, the strategies devised in our proposed framework are more computation-efficient and memory-friendly. Besides, this work provides an extensive analysis of the influence of different scene representations on the task. Additionally, we showcase the versatility of our core framework in various 3D modeling applications, including scene editing, composition, and animation. Very recently, another concurrent work[33] has been proposed with a similar goal, which focuses mainly on indoor scenes.

Generative Image Models. Since the introduction of Generative Adversarial Networks (GANs)[34], state-

of-the-art studies can now synthesize high-fidelity images[35–39]. More recently, diffusion models[40] have also shown dominance in this field. Despite the impressive success, most of them require a large set (typically dozens of thousands) of category-specific images to learn the data distribution. Therefore, a line of studies occurred to train a generative model on a single image[6, 7], and achieved compelling results with learned internal distributions. Particularly, SinGRAV is inspired by [6], but has to tackle unique challenges arising from learning the internal distribution from multi-view observations of a 3D scene. We elaborately choose the neural scene presentation with locality modeling, and propose an effective strategy to cope with the challenges of producing implausible spatial arrangements. Eventually, SinGRAV can generate diverse 3D scenes supporting circular-viewing. We note that SinGRAV inherits some limitations of [6], being unable to handle scenes with a dominant object that is highly structure-sensitive (e.g., human head).

3 Method

SinGRAV learns a powerful generative model for generating neural radiance volumes from multi-view observations $\mathcal{X} = \{\boldsymbol{x}^1, \ldots, \boldsymbol{x}^m\}$ of a single scene, where m denotes the viewpoint index. In contrast to learning class-specific priors, our specific aim is to learn the internal distribution of the input scene. To this end, we resort to a voxel-based neural scene representation, paired with convolutional generators, which inherently possess spatial locality bias, with limited receptive fields for learning over plenty of local regions. The generative model is learned via adversarial generation and discrimination through 2D projections of the generated volumes. During training, the camera pose is randomly selected from the training set. We will omit the notion of the camera pose for brevity. Overall, we use a multi-scale framework to learn properties at different scales ranging from global configurations to local fine texture details, and have to tackle the spatial geometric arrangement issues in 3D. Fig.2 presents an overview.

3.1 Neural Radiance Volume and Rendering

The generated scene is represented by a discrete 3D voxel grid, and is to be produced by a 3D convolutional network. Each voxel center stores a 4-channel vector that contains a density scalar σ and a color vector $\boldsymbol{c}$. Trilinear interpolation is used to define a continuous radiance field in the volume. We use the differentiable volume rendering[15] to render images from generated volumes. Specifically, for each camera ray $\boldsymbol{r}$, the expected color $\hat{C}$ is approximated by integrating over M samples spreading along the ray:

$$\hat{C}(\{\sigma_i, \boldsymbol{c}_i\}_{i=1}^{M}) = \sum_{i=1}^{M} T_i\big(1 - \exp(-\sigma_i \delta_i)\big)\boldsymbol{c}_i,$$

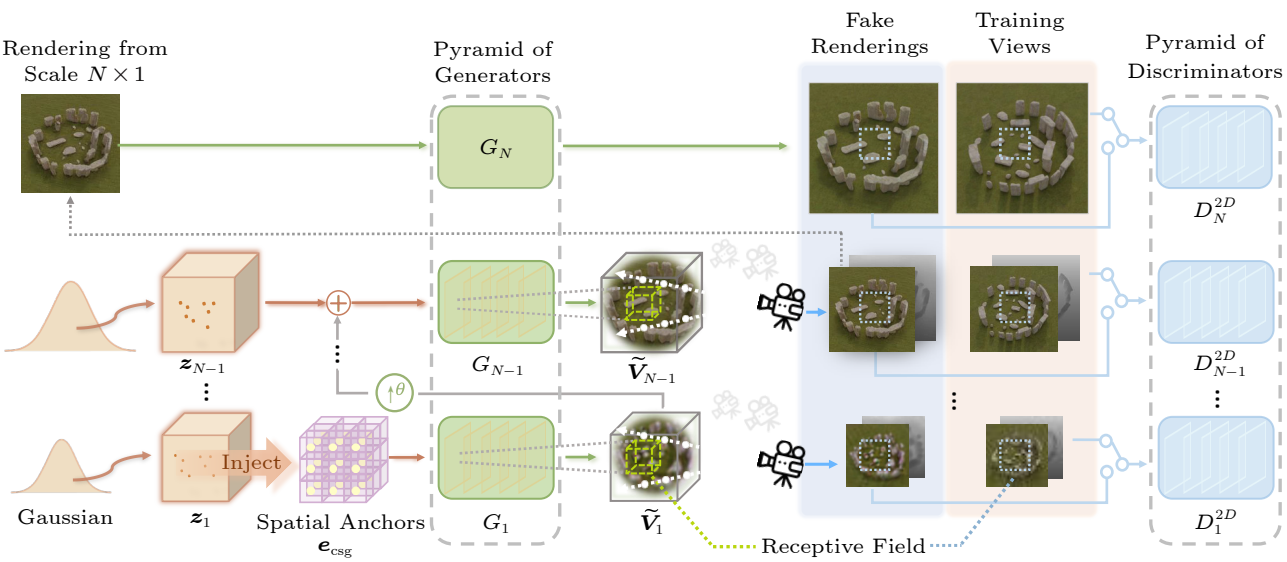


Fig.2. SinGRAV training setup. A series of convolutional generators are trained to generate a scene in a coarse-to-fine manner. At each scale, G_n learns to form a volume via generating realistic 3D overlapping patches, which collectively contribute to a volumetric-rendered imagery indistinguishable from the observation images of the input scene by the discriminator D_n. At the finest scale, the generator G_N operates purely on the 2D domain to super-resolve the imagery produced from scale $N-1$, significantly reducing the computation overhead. $\uparrow^{\theta}$ means volume upsampling.

and $T_i = \exp\Big(-\sum_{j=1}^{i-1}\sigma_j\delta_j\Big)$, where the subscript denotes the sample index between the near t_{n} and far t_{f} bound, $\delta_i = t_{i+1} - t_i$ is the distance between two consecutive samples, and T_i is the accumulated transmittance at sample i, which is obtained via integrating over the preceding samples along the ray. This function is differentiable and enables updating the volume based on the error derived from supervision signals.

Hybrid Multi-Scale Architecture. SinGRAV uses a hybrid multi-scale architecture that contains a generation pyramid with a hybrid use of 2D and 3D convolutional generators, and a discrimination pyramid inspecting local properties on 2D renderings at different scales.

3.1.1 Hybrid Generation Pyramid

There are a series of 3D convolutional generators $\{G_n\}_{n=1}^{N-1}$ and a lightweight 2D convolutional generator G_N (see Fig.2). Overall, 3D generators at coarser scales learn to generate a radiance volume at an increasing resolution, and the volume resolution in the pyramid is increased by a factor of θ between two consecutive scales. At the N-th scale, to avoid the overly high computation issue, we use a lightweight 2D generator G_N to directly super-resolve the rendering from the preceding scale by a factor μ_s, achieving higher-resolution imagery. Importantly, these generators are equipped with limited receptive fields to capture the distribution of local patches, instead of memorizing the whole scene (i.e., reconstruction).

At the coarsest scale, the generation is produced in an unconditional manner, i.e., the radiance volume at the coarsest scale is purely generated from a Gaussian noise volume $\boldsymbol{z}_1$. Notably, we observe that learning the internal distribution with spatial-invariant and receptive field-limited convolutional networks leads to more difficulties in producing plausible 3D structures. Inspired by [32], which alleviates a similar issue in image generation by introducing spatial inductive bias, we introduce spatial inductive bias into our framework by using the 3D normalized Cartesian Spatial Grid (CSG):

$$\boldsymbol{e}_{\mathrm{csg}}(x, y, z) = 2 \times \left[\frac{x}{W} - \frac{1}{2}, \frac{y}{H} - \frac{1}{2}, \frac{z}{U} - \frac{1}{2}\right],$$

where W, H, and U are the size of the volume along the x-, y- and z-axis respectively. As illustrated in Fig.3, the grid is equipped with distinct spatial anchors, empowering the model with better spatial localization. The spatial anchors provided by $\boldsymbol{e}_{\mathrm{csg}}$ are injected into the noise volume $\boldsymbol{z}_1$ at the coarsest level via an element-wise summation operation:

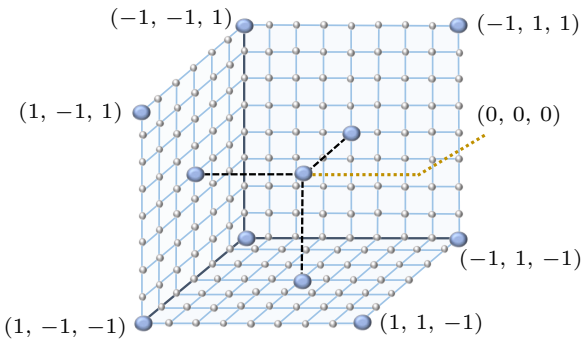


Fig.3. Spatial anchors provided by $\boldsymbol{e}_{\mathrm{csg}}$.

$$\tilde{\boldsymbol{V}}_1 = G_1(\boldsymbol{z}_1, \boldsymbol{e}_{\mathrm{csg}}) = G_1(\boldsymbol{z}_1 + \boldsymbol{e}_{\mathrm{csg}}),$$

where $\boldsymbol{z}_1$, $\boldsymbol{e}_{\mathrm{csg}} \in \mathbb{R}^{3\times U\times H\times W}$. Note we only inject the spatial inductive bias at the coarsest scale, as the positional-encoded information will be propagated through subsequent scales by convolution operations[32]. The receptive field of G_1 is around 40% of the volume of interest; thus G_1 learns to generate the overall layout.

Subsequently, G_n at finer scales $(1 < n < N)$ learns to add details missing from previous scales. Hence, each generator G_n takes as input a spatial noise volume $\boldsymbol{z}_n$ and an upsampled volume of $(\tilde{\boldsymbol{V}}_{n-1})\uparrow^{\theta}$ outputted from G_{n-1}. Specially, prior to being fed into G_n, $\boldsymbol{z}_n$ is added to $(\tilde{\boldsymbol{V}}_{n-1})\uparrow^{\theta}$, and, akin to residual learning[41], G_n only learns to generate missing details:

$$\tilde{\boldsymbol{V}}_n = (\tilde{\boldsymbol{V}}_{n-1})\uparrow^{\theta} + G_n\big(\boldsymbol{z}_n, (\tilde{\boldsymbol{V}}_{n-1})\uparrow^{\theta}\big),\ 1 < n < N.$$

At the finest scale, a 2D convolutional generator G_N takes as input only the rendering $\tilde{\boldsymbol{x}}_{N-1}$ produced from G_{N-1} and outputs a super-resolved image $\tilde{\boldsymbol{x}}_N$ with enhanced details:

$$\tilde{\boldsymbol{x}}_N = G_N(\tilde{\boldsymbol{x}}_{N-1}).$$

G_N utilizes upsampling layers introduced in [3], and produces the final image that is twice the resolution of $\tilde{\boldsymbol{x}}_{N-1}$.

3.1.2 Discrimination Pyramid

For supervising the generation at each scale, SinGRAV resorts to a pyramid of 2D discriminators that operate on 2D images obtained by volume-rendering the generated volume, instead of directly using expen-

sive 3D convolutions. Specifically, at coarser scales $[1, N-1]$, discriminators $\{D_n^{2D}\}_{n=1}^{N-1}$ jointly discriminate on the RGB and depth renderings, of which the resolution between consecutive scales increases by a factor of μ_r, to simultaneously inspect the texture and geometry. Although [18] also adopts joint discrimination on depth, their discriminator is to inspect the entire interior layout, while our $\{D_n^{2D}\}_{n=1}^{N-1}$ is to learn plausible geometric arrangements from local patches. At the finest scale, where the input is an RGB image directly generated from the generator, to improve the view consistency, we adopt the dual discrimination strategy as in [2] for the discriminator D_N^{2D} by concatenating the naively super-resolved $(\tilde{\boldsymbol{x}}_{N-1})\uparrow^{\mu_s}$ with the output from G_N. Overall, to progressively learn the internal priors, the receptive field of D_n^{2D} is so limited that local regions in the generated scene can be gradually crafted.

3.2 Training Loss

The multi-scale architecture is sequentially trained, from the coarsest to the finest scale. We construct a pyramid of resized input observations, $\mathcal{X}_n = \{\boldsymbol{x}_n\}_{n=1}^{N}$, for providing supervisions. The GANs at coarser scales are frozen once trained. The training objective is as follows:

$$\min_{G_n} \max_{D_n^{2D}} \mathcal{L}_{\text{adv}}(G_n, D_n^{2D}) + \mathcal{L}_{\text{rec}}(G_n) + \mathbb{1}(n = N)\mathcal{L}_{\text{swd}}(G_n),$$

where $\mathcal{L}_{\text{adv}}$ is an adversarial term, $\mathcal{L}_{\text{rec}}$ is a reconstruction term as similar in [6], $\mathbb{1}(\cdot)$ is an indicator function to activate the associated term iff the condition is satisfied, and $\mathcal{L}_{\text{swd}}$ calculates the Sliced Wasserstein Distance (SWD) as in [42]. SWD measures the distance between the textural distributions of two images, while neglecting the difference of the global layouts. Concretely, $\mathcal{L}_{\text{swd}}$ is given by: $\mathcal{L}_{\text{swd}} = \mathcal{L}_{\text{swd}}(\boldsymbol{x}_N, \tilde{\boldsymbol{x}}_N) = \sum_{k=1}^{K} \mathcal{L}_{\text{swd}}(\tilde{\boldsymbol{f}}^k, \boldsymbol{f}^k)$, where $\tilde{\boldsymbol{f}}^k$ and $\boldsymbol{f}^k$ are features from layer k of a pre-trained VGG-19 network[43] (please refer to [42] for more details). In the following, we elaborate the designs of $\mathcal{L}_{\text{adv}}$ and $\mathcal{L}_{\text{rec}}$.

3.2.1 *Adversarial Loss*

We use the WGAN-GP loss[44] as $\mathcal{L}_{\text{adv}}$ for stabilizing the training. The discrimination score is obtained by averaging over the patch discrimination map of D_n^{2D}. During training, we render the depth from the generated volume, and concatenate the depth and color images for the input to $\{D_n^{2D}\}_{n=1}^{N-1}$ as fake samples, while real samples for depth can be derived with multi-view geometry techniques trivially. For preparing the real input to discriminator D_N^{2D}, we upsample the resized observation $\boldsymbol{x}_{N-1}$ via bilinear upsampling and concatenate the upsampled image with the ground truth observation $\boldsymbol{x}_N$.

3.2.2 *Reconstruction Loss*

Inspired by [6], we introduce a specific set of input noise volumes to ensure that they can reconstruct the underlying scene depicted in the observations $\mathcal{X}$. Specifically, a set of fixed noise volumes are defined as $\{\boldsymbol{z}_n^*\}_{n=1}^{N-1} = \{\boldsymbol{z}_1^*, 0, \ldots, 0\}$. The reconstructed radiance volumes and associated renderings are denoted as $\{\boldsymbol{V}_n^*\}_{n=1}^{N-1}$ and $\{\tilde{\boldsymbol{x}}_n^*, \tilde{\boldsymbol{d}}_n^*\}_{n=1}^{N}$, respectively. Then the reconstruction loss $\mathcal{L}_{\text{rec}}$ is defined as:

$$\mathcal{L}_{\text{rec}} = \lambda_c ||\tilde{\boldsymbol{x}}_n^* - \boldsymbol{x}_n||_2^2 + \mathbb{1}(n < N) \cdot \lambda_d ||\tilde{\boldsymbol{d}}_n^* - \boldsymbol{d}_n||_2^2, \quad (1)$$

where λ_c and λ_d are balance parameters and we set $\lambda_c = 10$ and $\lambda_d = 30$. As shown in (1), we use supervisions on both color images and depth images for achieving higher quality. Note that the depth penalty term is removed at the last scale since G_N, which only works in the color image domain. As demonstrated and discussed later in Subsection 4.4.2, our method can be applied even in cases where ground-truth depth maps are unavailable. In such scenarios, reconstructed depth maps generated by NeRF[15] models or our proposed framework can be utilized.

4 Experiments

4.1 Settings

4.1.1 *Data*

To evaluate the proposed framework, we collect observation images from a dozen of diverse scenes, which exhibit ample variations over the global arrangements and constituents. Specifically, we collect 3D scene assets from this website②, under TurboSquid 3D Model License③. For eliminating the influence of data defects including incorrect camera pose estimation, incomplete scene coverage within the multi-view images, etc., we use rendered multi-view

②https://www.turbosquid.com, Jan. 2024.
③https://blog.turbosquid.com/turbosquid-3d-model-license/#3d-model-license, Jan. 2024.

images in our main experiments, comparison experiments, and ablation study. Specifically, we utilize the path-tracing renderer in Blender to get the multi-view RGB-D observation. For data rendering, we scale the scenes so that the volume of interest stays within a cube with side width $= 2$ (within the range $[-1, 1]$). Finally, for each scene, we render 200 observation images that fully cover the scene, for which the camera positions are randomly sampled on a hemisphere. Random natural scene generation of our method is demonstrated upon all collected scenes, whereas more evaluations are conducted on a subset (stonehenge, grass and flowers, and island).

Moreover, in Subsection 4.5, we test our framework on captured data obtained by a hand-hold phone. For captured multi-view images, we utilize COLMAP to reconstruct camera poses, which is a common practice in many neural scene representation frameworks[15, 45].

4.1.2 *Evaluation Measures*

We extend common metrics in single image generation to quantitatively assess $m = 50$ scenes generated from each input scene. For each generated scene, $k = 40$ images at random viewpoints are rendered for evaluation under a multi-view setting: 1) SIFID-MV measures how well the model captures the internal statistics of the input by SIFID[6] averaged over multi-view images of a generated scene; 2) Diversity-MV measures the diversity of generated scenes by the averaged image diversity[6] over multiple views.

4.2 Neural Scene Variations

Fig.1 and Fig.4 present qualitative results of SinGRAV, where the generated scenes depict reasonably new global layouts, and objects with various shapes and realistic looking. These results suggest the efficacy of SinGRAV in modeling the internal patch distribution within the input scene. On grass and flowers exhibiting uniform yet complicated textures over an open field, SinGRAV produces high-quality random generation results. Moreover, SinGRAV is able to capture the global illumination to some extent, as evidenced by the shadows around stones and islands, along with the illumination changes under spinning cameras on samples of mushroom. In Fig.5, we demonstrate examples for extracted meshes of the generated scenes from SinGRAV, which demonstrate the ability to capture the geometric distribution of the exemplar scene. In addition, we present more visual results in the supplementary video[④].

4.3 Comparisons

We compare our framework SinGRAV with three state-of-the-art neural scene generative models, namely, GRAF[4], pi-GAN[1], and GIRAFFE[5]. We use their official codes for training these baselines. Training details of each baseline can be found in the supplementary material[⑤]. For fair comparisons, similar to SinGRAV, we use ground truth cameras when training baselines, instead of using random cameras from a predefined camera distribution. The quantitative and qualitative results are presented in Table 1 and Fig.6, respectively. Table 1 shows that pi-GAN produces the best SIFID-MV score, but the value of Diversity-MV drastically degrades. GRAF and GIRAFFE also exhibit a significantly degraded diversity score. Qualitative results in Fig.6 show that these baselines suffer from severe mode collapse, due to the lack of diverse samples for learning category-level priors.

4.4 Validation of Design Choices

We conduct experiments to evaluate several key design choices and the quantitative results are reported in Table 2.

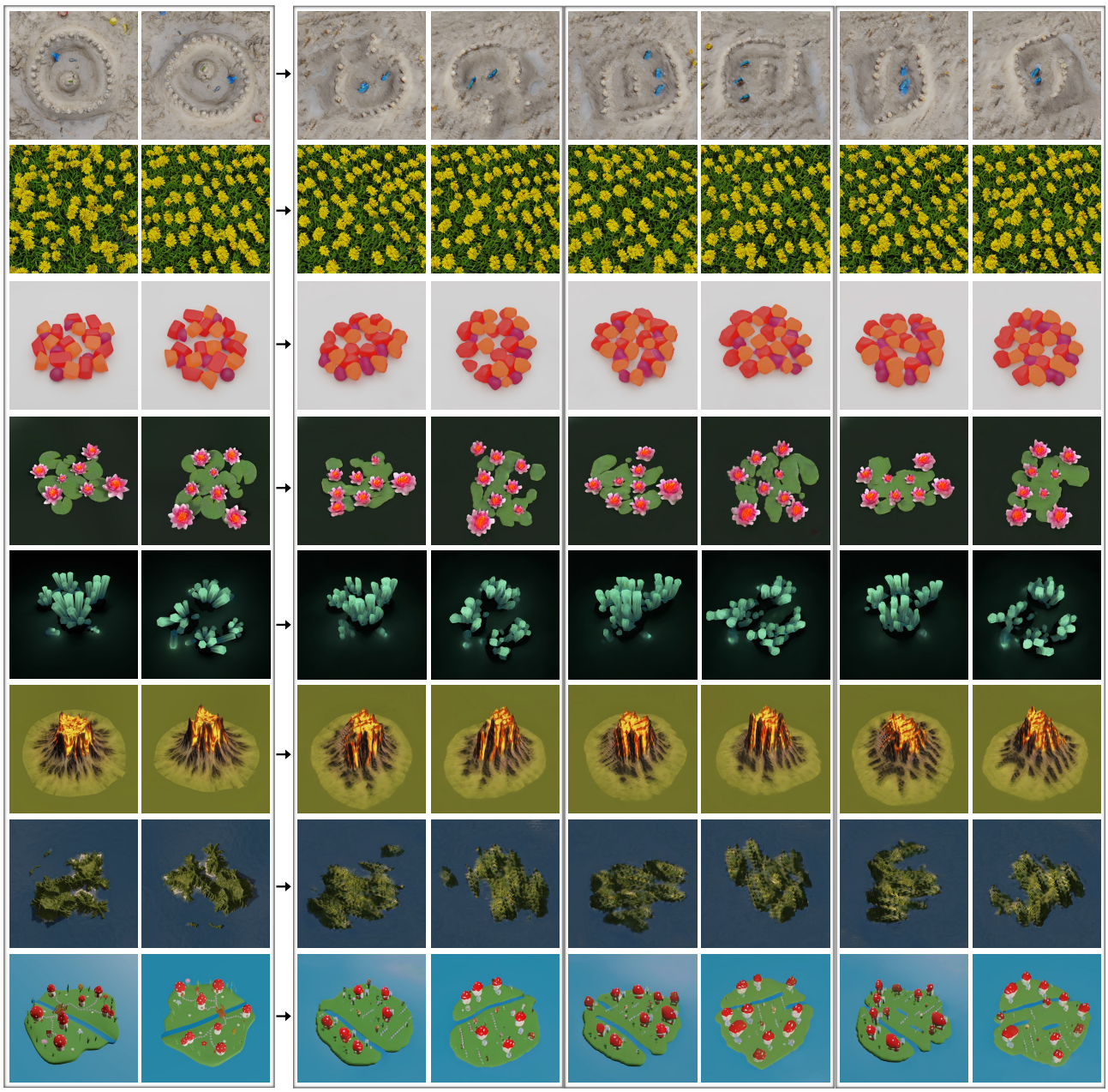
4.4.1 *Spatial Inductive Bias*

We build a variant—SinGRAV (wo. CSG), which is trained without the spatial anchor volume $\boldsymbol{e}_{\text{csg}}$. As shown in Table 2, compared with SinGRAV, SinGRAV (wo. CSG) produces a worse SIFID-MV score (increased by 17%) and an increased Diversity-MV score. While the latter suggests increased diversity, we observe from the visual results that the spatial arrangements of objects deviate significantly from that of the input, producing floating stones, as shown in Fig.7(a).

4.4.2 *Depth Supervision Strategy*

To investigate the role exerted by the depth supervision (depth sup.) and the influence of using re-

④https://youtu.be/n1jF3Sdlqy8, Jan. 2024.
⑤https://arxiv.org/pdf/2210.01202.pdf, Mar. 2024.



(a) (b)

Fig.4. Random scene generation. (a) Sampled views for different training scenes. (b) Rendered views from three randomly generated scenes. In each row, the images shown in (a) and (b) are under the same viewpoints. After training on multi-view observations of an input scene, SinGRAV learns to generate similar scenes with new objects and configurations. Note the observation images of the beach castle at the top row are taken from the real world, and novel beach castles with various layouts are generated by SinGRAV.

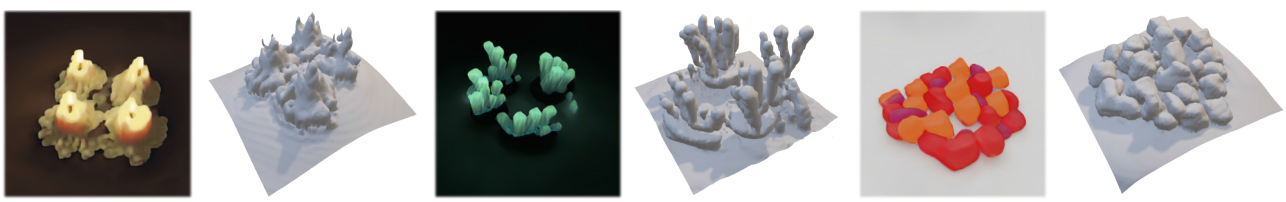


Fig.5. Examples for extracted meshes from generated scenes produced by SinGRAV.

constructed depth cues, we conduct experiments on three variants of SinGRAV, including SinGRAV (wo. depth sup.), SinGRAV (self-depth) and SinGRAV (NeRF-depth).

Table 1. Quantitative Comparisons

Method	Img. Res.	SIFID-MV ↓	Diversity-MV ↑
GRAF[4]	320 × 320	0.444 7	0.133 7
pi-GAN[1]	128 × 128	**0.013 3**	0.115 7
GIRAFFE[5]	320 × 320	0.471 0	**0.319 8**
SinGRAV	320 × 320	**0.111 3**	**0.776 9**

Note: The top two on each metric are bolded. All baselines suffer from severe mode collapse, producing low Diversity-MV scores. Img. Res.: image resolution. ↓: the lower, the better; ↑: the higher, the better.

In the first variant, SinGRAV (wo. depth sup.), we remove the depth supervisions and present the numerical results in Table 2. Visual results are also provided in Fig.7. While the numerical results for this variant are comparable to those of our full model SinGRAV, Fig.7 shows that the extracted point clouds exhibit undesirable geometric structures in the generated scenes of SinGRAV (wo. depth sup.). Additionally, in the supplementary video[⑥], we include multi-view videos to further illustrate the suboptimal geometry produced by this variant. These results emphasize the importance of incorporating depth supervisions during the training.

There are many scenarios where perfect ground-truth depth is not available; thus we further investigate the feasibility of utilizing reconstructed depth information from multi-view RGB images. The variant SinGRAV (NeRF-depth) uses the depth obtained from a NeRF[15] reconstruction of the input scene, and SinGRAV (self-depth) uses the depth obtained from the reconstructed volume with $\boldsymbol{z}^*$ from our framework trained with $\lambda_d = 0$. Table 2 shows that SinGRAV (NeRF-depth) and SinGRAV (self-depth) are able to achieve comparable performance to SinGRAV, implying that SinGRAV is robust to the quality of the depth data. The extracted point clouds from generated scenes, as shown in Fig.7, also demonstrate that using depth supervision coming from reconstructed depth can generate reasonable geometric arrangements. Furthermore, we provide multi-view videos of the generated scenes from two aforementioned variants for enhanced visualization. To encapsulate, depth data obtained via reconstruction methods suffice to guide the learning of global geometric structures, reinforcing the adaptability and versatility of SinGRAV in diverse scenarios.

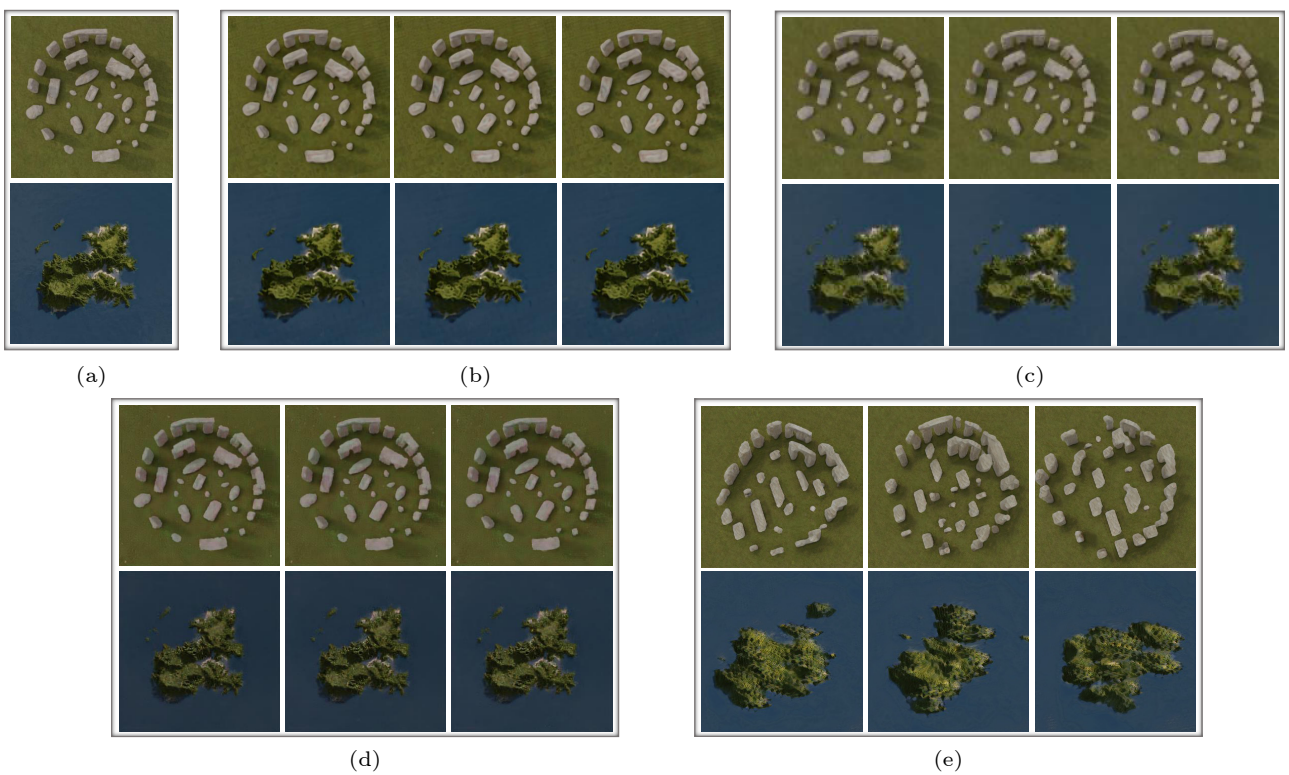


Fig.6. Qualitative results for comparisons. (a) Training scenes. (b) Randomly generated scenes from GRAF[4]. (c) Randomly generated scenes from pi-GAN[1]. (d) Randomly generated scenes from GIRAFFE[5]. (e) Randomly generated scenes from SinGRAV. All baselines encounter severe mode-collapse issues, while SinGRAV generates diverse samples.

⑥https://youtu.be/n1jF3Sdlqy8, Jan. 2024.

Table 2. Numerical Results for Variants of SinGRAV

Variant	Img. Res.	SIFID-MV ↓	Diversity-MV ↑
(wo. CSG)	320^2	0.130 7	**0.843 4**
(wo. depth sup.)	320^2	0.115 7	0.791 0
(NeRF-depth)	320^2	0.129 0	0.804 6
(self-depth)	320^2	0.112 0	0.786 2
(wo. SWD)	320^2	0.271 3	0.773 0
(w. MLP)	160^2	0.184 3	0.523 5
(w. MLP-LLG)	108^2	0.201 3	0.462 1
SinGRAV	320^2	**0.111 3**	0.776 9

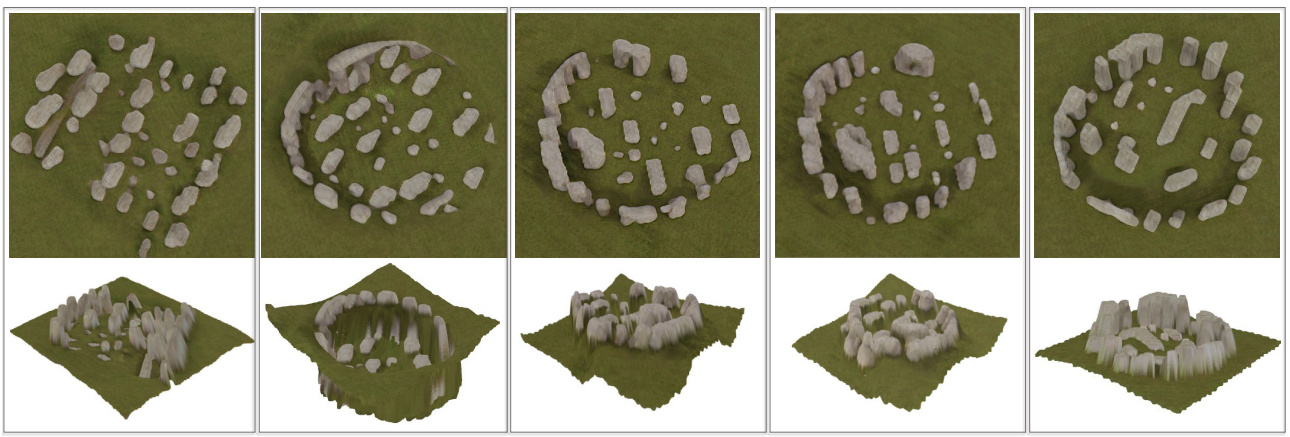
Note: wo.: without; w.: with.

4.4.3 SWD Loss

If $\mathcal{L}_{\text{swd}}$ is eliminated, the internal distribution of generated scenes would differ greatly from that of the input, leading to significantly increased SIFID-MV at the fifth row (wo. SWD) in Table 2. Correspondingly, the visual results in Fig.8 also show that SinGRAV(wo. $\mathcal{L}_{\text{swd}}$) produces blurry and less realistic textures, suggesting the efficacy of $\mathcal{L}_{\text{swd}}$ in improving visual quality.

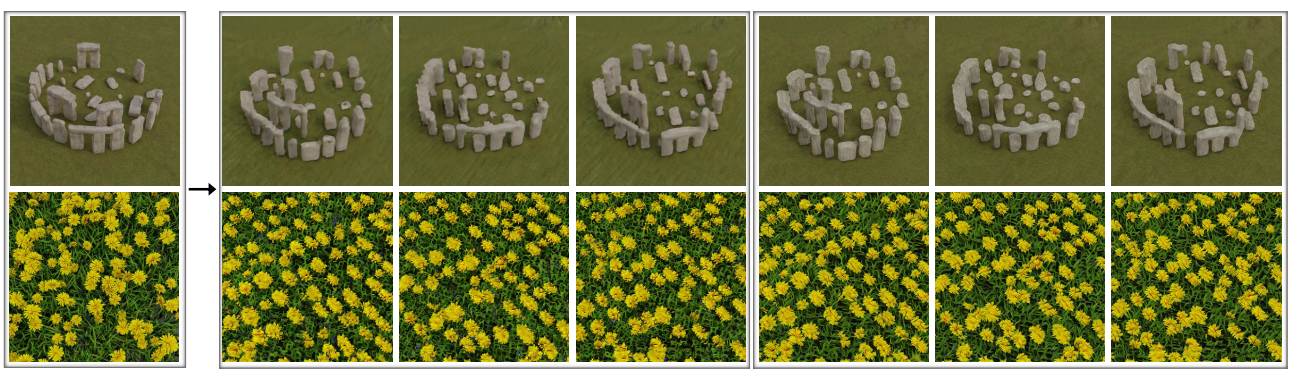
4.4.4 MLP vs Voxel

To validate our choice of adopting the voxel-based representation, we design a variant SinGRAV (w. MLP), which integrates a conditional MLP-based radiance field as in [4]. The numerical results of SinGRAV (w. MLP), which uses a conditional MLP-based radiance field as in GRAF, are reported in Table 2. The highest image resolution is 160×160 due to overly high memory consumption. SinGRAV (w. MLP) degrades in both the quality and diversity, which is also reflected by the visuals in Fig.2. Note that, although the same patch discrimination strate-



(a) (b) (c) (d) (e)

Fig.7. Influence of spatial inductive bias and depth supervision strategies. (a) Results from SinGRAV (wo. CSG). (b) Results from SinGRAV (wo. depth sup.). (c) Results from SinGRAV (NeRF-depth). (d) Results from SinGRAV (self-depth). (e) Results from SinGRAV with GT depth. One generated sample with the corresponding point cloud is shown in (a)–(e). Without the inductive bias or depth supervision, the generated scenes exhibit implausible geometric structures, while the variants that use reconstructed depth maps or ground-truth depth preserve the spatial arrangement well.



(a) (b) (c)

Fig.8. SinGRAV vs SinGRAV (wo. $\mathcal{L}_{\text{swd}}$). (a) Two training scenes. (b) Rendered results of generated scenes from SinGRAV (wo. $\mathcal{L}_{\text{swd}}$). (c) Rendered results of generated scenes from SinGRAV. Without the SWD loss at the finest scale, SinGRAV (wo. $\mathcal{L}_{\text{swd}}$) produces blurry textures and undesired artifacts, while training with SWD loss significantly improves the texture quality.

gy is used, SinGRAV (w. MLP) suffers from severe mode collapse. We believe this is because the generative output of coordinate-based MLPs is very likely to be dominated by the input coordinates. The poor generation is also possibly a product of the conflict between the lack of locality in the fully-connected layers[8] and the limited receptive field in the adversarial training. In addition, we train a variant SinGRAV (w. MLP-LLG), which incorporates a local latent grid (LLG) proposed in [18] to increase the capacity of the MLP-based representation. Fig.9 shows that the generated layouts are improved with the increased locality; however, severe mode collapse still exists. On the other hand, we believe more efforts can be made in the future to adapt MLP-based representations for our task.

(a) (b) (c)

Fig.9. Qualitative results of using MLP-based representations. (a) Training scenes. (b) Generated scenes from SinGRAV (w. MLP). (c) Generated scenes from SinGRAV (w. MLP-LLG). SinGRAV (w. MLP) produces grid patterns, which are alleviated by the use of local latent grid in SinGRAV (w. MLP-LLG). Nevertheless, both variants suffer from severe mode collapse, and generate almost identical scene samples.

4.4.5 *Influence of Varying Pyramid Depth*

We train variants with various numbers of scales. Specifically, for training a variant with t $(t < N)$ scales, we use generators $\{G_n\}_{n=N-t+1}^{N}$ and discriminators $\{D_n^{2D}\}_{n=N-t+1}^{N}$ to preserve the final image resolution. As shown in Fig.10, with less scales, the effective receptive field at the coarsest scale is rather small, resulting in a model that only captures local properties, whereas, using more scales allows modeling plausible global arrangements.

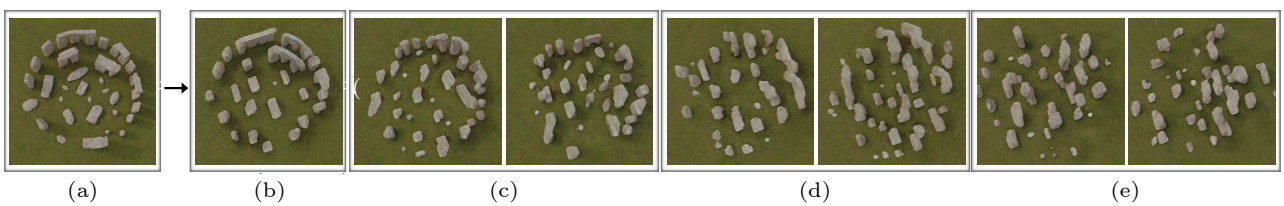
4.5 Results on Real-World Data

To investigate the applicability of SinGRAV on real-world data, we test SinGRAV on a daily scenario where people use a mobile phone to capture multi-view images of a desktop scene. Specifically, we capture dozens of images of two candy piles. The flash is turned on for removing the shadows introduced by the hands and the hand-hold devices. Then we use COLMAP to estimate the camera poses of the captured images. Once the poses are estimated, we adopt MultiNeRF[7] to reconstruct the captured scene. Finally, given the MultiNeRF-parameterized scene, we can render multi-view images with roughly consistent camera-scene distances for training SinGRAV. Since the ground-truth depth maps are not available for images captured by hand-hold phones, we use rendered depth maps from the reconstructed scene to train our framework.

The generated scenes are shown in Fig.11. As shown in Fig.11, SinGRAV is able to produce plausible scenes with considerable diversity, demonstrating the applicability of SinGRAV on real-world captured images. The results further manifest that the proposed framework can achieve appealing results by using reconstructed depth maps from multi-view images. The rendered multi-view observations from more generated scenes can be found in the supplementary video[8].

4.6 Applications

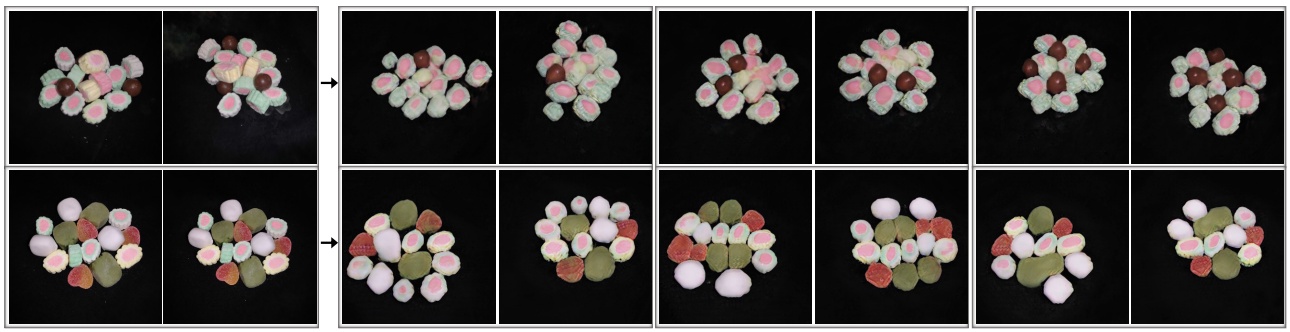
SinGRAV supports various applications, which can be achieved by naively manipulating generated volumes at coarse scales and using subsequent genera-



(a) (b) (c) (d) (e)

Fig.10. Influence of different numbers of scales. (a) Training scene. (b) Generated scene from SinGRAV with $N = 6$. (c) Generated scenes from SinGRAV with $N = 5$. (c) Generated scenes from SinGRAV with $N = 4$. (e) Generated scenes from SinGRAV with $N = 3$. Training with more scales is beneficial to the modeling of global arrangements, while a model with less scales tends to capture only local textures. N is set to 6 by default for SinGRAV.

[7]https://github.com/google-research/multinerf, Dec. 2022.
[8]https://youtu.be/n1jF3Sdlqy8, Jan. 2024.



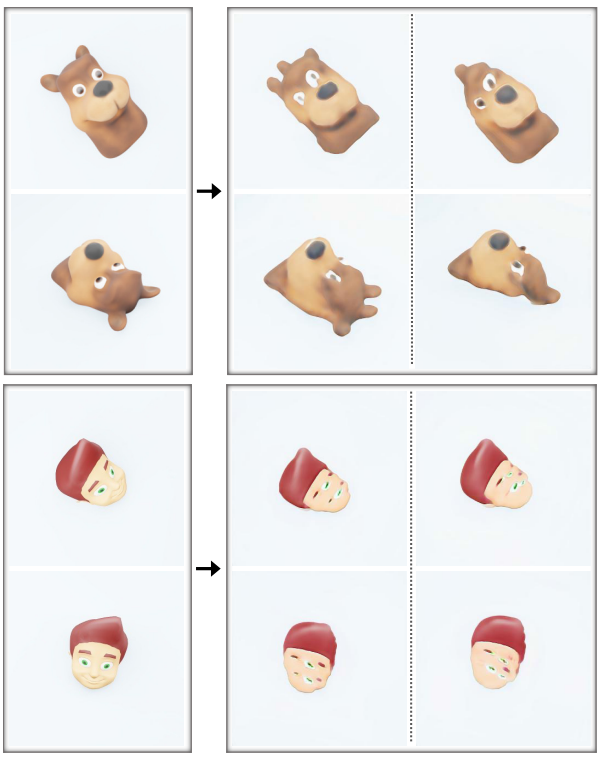
(a) (b)

Fig.11. Random scene samples generated from two real-world indoor scenes. (a) Two views of the training scenes. (b) Three randomly generated scenes from SinGRAV separately trained on the training scenes. Each scene is rendered with the same two views.

tors to harmonize the modifications. Specifically, we derive three applications with SinGRAV, including 3D scene editing with removal and duplicate operations, composition, and animation. Fig.1 presents the results. More results and implementation details are given in the supplementary material[⑨].

5 Conclusions

In this work, we made an attempt to learn a deep generative neural scene model from visual observations of a single scene. Once trained on a single scene, the model can generate novel scenes with plausible geometries and arrangements, which can be rendered with pleasing viewing effects. The importance of key design choices is validated. Despite successful demonstrations, our proposed SinGRAV has a few limitations. While SinGRAV learns from a single scene, bypassing the need for collecting data from many homogeneous 3D samples, multi-view images with sufficient coverage rate of the scene are yet required. Moreover, albeit validated, the use of voxel grids inherently limits the network capacity in modeling fine details, consequently hindering the model from achieving high-resolution imagery. Our remedy is to incorporate a 2D neural renderer that operates on the 2D domain to super-resolve the imagery, which inevitably introduces the multi-view inconsistency. There are view inconsistencies in complex textural areas, e.g., the thin structures in the grass and flowers scene. A future direction would be to overturn this design, with more endeavors on exploiting MLP-based representations to model continuous volumes. We also noticed that, when the exemplar scene is dominant by a structure-sensitive object, as demonstrated in Fig.12, SinGRAV may produce less satisfying results. Besides, the proposed method, in its current form, sometimes produces artifacts in the background, as it does not incorporate special designs for modeling the background. Hence, it would also be worth addressing this issue, especially for scenes with complicated backgrounds, potentially with special considerations on the foreground-background continuity.



(a) (b)

Fig.12. Results for structure-sensitive object-centric scenes. (a) Two training scenes. (b) Randomly generated scenes from SinGRAV. When the exemplar scene is predominantly occupied by a single object, SinGRAV possibly produces less satisfying results due to the insufficiency of exploitable patch priors and unawareness of the underlying semantics.

[⑨] https://arxiv.org/pdf/2210.01202.pdf, Mar. 2024.

Acknowledgement We thank Prof. Yi-Xin Zhuang from Fuzhou University for helpful discussions.

Conflict of Interest The authors declare that they have no conflict of interest.

References

[1] Chan E R, Monteiro M, Kellnhofer P, Wu J J, Wetzstein G. pi-GAN: Periodic implicit generative adversarial networks for 3D-aware image synthesis. In *Proc. the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2021, pp.5795–5805. DOI: 10.1109/CVPR46437.2021.00574.

[2] Chan E R, Lin C Z, Chan M A, Nagano K, Pan B X, de Mello S, Gallo O, Guibas L, Tremblay J, Khamis S, Karras T, Wetzstein G. Efficient geometry-aware 3D generative adversarial networks. In *Proc. the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2022, pp.16102–16112. DOI: 10.1109/CVPR52688.2022.01565.

[3] Gu J T, Liu L J, Wang P, Theobalt C. StyleNeRF: A style-based 3D aware generator for high-resolution image synthesis. In *Proc. the 10th International Conference on Learning Representations*, Apr. 2022.

[4] Schwarz K, Liao Y Y, Niemeyer M, Geiger A. GRAF: Generative radiance fields for 3D-aware image synthesis. In *Proc. the 34th International Conference on Neural Information Processing Systems*, Dec. 2020, Article No. 1692, pp.20154–20166. DOI: 10.5555/3495724.3497416.

[5] Niemeyer M, Geiger A. GIRAFFE: Representing scenes as compositional generative neural feature fields. In *Proc. the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2021, pp.11448–11459. DOI: 10.1109/CVPR46437.2021.01129.

[6] Shaham T R, Dekel T, Michaeli T. SinGAN: Learning a generative model from a single natural image. In *Proc. the 2019 IEEE/CVF International Conference on Computer Vision*, Oct. 2019, pp.4569–4579. DOI: 10.1109/ICCV.2019.00467.

[7] Shocher A, Bagon S, Isola P, Irani M. InGAN: Capturing and retargeting the "DNA" of a natural image. In *Proc. the 2019 IEEE/CVF International Conference on Computer Vision*, Oct. 2019, pp.4491–4500. DOI: 10.1109/ICCV.2019.00459.

[8] Ding X H, Chen H H, Zhang X Y, Han J G, Ding G G. RepMLPNet: Hierarchical vision MLP with re-parameterized locality. In *Proc. the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2022, pp.568–577. DOI: 10.1109/CVPR52688.2022.00066.

[9] Chen Z Q, Zhang H. Learning implicit fields for generative shape modeling. In *Proc. the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2019, pp.5932–5941. DOI: 10.1109/CVPR.2019.00609.

[10] Park J J, Florence P, Straub J, Newcombe R, Lovegrove S. DeepSDF: Learning continuous signed distance functions for shape representation. In *Proc. the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2019, pp.165–174. DOI: 10.1109/CVPR.2019.00025.

[11] Michalkiewicz M, Pontes J K, Jack D, Baktashmotlagh M, Eriksson A. Implicit surface representations as layers in neural networks. In *Proc. the 2019 IEEE/CVF International Conference on Computer Vision*, Oct. 27–Nov. 2, 2019, pp.4742–4751. DOI: 10.1109/ICCV.2019.00484.

[12] Takikawa T, Litalien J, Yin K X, Kreis K, Loop C, Nowrouzezahrai D, Jacobson A, McGuire M, Fidler S. Neural geometric level of detail: Real-time rendering with implicit 3D shapes. In *Proc. the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2021, pp.11353–11362. DOI: 10.1109/CVPR46437.2021.01120.

[13] Martel J N P, Lindell D B, Lin C Z, Chan E R, Monteiro M, Wetzstein G. Acorn: Adaptive coordinate networks for neural scene representation. *ACM Trans. Graphics*, 2021, 40(4): Article No. 58. DOI: 10.1145/3450626.3459785.

[14] Nguyen-Phuoc T, Li C, Theis L, Richardt C, Yang Y L. HoloGAN: Unsupervised learning of 3D representations from natural images. In *Proc. the 2019 IEEE/CVF International Conference on Computer Vision*, Oct. 2019, pp.7587–7596. DOI: 10.1109/ICCV.2019.00768.

[15] Mildenhall B, Srinivasan P P, Tancik M, Barron J T, Ramamoorthi R, Ng R. NeRF: Representing scenes as neural radiance fields for view synthesis. In *Proc. the 16th European Conference on Computer Vision*, Aug. 2020, pp.405–421. DOI: 10.1007/978-3-030-58452-8_24.

[16] Wiles O, Gkioxari G, Szeliski R, Johnson J. SynSin: End-to-end view synthesis from a single image. In *Proc. the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2020, pp.7465–7475. DOI: 10.1109/CVPR42600.2020.00749.

[17] Nguyen-Phuoc T, Richardt C, Mai L, Yang Y L, Mitra N. BlockGAN: Learning 3D object-aware scene representations from unlabelled images. In *Proc. the 34th Conference on Neural Information Processing Systems*, Dec. 2020, pp.6767–6778.

[18] DeVries T, Bautista M A, Srivastava N, Taylor G W, Susskind J M. Unconstrained scene generation with locally conditioned radiance fields. In *Proc. the 2021 IEEE/CVF International Conference on Computer Vision*, Oct. 2021, pp.14284–14293. DOI: 10.1109/ICCV48922.2021.01404.

[19] Wang W Y, Xu Q G, Ceylan D, Mech R, Neumann U. DISN: Deep implicit surface network for high-quality single-view 3D reconstruction. In *Proc. the 33rd International Conference on Neural Information Processing Systems*, Dec. 2019, Article No. 45. DOI: 10.5555/3454287.3454332.

[20] Sitzmann V, Thies J, Heide F, Nießner M, Wetzstein G, Zollhöfer M. DeepVoxels: Learning persistent 3D feature embeddings. In *Proc. the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2019, pp.2432–2441. DOI: 10.1109/CVPR.2019.00254.

[21] Thies J, Zollhöfer M, Nießner M. Deferred neural rendering: Image synthesis using neural textures. *ACM Trans. Graphics*, 2019, 38(4): Article No. 66. DOI: 10.1145/3306346.3323035.

[22] Liu L J, Gu J T, Lin K Z, Chua T S, Theobalt C. Neural sparse voxel fields. In *Proc. the 34th Conference on Neural Information Processing Systems*, Dec. 2020, pp.15651–15663.

[23] Rebain D, Jiang W, Yazdani S, Li K, Yi K M, Tagliasacchi A. DeRF: Decomposed radiance fields. arXiv: 2011.12490, 2020. https://doi.org/10.48550/arXiv.2011.12490, Mar. 2024.

[24] Zhang K, Riegler G, Snavely N, Koltun V. NeRF++: Analyzing and improving neural radiance fields. arXiv: 2010.07492, 2020. https://doi.org/10.48550/arXiv.2010.07492, Mar. 2024.

[25] Lindell D B, Martel J N P, Wetzstein G. AutoInt: Automatic integration for fast neural volume rendering. In *Proc. the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2021, pp.14551–14560. DOI: 10.1109/CVPR46437.2021.01432.

[26] Wizadwongsa S, Phongthawee P, Yenphraphai J, Suwajanakorn S. Nex: Real-time view synthesis with neural basis expansion. In *Proc. the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2021, pp.8530–8539. DOI: 10.1109/CVPR46437.2021.00843.

[27] Martin-Brualla R, Radwan N, Sajjadi M S M, Barron J T, Dosovitskiy A, Duckworth D. NeRF in the Wild: Neural radiance fields for unconstrained photo collections. In *Proc. the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2021, pp.7206–7215. DOI: 10.1109/CVPR46437.2021.00713.

[28] Lin C H, Ma W C, Torralba A, Lucey S. BARF: Bundle-adjusting neural radiance fields. arXiv: 2104.06405, 2021. https://doi.org/10.48550/arXiv.2104.06405, Mar. 2024.

[29] Wang Z R, Wu S Z, Xie W D, Chen M, Prisacariu V A. NeRF–: Neural radiance fields without known camera parameters. arXiv:2102.07064, 2022. https://doi.org/10.48550/arXiv.2102.07064, Mar. 2024.

[30] Lombardi S, Simon T, Schwartz G, Zollhoefer M, Sheikh Y, Saragih J. Mixture of volumetric primitives for efficient neural rendering. arXiv: 2103.01954, 2021. https://doi.org/10.48550/arXiv.2103.01954, Mar. 2024.

[31] Karnewar A, Wang O, Ritschel T, Mitra N J. 3inGAN: Learning a 3D generative model from images of a self-similar scene. In *Proc. the 2022 International Conference on 3D Vision*, Sept. 2022, pp.342–352. DOI: 10.1109/3DV57658.2022.00046.

[32] Xu R, Wang X T, Chen K, Zhou B L, Loy C C. Positional encoding as spatial inductive bias in GANs. In *Proc. the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2021, pp.13564–13573. DOI: 10.1109/CVPR46437.2021.01336.

[33] Son M J, Park J J, Guibas L, Wetzstein G. SinGRAF: Learning a 3D generative radiance field for a single scene. In *Proc. the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2023, pp.8507–8517. DOI: 10.1109/CVPR52729.2023.00822.

[34] Goodfellow I J, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y. Generative adversarial nets. In *Proc. the 27th International Conference on Neural Information Processing Systems*, Dec. 2014, pp.2672–2680. DOI: 10.5555/2969033.2969125.

[35] Karras T, Aila T, Laine S, Lehtinen J. Progressive growing of GANs for improved quality, stability, and variation. arXiv:1710.10196, 2017. https://doi.org/10.48550/arXiv.1710.10196, Mar. 2024.

[36] Brock A, Donahue J, Simonyan K. Large scale GAN training for high fidelity natural image synthesis. In *Proc. the 7th International Conference on Learning Representations*, May 2019.

[37] Karras T, Laine S, Aila T. A style-based generator architecture for generative adversarial networks. In *Proc. the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2019, pp.4396–4405. DOI: 10.1109/CVPR.2019.00453.

[38] Karras T, Laine S, Aittala M, Hellsten J, Lehtinen J, Aila T. Analyzing and improving the image quality of StyleGAN. In *Proc. the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2020, pp.8107–8116. DOI: 10.1109/CVPR42600.2020.00813.

[39] Karras T, Aittala M, Laine S, Härkönen E, Hellsten J, Lehtinen J, Aila T. Alias-free generative adversarial networks. In *Proc. the 35th Conference on Neural Information Processing Systems*, Dec. 2021, pp.852–863.

[40] Dhariwal P, Nichol A. Diffusion models beat GANs on image synthesis. In *Proc. the 35th Conference on Neural Information Processing Systems*, Dec. 2021, pp.8780–8794.

[41] He K M, Zhang X Y, Ren S Q, Sun J. Deep residual learning for image recognition. In *Proc. the 2016 IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2016, pp.770–778. DOI: 10.1109/CVPR.2016.90.

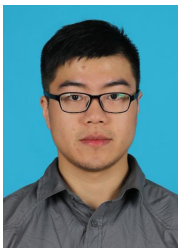
[42] Heitz E, Vanhoey K, Chambon T, Belcour L. A sliced Wasserstein loss for neural texture synthesis. In *Proc. the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2021, pp.9407–9415. DOI: 10.1109/CVPR46437.2021.00929.

[43] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. In *Proc. the 3rd International Conference on Learning Representations*, May 2015.

[44] Gulrajani I, Ahmed F, Arjovsky M, Dumoulin V, Courville A. Improved training of Wasserstein GANs. In *Proc. the 31st International Conference on Neural Information Processing Systems*, Dec. 2017, pp.5769–5779. DOI: 10.5555/3295222.3295327.

[45] Wang P, Liu L J, Liu Y, Theobalt C, Komura T, Wang W P. NeuS: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. In *Proc. the 35th Conference on Neural Information Processing Systems*, Dec. 2021, pp.27171–27183.

Yu-Jie Wang is currently a Ph.D. candidate in computer science and technology at Shandong University, Qingdao. She is also a visiting student at Peking University, Beijing. She received her M.S. degree in computer science and technology from Tianjin University, Tianjin, in 2019. Her research interests include computer-generated holography, and 3D scene reconstruction and generation.



Xue-Lin Chen is currently a senior researcher at the Center of Visual Computing, Tencent AI Lab, Shenzhen. He obtained his Ph.D. degree in computer science from Shandong University, Qingdao, in 2020. During his Ph.D. study, he also worked as a visiting Ph.D. student at the Smart Geometry Processing Group, University College London, London. His research interests are in computer graphics, visual computing, and deep learning. Currently, his focus is on generative models for creating virtual entities, including objects, scenes, and humans within the virtual universe.

Bao-Quan Chen is a professor of the School of Intelligence Science and Technology at Peking University, Beijing. He received his Ph.D. degree in computer science from the State University of New York at Stony Brook, New York, in 1999, and his M.S. degree in electronic engineering from Tsinghua University, Beijing, in 1994. His research interests generally lie in computer graphics, computer vision, visualization, and human-computer interaction. He has served as associate editor of ACM Transactions on Graphics (TOG)/IEEE Transactions on Visualization and Graphics (TVCG), conference steering committee member of ACM SIGGRAPH Asia/IEEE VIS, conference chair of SIGGRAPH Asia 2014/IEEE Visualization 2005, and program chair of IEEE Visualization 2004.