Axis-attention-enhanced generative network for the synthesis of 3D micro-structures

Anonymous ECCV submission

Paper ID 484

Abstract. Synthesizing 3D micro-structure models has significant influence in many areas, such as biology, mechanical and material science. This task is very challenging due to the complex 3D shape and the target of the synthesized model should be strictly similar to the real exemplar in both visual, geometry and statistical senses. To handle this, we present 3D-axisGAN, a novel 3D axis-attention-enhanced generative network, which consists of a spatial-spectral-aware encoding sub-network and a generative adversarial sub-network. A novel 3D-axis attention module is proposed to incorporate into each sub-net, which effectively enhances the intermediate feature maps with attention computation from x, y and z-axis respectively. A comparison study on five datasets demonstrates that our network outperforms state-of-the-art methods, producing synthesized 3D micro-structure with much higher visual, geometry quality and desirable statistical metrics.

Keywords: 3D synthesis, GAN, attention, 3D micro-structure

1 Introduction

3D synthesis of micro-structures with fine-grained details has shown its significance in the recent development of biological, mechanics and material science, such as scaffolds in bone regeneration [5] and tissue engineering [17], strong supporting or filled structure with lightweight in additive manufacturing [8] and computational advanced materials design [37]. However, this task is very challenging, not only because of the intricate internal structural shape of the material by nature (e.g.*metal foam Ni*, in Fig.1(a)), but also the scarcity of 3D digital models (e.g.*bones*, in Fig.2(a)). To this end, a method that could make use of an exemplar via a 3D CT scan, of which the size is relatively small, and synthesize visually, geometrically and statistically similar outputs without size constraint to meet practical necessity is required.

There exist works aiming at this problem [37, 8], which had already led to a sort of remarkable progress. However, usually they were only capable of replicating outputs with regularly, periodically distributed shapes, but failed to intrinsically model and capture the highly varied, randomness and complex shapes which are quite ubiquitous in the real-world. Example-based methods [19, 7, 48] could generate results with a locally similar appearance of the example, while the synthesized results fail to model the long-range connectivity and dependency

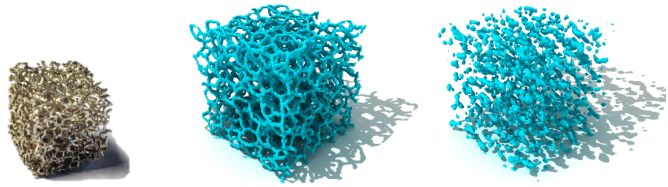

Fig. 1. Complexity of 3d micro-structures. (a) is the real 3D micro-structure material (*Metal foam Ni*); (b) illustrates the visualization of the CT scan of (a); (c) illustrates the failure case with disconnected and floating parts.

due to the lack of global constraints, which still exists floating and disconnected parts as illustrated in Fig. 1(c).

Recently, deep learning techniques on 3D shape generation [42, 44, 21, 20] have gained great performance improvement, which shows good results in generating 3d man-made models with semantic context, regular and repetitive shape, such as chairs, cars, planes in shapenet [6]. The quality of their synthesized results depends on the shape of surface appearance without considering internal structures. However, when synthesizing 3D micro-structure, these architectures may also generate failure cases as illustrated in Fig.1(c), due to the lack of specific modules target with internal long-range connectivity. Besides, since the limited size of the receptive field, the architectures may mainly catch context information within a range instead of coping with all positions in the spatial domain. To mitigate this, the attention mechanism [2, 40, 9, 47] can enable the model to focus on the most relevant part of images or features as needed. Inspired by this, we propose a 3D cross-axis attention module to calculate attention along x, y and z-axis respectively, to capture adequate contextual information from different directions. Due to the input resolution is unwieldy, we use 3D convolution with anisotropic stride on x, y and z-axis to reduce the overlapping of receptive fields, which drastically reduces the computation burden caused by large resolution of feature maps. By design, this 3D attention module can be easily inserted between consecutive convolution blocks in our deep neural networks.

To be specific, we propose 3D-axisGAN Network, a novel generative network for synthesizing 3D complex micro-structures, which contains a spatial-spectral encoding sub-network and a generator-discriminator sub-network (Fig.2(c)). We also design a novel 3d axis-attention module to capture features from three directions in spatial space and insert them into mid-level layers of the sub-networks. Our network is efficiently trained on volumetric samples (Fig.2(b)) extracted from a single piece of material exemplar (e.g., bones; Fig.2(a)) of a limited size. The training process requires no manual annotations. At the running time, our network takes a random vector in the latent space as input. It then generates 3D results with arbitrary sizes (Fig.2(d)) which strongly resemble the exemplar in both visual and statistical senses. We extensively evaluated our results through qualitative and quantitative analyses. Experiments on five datasets demonstrate that our 3D-axisGAN Network outperforms state-of-the-art methods visually and statistically. The effectiveness of our proposed module and network is also validated by ablation experiments.

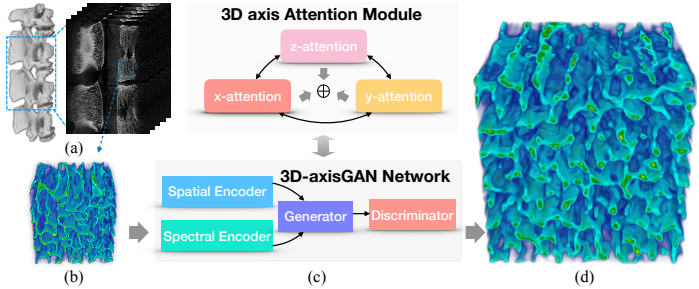

Fig. 2. Given a single 3D exemplar as input (b) extracted from a CT scan of bone (a), our 3D axis-attention enhanced generative network (c) (dubbed 3D-axisGAN) synthesizes complex 3D micro-structures of arbitrary sizes (d) that visually, geometrically and statistically resembles the input. The 3D axis attention module in (c) captures the contextual information along three axial (x, y, z) directions of the 3D feature maps, which is inserted in 3D-axisGAN.

Our technical contributions are summarized as follows:

- A novel attention-enhanced generative framework, 3D-axisGAN, for the synthesis of complex 3D micro-structures is proposed, which encodes both spatial and spectral features.
- A novel 3D axis-attention module along x, y and z-axis is proposed to capture the long-range connectivity and dependency, which generates results with higher visual, geometry quality and desirable statistical metrics.
- Our trained framework could generate 3d micro-structure with an arbitrary size, in which training samples are extracted from a small piece of examplar.
- Comprehensive qualitative and quantitative evaluations demonstrate that our framework significantly outperforms state-of-the-art methods.

2 Related Work

3D content generation using deep learning. Generating 3D shapes and objects has been attracted increasing interest. Wu et al. [43] proposed a generative adversarial networks [11, 34, 14] to generate 3D content. Most aforementioned studies on 3D content generation aim to synthesize man-made objects, such as chairs and airplanes. However few approaches focus on generating 3D objects with intricate internal structures. A recent work [29] is proposed to generate 3D porous materials via a direct usage of GAN, which is barely adequate to derive materials with desired properties. In summary, current studies have not carefully considered the task of generating 3D models with complex internal structures. Therefore, in this paper, we attempt to propose a 3D-axisGAN for this task and discuss the effectiveness of the novel terms added to the baseline GAN model.
Attention Models. Attention mechanism [40] has been one of the most promising mechanisms to be integrated into a deep learning framework. It has been proved to be very effective in many vision tasks [28, 47, 45, 38, 26]. Attention mechanism allows the model to focus on the most relevant part of images or

features as needed. Among this, self-attention [40] has been proposed for calculating the response at a position in a sequence by considering all positions within the same sequence. Zhang et al.[47] show that the self-attention model can capture the multi-level dependencies across image regions which could generate fine details based on the GAN framework. Parmar et al. [33] proposed an Image Transformer model, which adds the self-attention into an auto-regressive model for generating images. Yao et al. [46] combine the self-attention with a residual feature map to conduct style transfer, which is to catch salient characteristics within content images. Park et al. [32] have conduct arbitrary style transfer based on the style-attentional Networks.

Image and texture synthesis. Image and texture synthesis can be divided into two categories: conventional exampled based methods and deep learning methods. Example-based texture synthesis method can generate large-size images with similar patterns (e.g., [22] and [41]) given 2D image as input. Among them, studies [19, 7, 27], [48] employed 2D images or 3D cubic to synthesize 3D structures; however, only limited information is contained in 2D slices and significant memory computational cost when enlarging the size of a 3D cubical region in 3D case. Deep learning-based methods [10, 39, 25] have enabled general users to obtain visually compelling image synthesis results. In order to capture the intrinsic structure and style of the exemplar, Gatys [10] proposed the Gram matrix as a feature descriptor for texture synthesis, nicely producing visually appealing results. Li and Wand [24] compute patch-based texture synthesis with deep neural features, which could generate plausible style transfer results but not considering global control. Sendik and Cohen-Or [35] and Bergmann et al.[3] proposed advanced methods for handling image textures with strong periodicity.

3 Approach

We present a novel 3D cross-axis attention-enhanced generative network that enhances the correlation among x-axis, y-axis and z-axis. In the following sections, we detail the 3D axis attention module and present the generative framework with designed loss functions.

3.1 3D Attention Module

Attention module architecture

The input to our attention module is a set of feature maps $\mathbf{F} \in \mathbb{R}^{C\times H\times W\times D}$, where H, W, D and C denote width, height, length, and channel, respectively. The output has the same size as input feature maps, where the module can be plugged into our architectures at any resolution. We split our 3D attention module into several sequential operations to clearly illustrate each step and its purpose shown in Fig. 3.

Channel reduction reduces the size of input channel from C to C_l through convolutions layers along x, y and z-axis respectively, which could efficiently model the relationships. Specifically, ϕ is dedicated to attention computation

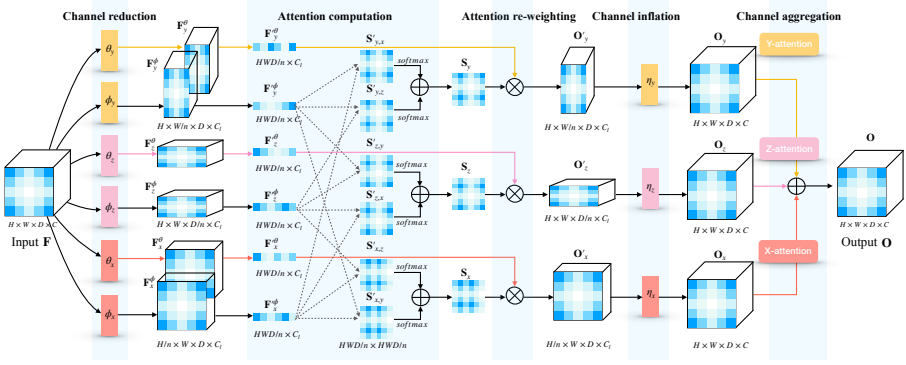

Fig. 3. Illustration of our 3D attention module. Given feature maps $\mathbf{F}$ as input and $\mathbf{O}$ as output, it consists of five sequential operations, i.e.channel reduction, attention computation, attention re-weighting, channel inflation, and attention aggregation.

and θ is for feature distillation, which are two parallel convolution layers with an anisotropic stride of n along a specific axis.

Without losing generality, we take x-axis as example. Denote the reduced feature maps by convolutional layer ϕ_x as $\mathbf{F}_x^{\phi}$ and those by convolutional layer θ_x as $\mathbf{F}_x^{\theta}$. Both feature maps have the size of $\mathbb{R}^{C_l \times \frac{H}{n} \times W \times D}$, where the fraction $\frac{1}{n}$ is attributed to the anisotropic stride of kernel along x-axis. For next correlation computation, $\mathbf{F}_x^{\phi}$ and $\mathbf{F}_x^{\theta}$ are reshaped into size of $\mathbf{F'}_x^{\phi}, \mathbf{F'}_x^{\theta} \in \mathbb{R}^{C_l \times \frac{1}{n} HWD}$. For y-axis and z-axis, we do same operation and finally get six feature maps of x, y and z-axis seperately for next cross-axis attention calculation.

Attention computation is conducted to calculate attention for x, y and z-axis individually to enhance the output feature maps. The attention matrix is calculated across every two feature maps of $\mathbf{F'}_x^{\phi}$, $\mathbf{F'}_y^{\phi}$ and $\mathbf{F'}_z^{\phi}$.

Take the correlation cross x and y-axis as example: we compute the correlation matrix between every pair of voxels[1] in feature maps $\mathbf{F'}_x^{\phi}$ and $\mathbf{F'}_y^{\phi}$:

$$\mathbf{S'}_{x,y} = \mathbf{F'}_x^{\phi^T} \mathbf{F'}_y^{\phi}, \in \mathbb{R}^{\frac{1}{n} HWD \times \frac{1}{n} HWD} \tag{1}$$

The voxel v_i at position i from feature maps of x-axis and voxel v_j at position j from feature map of y-axis is denoted as pair voxels (v_i, v_j), whose correlation is $s_{ij} = v_i v_j, v_i \in \mathbf{F'}_x^{\phi}, v_j \in \mathbf{F'}_y^{\phi}$. Softmax function is used for normalizing $\mathbf{S'}_{x,y}$ into the attention matrix $\mathbf{S}_{x,y} \in \mathbb{R}^{\frac{1}{n} HWD \times \frac{1}{n} HWD}$. Attention matrix $\mathbf{S'}_{x,z} = \mathbf{F'}_x^{\phi^T} \mathbf{F'}_z^{\phi}$ cross x-axis and z-axis is also calculated and normolized into $\mathbf{S}_{x,z}$ by softmax function. Then we have two cross-axis attention matrix $\mathbf{S}_{x,y}$ and $\mathbf{S}_{x,z}$, where $\mathbf{S}_{x,y}$ denotes the correlation matrix between $\mathbf{F'}_x^{\phi}$ and $\mathbf{F'}_y^{\phi}$, $\mathbf{S}_{x,z}$ denotes the correlation matrix between $\mathbf{F'}_x^{\phi}$ and $\mathbf{F'}_z^{\phi}$. By adding two feature maps along x-axis together, the final attention matrix on x-axis is shown below:

$$\mathbf{S}_x = \mathbf{S}_{x,y} + \mathbf{S}_{x,z}, \in \mathbb{R}^{\frac{1}{n} HWD \times \frac{1}{n} HWD} \tag{2}$$

[1] we use the term *voxel* v here after to refer to a general location in the feature map

Similarly with x-axis, the attention matrix $\mathbf{S}_y$ on y-axis is calculated by adding the cross-axis correlation matrix $\mathbf{S}_{y,z}$, $\mathbf{S}_{y,x}$, while the attention matrix $\mathbf{S}_z$ on z-axis is calculated by adding the cross-axis correlation matrix $\mathbf{S}_{z,x}$, $\mathbf{S}_{z,y}$.

Attention re-weighting is calculated by multiply the attention matrix to its corresponding feature maps for feature enhancing. We multiply the x-axis attention matrix $\mathbf{S}_x$ to feature maps $\mathbf{F'}^{\theta}_{x} \in \mathbb{R}^{\frac{1}{n}HWD \times C_l}$,

$$\mathbf{O}'_x = \mathbf{S}_x \mathbf{F'}^{\theta}_{x}, \in \mathbb{R}^{C_l \times H \times W/n \times D} \tag{3}$$

where $\mathbf{O}'_x$ is the attention re-weighted feature maps along x-axis. Same operations are done for y-axis and z-axis to get re-weighted attentions $\mathbf{O'}_y$ and $\mathbf{O'}_z$.

Channel inflation and Attention aggregation recover the size of feature maps as the same with input. To make the attention module readily usable, another convolution layer η_x is used to inflate $\mathbf{O}'_x$ to $\mathbf{O}_x \in \mathbb{R}^{C \times H \times W \times D}$ with same channel size of input $\mathbf{F}$. Thus, the attention process can be viewed as a bottleneck structure that attends to the distilled information. For y-axis and z-axis, we inflate and get $\mathbf{O}_y$ and $\mathbf{O}_z$.

The three inflated attention re-weighed features $\mathbf{O}_x$, $\mathbf{O}_y$, and $\mathbf{O}_z$ are then aggregated to form the output attention $\mathbf{O} \in \mathbb{R}^{C \times H \times W \times D}$, which is

$$\mathbf{O} = \mathbf{O}_x + \mathbf{O}_y + \mathbf{O}_z, \in \mathbb{R}^{C \times H \times W \times D} \tag{4}$$

The output attention contains correlation among every two positions from different directions, even with long-range distance. Our 3D attention module enhances the dependency among the whole feature maps in 3D spatial domain.

3.2 3D cross-axis Attention Enhanced Generative Framework

Our proposed framework is shown in Fig. 4, referred to as 3D cross-axis attention generative network, has four sub-networks: spatial-domain encoder $\mathcal{E}_s$, spectral-domain encoder $\mathcal{E}_f$, generator $\mathcal{G}$ and discriminator $\mathcal{D}$. The proposed 3D axis attention module $\mathcal{A}$ is integrated to all four sub-networks.

Spatial-spectral-aware Encoding sub-network

Our spatial-spectral-aware encoding network consists of two feature encoders, namely, a spatial-aware encoder ($\mathcal{E}_s$) to reproduce the visual appearance, and a spectral-aware encoder ($\mathcal{E}_f$) that maintains the statistics in implicit means. Both encoders consist of five 3D convolution blocks with the output channel sizes of $16, 32, 64, 128$ and 256, respectively, and two attention modules.

Fast Fourier Transform is used to convert the input volumetric training sample $\mathbf{x}$ to spectral domain, $\mathbf{x}_f = \mathcal{FFT}(\mathbf{x})$, both of which have the same size, $\mathbb{R}^{C \times H \times W \times D}$. We fuse the extracted spatial and spectral features right before sending them to the attention modules. The fusion is a simple weighted addition. The axis-aligned 3D attention modules are integrated into both encoders after the second and third convolution blocks, aiming to learn mid-level features. It captures the long-range feature dependency along with three axial directions of both the spatial and spectral domains. After feature extraction layers, the

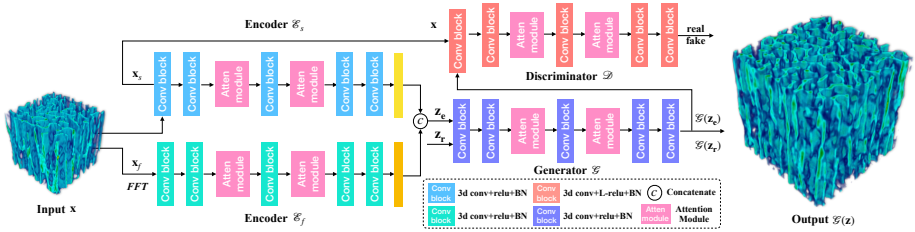

Fig. 4. The architecture of our proposed 3D-axisGAN network. During training, Generator $\mathcal{G}$ takes $\mathbf{z_e}$ and $\mathbf{z_r}$ iteratively as input, where $\mathbf{z}_e = \text{concat}(\mathcal{E}_s(\mathbf{x}), \mathcal{E}_f(\mathbf{x}_f))$ is the embedded vector from spatial encoder $\mathcal{E}_s$ and spectral encoder $\mathcal{E}_f$, $\mathbf{z_r}$ is a random gaussian noise. Discriminator $\mathcal{D}$ distinguishes real training samples $\mathbf{x}$ from fake synthesized results $\mathcal{G}(\mathbf{z_e})$ and $\mathcal{G}(\mathbf{z_r})$. After training, the Generator $\mathcal{G}$ takes a random gaussian noise $\mathbf{z_r}$ as input to synthesize the ouptup 3D micro-structure result $G(\mathbf{z_r})$.

features are flattened into vectors, which denoted as $\mathcal{E}_s(\mathbf{x})$ for spatial domain, $\mathcal{E}_f(\mathbf{x}_f)$ for spectral domain. The vectors are then concatenated to form the embedding vector $\mathbf{z_e} = \text{concat}(\mathcal{E}_s(\mathbf{x}), \mathcal{E}_f(\mathbf{x}_f))$ in the latent space of the 3D shape representation. In this way, the encoding network can extract both features from spatial and spectral domains to produce better results.

Attention-aware generator and discriminator sub-network

Generator $\mathcal{G}$ also consists of five 3D convolution blocks with the output channel size of $512, 256, 128, 64$, and 1, respectively, and two 3D attention modules after the second and third convolution layers. Given the latent feature vector $\mathbf{z_e}$ as input, we have the generated result $\tilde{\mathbf{x}} = \mathcal{G}(\mathbf{z_e}) \in \mathbb{R}^{H\times W\times D\times 1}$. The generator is also given gaussian noise $\mathbf{z_r}$ as input, which dimension is the same as $\mathbf{z_e}$. Generator $\mathcal{G}$ and discriminator $\mathcal{D}$ are trained alternately by solving a minimax problem defined as below:

$$\begin{aligned}\min_{\mathcal{G}}\max_{\mathcal{D}} \mathcal{L}_{adv}(\mathcal{D},\mathcal{G}) = {} & \mathbb{E}_{\mathbf{x}\sim p_{data}(\mathbf{x})}[log(\mathcal{D}(\mathbf{x})] \\ & + \mathbb{E}_{\mathbf{z_r}\sim p_{\mathbf{z}}(\mathbf{z})}[log(1-\mathcal{D}(\mathcal{G}(\mathbf{z_r})))] \\ & + \mathbb{E}_{\mathbf{z_e}\sim p_{data}(\mathbf{x})}[log(1-\mathcal{D}(\mathcal{G}(\mathbf{z_e})))]\end{aligned} \tag{5}$$

where $\mathbf{x}$ denotes training samples, $\mathbf{z_r}$ noise tensors randomly sampled from a gaussian distribution $p_{\mathbf{z}}$, and $\mathbf{z_e}$ the encoded latent vector corresponding to $\mathbf{x}$ as described above.

The performance of discriminator $\mathcal{D}$ is maximized so that it can correctly distinguish real samples $\mathbf{x}$ from synthesized results $\mathcal{G}(\mathbf{z_r})$ and $\mathcal{G}(\mathbf{z_e})$; on the other hand, generator $\mathcal{G}$ is trained to minimize $log(1-\mathcal{D}(\mathcal{G}(\mathbf{z_r})))$ and $log(1-\mathcal{D}(\mathcal{G}(\mathbf{z_e})))$ to produce indistinguishable synthesized results. Discriminator $\mathcal{D}$ is employed to distinguish real exemplars from the synthesized results, while the attention module is integrated with to improve the ability to discriminate the fake synthesized results.

3.3 Loss Function for Training

The adversarial training facilitates learning the data distribution $p_{data(\mathbf{x})}$ of the training samples. However, it is not enough to explicitly enforce either visual or statistical similarity between synthesized results and training samples. Thus, we incorporate three additional loss terms, i.e.the spatial similarity loss ($\mathcal{L}_s$), the spectral similarity loss ($\mathcal{L}_f$) and the regularized term $\mathcal{L}_{tv}$, into the adversarial training process. The loss function is defined as:

$$\mathcal{L}_{total} = \mathcal{L}_{adv} + \lambda_s \mathcal{L}_s + \lambda_f \mathcal{L}_f + \lambda_{tv} \mathcal{L}_{tv}, \tag{6}$$

where λ_s, λ_f and λ_{tv} are balancing weights.

The spatial similarity loss $\mathcal{L}_s$ is designed to ensure the visual appearance of the synthesized results to be similar to that of the training samples. The spectral similarity loss $\mathcal{L}_f$ serves a similar purpose, enforcing the similarity in the spectral domain, thus maintaining the statistical metrics to be close between synthesized results and training samples. The last term, total variational regularization loss $\mathcal{L}_{tv}$ contains the KL divergence loss [23] and the Diversity regularization [26]. The former is used to force the distribution of the latent vector similar to Gaussian distribution, which can be treated as a regularizer; the latter is to ensure that the spatial attention modules focus on various regions in the input feature maps instead of on solely the most discriminate region.

Spatial similarity loss for visual appearance The spatial similarity loss is defined as the sum of perceptual loss [15] at the feature level and per-voxel reconstruction error as follow,

$$\mathcal{L}_s = \sum_{l \in \{2,3,4\}} \sum_{i \in \mathcal{I}} \|\mathcal{D}_{i,l}(\tilde{\mathbf{x}}) - \mathcal{D}_{i,l}(\mathbf{x})\|^2 + \lambda_r \|\mathcal{G}(\tilde{\mathbf{x}}) - \mathbf{x}\|^2, \tag{7}$$

where $\mathcal{D}(\tilde{\mathbf{x}})$ and $\mathcal{D}(\mathbf{x})$ correspond to the feature maps of discriminator $\mathcal{D}$ of generated result $\tilde{\mathbf{x}}$ and ground truth sample $\mathbf{x}$, respectively; subscripts (i, l) in $\mathcal{D}_{i,l}$ indicate voxel i at l-th layer of the feature maps; $\mathcal{I}$ is the set of voxels at which sampled cubes are centered; and λ_r the balancing coefficient. The cubes (with size of $\mathbb{R}^{C \times H/c \times W/c \times D/c}$) are sampled in all three axial directions with a fixed stride for each layer in our case.

Spectral similarity loss for statistical metrics

Instead of explicit incorporation of multiple statistical metrics in our formulation, we devise an indirect means for such purpose by minimizing the difference between the training samples and the synthesized ones in the spectral domain. We write the spectral similarity loss as below,

$$\mathcal{L}_f = \|\mathcal{FFT}(\mathcal{G}(\mathbf{z_r})) - \mathbf{x}_f\|^2 \tag{8}$$

where $\mathcal{FFT}()$ denotes Fast Fourier Transform in spectral domain, $\mathbf{x}_f = \mathcal{FFT}(\mathbf{x})$. We do Fast Fourier Transform for the generated data $\mathcal{G}(\mathbf{z}_r)$ and the training samples $\mathbf{x}$ to make our results similar with examplar not only in the spatial domain but also in the spectral domain.

4 Experiments

We evaluate our methods on five datasets: *Bone* dataset (Femur, Trabecular, Sheep Spine); *Metal foam* dataset (Ni, Cu and Al); ICL dataset [30] (Bentheimer, Doddington, Estaillades and Ketton); NETL dataset [18] and ICP dataset [13]. All five datasets contains diverse shapes, each of which containing from $6k$ to $12k$ training samples. We evaluate our network compare with state-of-the-art methods and through ablation study to verify our specific designs.

4.1 Data and Setup

Data preparation Exemplars in *Bone* and *Metal foam* datasets are acquired from real-world materials via high-resolution micro-CT imaging system (SkyScan 1076). In *Bone* dataset, bone exemplars are cropped from scans of micro-CT, each of which consists of 7146 slices of size 3936×3936 at the resolution of $17\mu m$. The *Metal foam* dataset contains three categories of exemplars (i.e., Ni, Cu and Al) which also exhibit intricate internal structures. Each exemplar in this dataset consists of 1563 micro-CT slices of size 1952×1824 at the resolution of $8\mu m$. To constrain the computational overhead, the exemplars are cropped and resized from the scanned volumetric data while ensuring that substantial geometric details are preserved. Training samples $\mathbf{x}$ (64^3) are then uniformly cropped from the exemplar (100^3) with a stride of 2, forming a training set containing around $6k$ samples.

Implementation details We train our network for each exemplar separately. Each exemplar in the five datasets is a distinctive object and quite different from others in geometries and appearances. In each iteration, a batch of input training samples is randomly selected from the training set. ADAM is chosen as the optimizer with the settings of [34]. After training, the synthesized 3D micro-structure $\mathcal{G}(\mathbf{z_r})$ is generated by providing with generator $\mathcal{G}$ a random noise $\mathbf{z_r}$ sampled from the gaussian distribution. By changing the size of random noise, our generator could synthesize large results with arbitrary size within the memory of the graphic card in seconds.

4.2 Comparison with State-of-the-Art Methods

We evaluate our proposed framework with 3D axis attention module via qualitative and quantitative comparisons with the state-of-the-art methods: 1) 3D-GAN [42]; 2) VAE-GAN [23]; 3)SA-GAN [47]; 4) SolidTS [19]; and 5) 3DTS [48].

Qualitative evaluation Visual appearance comparison of 3D synthesized results by our method and the state-of-the-art approaches are shown in Fig. 5. The cross-sectional views with 10 consecutive slices randomly picked are rendered using ParaView [1].

For conventional example-based texture synthesis method, it could generate synthesized results locally similar with example by minimizing the difference between 2d patches in SolidTS [19] or 3D volume in 3DTS [48] sampled from the exemplar and the synthesized result based on Markov random fields (MRFs).

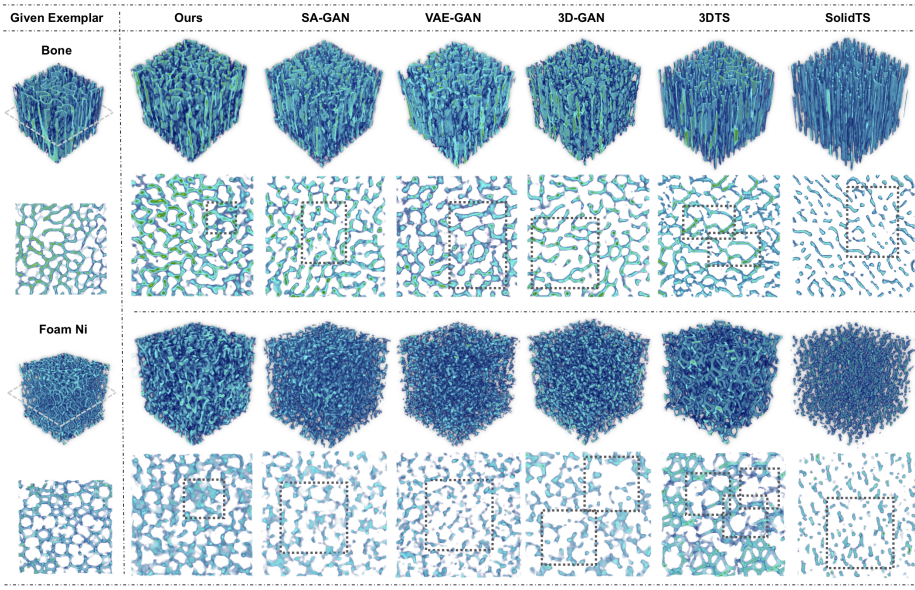

Fig. 5. Qualitative comparison of our synthesized results with those produced by state-of-the-art methods. 3D synthesized results of size 120^3 (larger than that of the exemplar) of *Bone* and *Metal foam Ni* are shown at 1st and 3rd rows, resp. Renders of 10 consecutive slices randomly selected from the corresponding 3D results are shown in 2nd and 4th rows, resp. Rectangles highlight failure parts, e.g., voids in 3D-GAN, repeated parts in 3DTS, and undesired intertwined structures in ours.

However, lacking global control, they may produce results with undesirable repeated patterns (for *Bone*) and disconnected branches (for *Metal foam Ni*), highlighted by the dashed rectangles in the cross-sectional views in Fig. 5, which fails to maintain the global connectivity or randomness shape as required.

For GAN based networks, synthesized results by 3D-GAN [42] and VAE-GAN [23] also suffer from the difficulty of reproducing similar structures with given exemplars. The cross-sectional views reveal more disconnected branches in the synthesized results of *Bone* and *Metal foam Ni*. Besides, large voids (highlighted by dashed rectangles) can be observed in the result of *Metal foam Ni* by 3D-GAN. Both networks focus on the general 3D shape reconstruction, which lacks modules to deal with the internal specific details, especially the complex connectivity and correlations of branches in 3D micro-structures. Thus they cannot well preserve the visual appearance and internal spatial coherence observed in the exemplar. We then extend SA-GAN [47] to 3D synthesis, the results show better connectivity than the previous two methods by adding attention to enhance feature correlations. However, the SA-GAN still has many floating disconnected branches, especially in *Metal Foam Ni*. This is because this attention module calculates correlation on every single individual position on feature maps, thus the value of two neighbor positions may be quite different, which is the disconnected part reflecting on the visual appearance. Our results are bet-

Table 1. Quantitative comparison with state-of-the-art methods on *Bone* and *Metal foam Ni* via metrics of FID, relative errors of porosity ($\varepsilon_{err}\%$), L_2 difference of two-point correlation functions ($D_2(\Phi)$) between the synthesized results and the exemplars, the number of connected components (N_{cc}) / floating parts (N_f) and the Wasserstein distance (D_w) for connectivity measure.

Evaluation Dataset	*Bone*					*Metal foam Ni*				
Methods	FID	$\varepsilon_{err}\%$	$D_2(\phi)$	N_{cc}/N_f	D_w	FID	$\varepsilon_{err}\%$	$D_2(\phi)$	N_{cc}/N_f	D_w
Ground Truth	N.A.	73.66	N.A.	6/4	N.A.	N.A.	88.23	N.A.	4/2	N.A.
SolidTS [19]	204.7	20.27	43.39	268/159	5.18	208.4	10.07	44.88	878/121	2.35
3DTS [48]	179.2	2.86	8.97	20/11	4.02	186.0	2.11	13.67	22/13	1.91
3D-GAN [42]	195.3	1.51	5.95	40/17	4.60	182.3	2.02	18.30	28/8	2.00
VAE-GAN [23]	187.0	1.48	5.76	37/21	4.71	171.4	1.89	17.46	30/10	2.02
SA-GAN [47]	181.5	1.11	5.27	17/9	3.89	163.9	1.25	17.04	20/8	2.03
Ours	**139.2**	**0.58**	**2.98**	**6/3**	**1.49**	**137.7**	**0.61**	**4.97**	**6/3**	**0.50**

ter than SA-GAN due to we have channel reduction with anisophic inflation and dilation on x, y and z-axis respectively, which brings more connectivity and neighborhood dependency through the different directional axis.

Through visual comparison with the above approaches, we have a higher degree of visual similarity to both *Bone* and *Metal foam Ni*. For *Metal foam Ni*, cellular structures are perceivable in our result, while others fail to reproduce such geometric configuration. In summary, our network can effectively enhance the visual quality of the synthesized results compared to those produced by state-of-the-art methods.

Quantitative evaluation We next quantitatively analyze the quality of the synthesized results and how well our network captures the statistics of exemplar.
FID as generation quality metric. We use Fréchet Inception Distance, or FID [12], as our metric for evaluating the generative results of different methods (including conventional methods). We modified the FID criterion developed in [36] to quantify our generated results.
Material characterization metrics. We adopt several statistical metrics widely used in materials sciences, i.e., Porosity ε [16] and Two-Point Correlation Function Φ [4]. Noted that, they are chosen for evaluation, not as loss terms is partly because they are only necessary conditions, but not sufficient, to produce 3D micro-structure with similar statistic properties. Simply adding multiple loss terms could lead to convergence problems during training. Additionally, direct optimization of these metrics can be tedious and may lead to conflict.

Porosity ε is a measure of the void volume fraction in a specific volume, which is defined as: $\varepsilon = |V_0|/|V|$, where $|V_0|$ and $|V|$ are the numbers of void and total voxels in the 3D model, respectively. The relative error between the exemplar (ε_E) and the synthesized (ε_S) is measured as $\varepsilon_{err} = |\varepsilon_E - \varepsilon_S|/\varepsilon_E$.

Two-Point Correlation Function $\Phi(r)$ measures the distribution of connected components (of length r) in a volume, which details refer to [31]. We evaluate the synthesized result by measuring the L2 difference of the Two-Point Correlation Function, $D_2(\Phi) = ||\Phi_S(r) - \Phi_E(r)||_2$, where $\Phi_S(r)$ and $\Phi_E(r)$ correspond to synthesized results and exemplars.
Connectivity metrics. Connectivity is a crucial property for synthesizing 3D micro-structure in real applications. We measure it using three metrics, i.e., the

number of connected components (N_{cc}), the number of floating parts (N_f), and a measure (D_w) based on the Wasserstein distance between two set of connected components, H_S and H_E, from two respective volumetric models, which is defined as $D_w = \mathrm{WD}(H_S, H_E)$. D_w thus reflects a similarity between two models in terms of how voxels are distributed and grouped into connected components.

Quantitative comparison of our network with the state-of-the-art methods are shown in Table 1, using different metrics on *Bone* and *Metal foam Ni* datesets. As seen from Table 1, the porosity metrics of our synthesized results are much closer to those of both exemplars (73.66% and 88.23% for *Bone* and *Ni*, respectively), whose relative error are 0.58%(*Bone*) and 0.61% (*Ni*). Both errors are half smaller than those (1.51%, 1.48% and 1.11% for *Bone*, 2.02%, 1.89% and 1.25% for *Ni*) produced by 3D-GAN, VAE-GAN and SA-GAN. The L_2 differences, denoted $D_2(\Phi)$, of two-point correlation functions of our synthesized results (2.98 for *bone*, 4.97 for *Ni*) are the smallest among all methods. In addition, our results have the minimal number of connected components $N_{cc} = 6$ and floating part $N_f = 3$ for both *Bone* and *Ni*, while the values of N_{cc}/N_f for *Bone* and *Ni* are (6/4) and (4/2), respectively. This shows that the connectivity of synthesized results is stable and quite similar to the exemplars.

4.3 Ablation study

We comprehensively evaluate our proposed 3D axis-attention module, framework and loss function on *Bone* and *Metal foam Ni* dataset using FID.

To evaluate the effectiveness of *design strategy for 3D axis-attentions*, we designed 9 experiments to compare the FID score with one axis, two axis and three axis attention module design shown in Table 2. $S_{x,x}$ denotes the module calculation between x-axis and x-axis, which is the same with self attention module. $S_{y,x}$ denotes the module with only one attention calculation across y-axis and x-axis. ($S_{z,x}$ $S_{x,z}$ $S_{x,y}$) denotes three axes attention, which are half of our designed six attention. ($S_{y,x}$ $S_{y,z}$ $S_{z,y}$ $S_{z,x}$ $S_{x,z}$ $S_{x,y}$) denotes our designed attention calculation across x, y and x-axis.

Table 2. FID score of synthesized results using attention module designed with one, two and three axis respectively.

	One axis			Two axes			Three axes		
	$S_{x,x}$	$S_{y,y}$	$S_{z,z}$	$S_{y,x}$	$S_{y,z}$	$S_{z,y}$	$S_{y,x}$ $S_{y,z}$ $S_{z,y}$	$S_{z,x}$ $S_{x,z}$ $S_{x,y}$	$S_{y,x}$ $S_{y,z}$ $S_{z,y}$ $S_{z,x}$ $S_{x,z}$ $S_{x,y}$
Bone	178.9	182.5	179.5	168.3	170.8	177.9	144.5	138.1	**139.2**
Foam Ni	162.2	165.1	166.5	156.3	159.7	152.8	142.7	143.9	**137.7**

As seen from Table 2, the FID score of our designed are less than that of using other combinations for both *Bone* and *Ni*, respectively. This is because of attention with anisotropic feature maps of x, y and z-axis captures more context features than a single one and two axes, which greatly enhances the capabilities to catch relationships among features.

To validate the effectiveness of our design choice of *the placement of the attention modules*, we conduct the following experiments. Table 3(left) shows the

FID score for different placements of our 3D attention module in the network. In Table 3, L2-3-4 denotes the module is inserted between the second, third and fourth convolution layers. We find that by inserting the 3D attention module after the second and third layers, the network produces the lowest FID score. This indicates that our attention module is more effective in response to the mid-level feature layers. This conforms to our initial guess, as the features in mid-level layers contain rich information with both 3D shape geometry and details feature.

To validate the chosen of our network architecture, we conduct experiments regarding *different network configurations* in Table 4. The configuration in Table 4(c), with additional spectral information, yields the significant performance improvement over the standard GAN in Table 4(c). In particular, the results in Table 4(h) outrun others Table 4(d,e,f) by adding the 3D attention module, where FID score sharply decreases from 178.6 and 164.1 to 139.2 and 137.7.

Table 3. Placement of our attention module among different convolution layers on FID score (left). And ablations study of the proposed training loss on FID score (right).

Config.	None	L2-3-4	L3-4-5	L1-2-3	L1-2	L2-3	L3-4
Bone	182.6	**139.2**	139.1	168.2	169.0	142.0	150.3
Foam Ni	168.1	**137.7**	145.5	151.7	154.6	141.8	147.2

Config.	$\mathcal{L}_{adv}$, $\mathcal{L}_{tv}$,$\mathcal{L}_s$	$\mathcal{L}_{adv}$, $\mathcal{L}_{tv}$,$\mathcal{L}_f$	$\mathcal{L}_{adv}$, $\mathcal{L}_{tv}$,$\mathcal{L}_f$,$\mathcal{L}_s$
Bone	144.5	145.1	**139.2**
Foam Ni	148.0	155.9	**137.7**

Table 4. Ablation study of our proposed attention module ^{att} integrated on the encoder $\mathcal{E}_s$, $\mathcal{E}_f$, generator $\mathcal{G}$ and discriminator $\mathcal{D}$ on FID score.

No.	Config.	*Bone*	*Foam Ni*	No.	Config.	*Bone*	*Foam Ni*
a	$\mathcal{G}$ +$\mathcal{D}$	195.3	182.3	e	$\mathcal{G}$ + $\mathcal{D}^{att}$ + $\mathcal{E}_s$ + $\mathcal{E}_f$	171.6	154.9
b	$\mathcal{G}$ + $\mathcal{D}$ +$\mathcal{E}_s$	187.0	171.4	f	$\mathcal{G}$ + $\mathcal{D}$ + $\mathcal{E}_s^{att}$ + $\mathcal{E}_f^{att}$	150.1	142.6
c	$\mathcal{G}$ + $\mathcal{D}$ +$\mathcal{E}_s$ +$\mathcal{E}_f$	178.6	164.1	g	$\mathcal{G}$ + $\mathcal{D}^{att}$ +$\mathcal{E}_s^{att}$ +$\mathcal{E}_f^{att}$	148.0	140.5
d	$\mathcal{G}^{att}$ + $\mathcal{D}$ + $\mathcal{E}_s$ +$\mathcal{E}_f$	167.2	155.4	h	$\mathcal{G}^{att}$ + $\mathcal{D}^{att}$ +$\mathcal{E}_s^{att}$ +$\mathcal{E}_f^{att}$	**139.2**	**137.7**

Finally, we use configuration in Table 4 (h) to evaluate the contribution of each loss term on FID score. As shown in Table 3 (right), the combination of spatial and spectral losses ($\mathcal{L}_{adv}$, $\mathcal{L}_{tv}$, $\mathcal{L}_f$, $\mathcal{L}_s$) can work synergistically to get far lower FID score than when they work individually, supporting our loss design.

4.4 More experiments and 3D printing results

To analyze the generalizablity of our method, we also present the results on synthesis of other datasets. ICL dataset [30], NETL dataset [18], ICP dataset [13] and *Metal foams Cu* and *Metal foams Al*, which all have highly complex internal micro-structures. For each exemplar, we randomly generate 10 different results and calculate the mean value for each statistical metric. The comparison on metrics between our methods and the state-of-the-arts are shown in Table 5 and Table 6. These experiments illustrate that our network can also produce the most convincing results in terms of statistical metrics.

To demonstrate the application of our synthesized 3D micro-structures, we do bool calculation on 3D model masks with the synthesized result, as shown in Fig.6(a) and (b). We also fabricate the digitally synthesized 3D micro-structure obtained by our network using a photosensitive resin 3D printer in Fig.6(c).

The 3D printed results retain rich fine-grained details and exhibit high visual similarity with the corresponding exemplars, implying that synthesized results by our method can be reliably fabricated while maintaining the design intent.

Table 5. Quantitative comparison with state-of-the-art methods on the ICL dataset.

Datasets	*Bentheimer*			*Doddington*			*Estaillades*			*Ketton*		
Metrics	FID_v	ε_{err}	$D_2(\Phi)$	FID_v	ε_{err}	$D_2(\Phi)$	FID_v	ε_{err}	$D_2(\Phi)$	FID_v	ε_{err}	$D_2(\Phi)$
SolidTS [19]	198.9	6.29	1.33	203.5	16.16	4.13	199.3	9.51	6.87	192.7	9.42	2.16
3DTS [48]	149.0	1.85	3.51	179.5	11.41	4.86	159.6	13.31	9.29	163.9	11.39	5.34
3D-GAN [42]	178.5	11.72	0.40	166.7	11.23	3.10	163.9	12.50	6.89	177.1	7.89	2.95
VAE-GAN [23]	159.2	4.67	0.41	143.2	5.87	3.05	155.6	4.49	2.84	167.8	5.09	2.18
SA-GAN [47]	141.9	2.29	0.45	138.4	3.91	2.24	153.5	3.53	2.52	152.6	4.05	1.99
Our approach	**125.3**	**1.81**	**0.37**	**131.1**	**3.20**	**1.21**	**145.3**	**1.63**	**1.96**	**139.3**	**0.64**	**1.94**

Table 6. Quantitative comparisons on NETL dataset, ICP dataset and *Metal Foam*.

Datasets	*Fontainebleau*			*Mt. Simon*			*Metal foam Cu*			*Metal foam Al*		
Metrics	FID_v	ε_{err}	$D_2(\Phi)$	FID_v	ε_{err}	$D_2(\Phi)$	FID_v	ε_{err}	$D_2(\Phi)$	FID_v	ε_{err}	$D_2(\Phi)$
SolidTS [19]	212.0	5.35	3.23	189.0	9.88	5.89	201.4	5.94	1.10	194.4	5.24	1.63
3DTS [48]	189.1	2.78	2.00	177.1	4.21	2.08	178.9	10.19	4.06	168.7	12.87	4.39
3D-GAN [42]	179.0	3.21	2.89	162.9	3.22	3.01	169.6	5.89	2.30	153.9	6.49	2.81
VAE-GAN [23]	181.0	3.33	4.02	167.0	4.94	3.86	143.0	5.51	1.82	155.0	4.80	2.05
SA-GAN [47]	166.2	3.14	2.35	163.0	3.29	2.33	141.4	5.24	1.20	143.9	3.11	1.41
Our approach	**162.4**	**1.89**	**1.09**	**155.9**	**2.08**	**0.89**	**133.1**	**3.87**	**0.49**	**128.6**	**1.98**	**0.54**

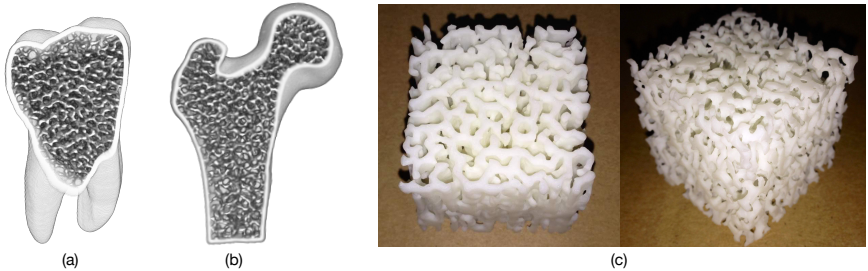

Fig. 6. Synthesized results by our network. (a) illustrates our synthesized 3D micro-structure used as scaffold for tooth implant; (b) illustrates our output used as supporting structures for bone replacement; (c) shows the synthesized results by 3D printing.

5 Conclusion

In this paper, we present 3D-axisGAN, a novel framework based on the generative network for the synthesis of 3D micro-structures, which is equipped with a novel 3D-axis attention module. We collect two datasets containing bone exemplars from different real bones and metal foams, all of which are acquired via high-resolution microCT scanning. Extensive experimental results show that our proposed 3D axis attention module, as well as the network design, are favorable to reproduce desired 3D complex micro-structures that visually, geometrically and statistically resemble the exemplars. Our method outperforms both conventional methods and GAN-based models. We will make our collected datasets publicly available to facilitate future studies in the relevant communities.

References

1. Ahrens, J., Geveci, B., Law, C.: Paraview: An end-user tool for large data visualization. The visualization handbook **717** (2005)
2. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473 (2014)
3. Bergmann, U., Jetchev, N., Vollgraf, R.: Learning texture manifolds with the periodic spatial gan. In: Proceedings of the 34-th International Conference on Machine Learning (2017)
4. Berryman, J.G., Blair, S.C.: Use of digital image analysis to estimate fluid permeability of porous materials: Application of two-point correlation functions. Journal of applied Physics **60**(6), 1930–1938 (1986)
5. Bucklen, B., Wettergreen, W., Yuksel, E., Liebschner, M.: Bone-derived cad library for assembly of scaffolds in computer-aided tissue engineering. Virtual and Physical Prototyping **3**(1), 13–23 (2008)
6. Chang, A.X., Funkhouser, T., Guibas, L., Hanrahan, P., Huang, Q., Li, Z., Savarese, S., Savva, M., Song, S., Su, H., et al.: Shapenet: An information-rich 3d model repository. arXiv preprint arXiv:1512.03012 (2015)
7. Chen, J., Wang, B.: High quality solid texture synthesis using position and index histogram matching. The Visual Computer **26**(4), 253–262 (2010)
8. Derby, B.: Printing and prototyping of tissues and scaffolds. Science **338**(6109), 921–926 (2012)
9. Fu, J., Zheng, H., Mei, T.: Look closer to see better: Recurrent attention convolutional neural network for fine-grained image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4438–4446 (2017)
10. Gatys, L., Ecker, A.S., Bethge, M.: Texture synthesis using convolutional neural networks. In: Advances in Neural Information Processing Systems. pp. 262–270 (2015)
11. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: Advances in neural information processing systems. pp. 2672–2680 (2014)
12. 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: Advances in Neural Information Processing Systems. pp. 6626–6637 (2017)
13. Hilfer, R., Lemmer, A.: Differential porosimetry and permeametry for random porous media. Physical Review E **92**(1), 013305 (2015)
14. Jetchev, N., Bergmann, U., Vollgraf, R.: Texture synthesis with spatial generative adversarial networks. arXiv preprint arXiv:1611.08207 (2016)
15. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: European Conference on Computer Vision. pp. 694–711. Springer (2016)
16. Karageorgiou, V., Kaplan, D.: Porosity of 3d biomaterial scaffolds and osteogenesis. Biomaterials **26**(27), 5474–5491 (2005)
17. Khan, Y., Yaszemski, M.J., Mikos, A.G., Laurencin, C.T.: Tissue engineering of bone: material and matrix considerations. JBJS **90**, 36–42 (2008)
18. Kohanpur, A.H., Valocchi, A., Crandall, D.: Micro-ct images of a heterogeneous mt. simon sandstone sample. http://www.digitalrocksportal.org/projects/247 (2019). https://doi.org/10.17612/1dvh-1n64
19. Kopf, J., Fu, C.W., Cohen-Or, D., Deussen, O., Lischinski, D., Wong, T.T.: Solid texture synthesis from 2d exemplars. ACM Transactions on Graphics (TOG) **26**(3), 2 (2007)

20. Ku, J., Mozifian, M., Lee, J., Harakeh, A., Waslander, S.L.: Joint 3d proposal generation and object detection from view aggregation. In: 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 1–8. IEEE (2018)
21. Kurenkov, A., Ji, J., Garg, A., Mehta, V., Gwak, J., Choy, C., Savarese, S.: Deformnet: Free-form deformation network for 3d shape reconstruction from a single image. In: 2018 IEEE Winter Conference on Applications of Computer Vision (WACV). pp. 858–866. IEEE (2018)
22. Kwatra, V., Essa, I., Bobick, A., Kwatra, N.: Texture optimization for example-based synthesis. ACM Transactions on Graphics (ToG) **24**(3), 795–802 (2005)
23. Larsen, A.B.L., Sønderby, S.K., Larochelle, H., Winther, O.: Autoencoding beyond pixels using a learned similarity metric. arXiv preprint arXiv:1512.09300 (2015)
24. Li, C., Wand, M.: Combining markov random fields and convolutional neural networks for image synthesis. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2479–2486 (2016)
25. Li, C., Wand, M.: Precomputed real-time texture synthesis with markovian generative adversarial networks. In: European Conference on Computer Vision. pp. 702–716 (2016)
26. Li, S., Bak, S., Carr, P., Wang, X.: Diversity regularized spatiotemporal attention for video-based person re-identification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 369–378 (2018)
27. Liu, X., Shapiro, V.: Random heterogeneous materials via texture synthesis. Computational Materials Science **99**, 177–189 (2015)
28. Ma, S., Fu, J., Wen Chen, C., Mei, T.: Da-gan: Instance-level image translation by deep attention generative adversarial networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5657–5666 (2018)
29. Mosser, L., Dubrule, O., Blunt, M.J.: Reconstruction of three-dimensional porous media using generative adversarial neural networks. Physical Review E **96**(4), 043309 (2017)
30. Muljadi, B.P., Blunt, M.J., Raeini, A.Q., Bijeljic, B.: The impact of porous media heterogeneity on non-darcy flow behaviour from pore-scale simulation. Advances in Water Resources **95**, 329–340 (2016)
31. Pant, L.M.: Stochastic characterization and reconstruction of porous media. Ph.D. thesis, University of Alberta (2016)
32. Park, D.Y., Lee, K.H.: Arbitrary style transfer with style-attentional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5880–5888 (2019)
33. Parmar, N., Vaswani, A., Uszkoreit, J., Kaiser, Ł., Shazeer, N., Ku, A., Tran, D.: Image transformer. arXiv preprint arXiv:1802.05751 (2018)
34. Radford, A., Metz, L., Chintala, S.: Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv preprint:1511.06434 (2015)
35. Sendik, O., Cohenor, D.: Deep correlations for texture synthesis. Acm Transactions on Graphics **36**(4), 1 (2017)
36. Shaham, T.R., Dekel, T., Michaeli, T.: Singan: Learning a generative model from a single natural image. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 4570–4580 (2019)
37. Smith, M., Flanagan, C., Kemppainen, J., Sack, J., Chung, H., Das, S., Hollister, S., Feinberg, S.: Computed tomography-based tissue-engineered scaffolds in craniomaxillofacial surgery. The International Journal of Medical Robotics and Computer Assisted Surgery **3**(3), 207–216 (2007)

38. Subramaniam, A., Nambiar, A., Mittal, A.: Co-segmentation inspired attention networks for video-based person re-identification. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 562–572 (2019)
39. Ulyanov, D., Lebedev, V., Vedaldi, A., Lempitsky, V.S.: Texture networks: Feed-forward synthesis of textures and stylized images. In: ICML. pp. 1349–1357 (2016)
40. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. In: Advances in neural information processing systems. pp. 5998–6008 (2017)
41. Wei, L.Y., Lefebvre, S., Kwatra, V., Turk, G.: State of the art in example-based texture synthesis. In: Eurographics 2009, State of the Art Report, EG-STAR. pp. 93–117. Eurographics Association (2009)
42. Wu, J., Zhang, C., Xue, T., Freeman, B., Tenenbaum, J.: Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling. In: Advances in Neural Information Processing Systems. pp. 82–90 (2016)
43. Wu, Z., Song, S., Khosla, A., Yu, F., Zhang, L., Tang, X., Xiao, J.: 3d shapenets: A deep representation for volumetric shapes. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1912–1920 (2015)
44. Xie, J., Zheng, Z., Gao, R., Wang, W., Zhu, S.C., Wu, Y.N.: Learning descriptor networks for 3d shape synthesis and analysis. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 8629–8638 (2018)
45. Xu, T., Zhang, P., Huang, Q., Zhang, H., Gan, Z., Huang, X., He, X.: Attngan: Fine-grained text to image generation with attentional generative adversarial networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1316–1324 (2018)
46. Yao, Y., Ren, J., Xie, X., Liu, W., Liu, Y.J., Wang, J.: Attention-aware multi-stroke style transfer. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1467–1475 (2019)
47. Zhang, H., Goodfellow, I., Metaxas, D., Odena, A.: Self-attention generative adversarial networks. arXiv preprint arXiv:1805.08318 (2018)
48. Zhang, H., Chen, W., Wang, B., Wang, W.: By example synthesis of three-dimensional porous materials. Computer Aided Geometric Design **52**, 285–296 (2017)