Improved $\alpha$-GAN architecture for generating 3D connected volumes with an application to radiosurgery treatment planning

Sanaz Mohammad Jafari 1 · Mucahit Cevik 1 · Ayse Basar 1

Accepted: 9 March 2023 / Published online: 9 May 2023 © The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2023

Abstract

Generative Adversarial Networks (GANs) have gained significant attention in several computer vision tasks for generating high-quality synthetic data. Various medical applications including diagnostic imaging and radiation therapy can benefit greatly from synthetic data generation due to data scarcity in the domain. However, medical image data is typically kept in 3D space, and generative models suffer from the curse of dimensionality issues in generating such synthetic data. In this paper, we investigate the potential of GANs for generating connected 3D volumes. We propose an improved version of 3D $\alpha$-GAN by incorporating various architectural enhancements. On a synthetic dataset of connected 3D spheres and ellipsoids, our model can generate fully connected 3D shapes with similar geometrical characteristics to that of training data. We also show that our 3D GAN model can successfully generate high-quality 3D tumor volumes and associated treatment specifications (e.g., isocenter locations). Similar moment invariants to the training data as well as fully connected 3D shapes confirm that improved 3D $\alpha$-GAN implicitly learns the training data distribution, and generates realistic-looking samples. The capability of improved 3D $\alpha$-GAN makes it a valuable source for generating synthetic medical image data that can help future research in this domain.

Keywords

Generative adversarial networks · 3D image data · Image synthesis · Connected components · Radiation therapy

1 Introduction

Over the past decade, Generative Adversarial Networks (GANs) have shown promising results in modeling a distribution from the available samples [1]. GANs have been frequently employed for image-to-image translation [2], image super-resolution [3] and specifically for synthesizing real-world images from random noise inputs [4–6]. GANs consist of two competing networks, namely, a generator that generates similar samples to the true distribution, and a discriminator that differentiates between the training data and generated samples. GANs’ goal is to generate novel and diverse samples which are representative of the real data distribution.

The majority of the recent works on GANs are typically confined to 2D datasets. Increasing datasets’ dimensionality significantly impacts the computational complexity of the generative models. However, the development of powerful computing architectures and decreasing costs of computational resources have made it possible to directly work with 3D images. Although moving from 2D to 3D images increases the computational burden, it provides a more detailed representation of the instances in the dataset. The impact of this transition is highlighted in the medical domain, where 3D images offer a more detailed view of human organs that are crucial to detect abnormalities [7].

We note that many recent studies on 3D GANs have focused on generating 3D objects from available 2D views. For instance, Chan et al. [8] and Or-El et al. [9] employed an encoder structure to extract features from the available 2D data and used the learned latent space for 3D image generation. On the other hand, training instability and mode collapse issues in GANs’ training procedure contribute to the lack of research on generating 3D images directly from random noise vectors.

GANs’ potential capabilities in generating high-quality 3D synthetic images have inspired various real-life applications, particularly in the medical domain, where the available data is limited due to privacy concerns, and publicly available data is rarely found. For instance, Han
et al. [10] used GANs to generate 3D lung nodules’ Computed Tomography (CT) samples for data augmentation and enhanced the performance of object detection. Another important application of 3D GANs is generating treatment plans as data samples for knowledge-based treatment planning [11]. Automation in radiation therapy treatment planning is highly desirable as the radiological images require image analysis and diagnosis by a trained human radiologist. That is, the high cost of training radiologists and the time-consuming procedure of manual diagnosis make machine learning models good candidates for assisting clinicians to reduce the workload. One specific use case for synthetic 3D image generation is radiosurgery treatment planning, where the objective is to find a set of isocenters (i.e., focus points for radiation beams) and associated radiation amounts so that tumor volumes are eradicated while delivering minimum radiation to surrounding healthy tissues. Berdyshev et al. [12] proposed ResNet models to learn from existing brain tumor data and previous treatment plans. The authors noted the need for larger datasets for a high-performance automated ML-based treatment planning. However, data privacy concerns prevent making such datasets publicly available, and synthetic data generation can offer a remedy for this issue.

Research goal We propose an improved version of 3D α-GAN architecture to generate synthetic 3D connected volumes. Although recent works have explored the generation of 3D objects, they typically fall short of addressing the connectivity requirements for the generated pixels. Our proposed model benefits from inception blocks inside the discriminator’s network among other modifications. The small and variant sizes of kernels in the inception block help the discriminator with identifying connectivity in data. Moreover, we evaluate our model on four synthetically generated datasets and extend the evaluation to synthetic data generation for radiosurgery treatment planning.

Contributions The main contribution of our paper is the design of an improved 3D α-GAN structure for the generation of 3D connected volumes. Specific contributions of our study can be summarized as follows:

- We modify the 3D α-GAN [13] architecture by introducing a new discriminator that incorporates the inception blocks. The diverse set of kernels in the inception block helps to extract fine details such as connectivity. As such, our proposed architecture constitutes a novel adaptation of 3D α-GAN.
- We focus on the connectivity considerations between an object’s pixels and we design a set of evaluation metrics to assess the connected components in 3D space. To the best of our knowledge, this is the first study to investigate connectivity considerations in 3D image generation.
- We propose an extensive set of evaluation metrics to estimate the shape and proper distribution of 3D volumes and the artifacts within. We also conduct a thorough numerical study to evaluate different GAN variants’ performance.
- We apply 3D GANs to a novel practical problem of data augmentation for radiosurgery treatment planning. Our results offer new insights into 3D object generation using voxelized representations of tumor volumes. Accordingly, our analysis provides evidence for the potential of GANs in synthetic data generation for radiosurgery treatment planning.

Organization of the paper The rest of this paper is organized as follows. Section 2 summarizes the recent developments in 3D object generation using GANs. Section 3 briefly describes the generic Variational Auto-encoder (VAE) and GAN structures and explains the proposed architecture in detail. Section 4 provides a comprehensive list of our proposed evaluation metrics along with a discussion on the model performance for four different datasets. Section 5 provides a discussion on the study findings from our detailed numerical study. Finally, Section 6 concludes the paper with a summary of research outcomes and a discussion of potential future research directions.

2 Background

In this section, we review recent studies on 3D object synthesis and discuss various GAN architectures that are used to generating novel, diverse 3D shapes. Moreover, we explore the applications of 3D object generation in radiosurgery treatment planning. A summary of the most closely related studies on 3D GANs is reported in Table 1, which contains information on the proposed methodologies and employed datasets.

3D object generation and detection are one of the most important research areas in computer vision [23, 24]. Three-dimensional images are typically represented in four formats, including point cloud, voxel grid, triangle mesh, and multi-view representation. Voxels are 3D versions of image pixels and can be interpreted as quantized, fixed-sized point clouds. On the other hand, point clouds have an infinite number of points in space with float pixel coordinates. Voxel grids suffer from large memory consumption and low-resolution representation. However, in certain problems (e.g., radiotherapy) the limited size of 3D shapes and lack of fine details make the 3D voxel grid a
Our study improved a 3D α-GAN with an inception-layered discriminator. This work led to a large number of studies that focused on reconstructing a 3D object from its 2D slices, leveraging the available information in 2D representation [31, 32]. However, using 2D representation limits the generation of novel and diverse samples as it confines the latent space of images. Hong et al. [21] proposed an extended version of α-GAN plus WGAN-GP [33], called 3D-StyleGAN, to generate 3D MNI scans of brain. Their architecture has a progressive structure, where each part receives a version of the mapped latent noise vector and style vector and passes them through 3D convolutions. The model generates high-quality 3D brain scans that can be manipulated using the style vectors. Similarly, our study proposed M-GAN by adapting the StyleGAN [34] to generate 3D shape volumes in Polycrystals. They noted that, although their model generates reasonable small 3D brain volumes, grain connectivity relationships are not guaranteed and should be investigated. Following the same procedure, Kwon et al. [13] suggested an adaptation of α-GAN [35] for 3D space by adding the Wasserstein GAN with Gradient Penalty (WGAN-GP) [36] loss functions. The proposed model uses a combination of GANs and VAEs to address the mode collapse and image blurriness.

3D object generation is an important problem in the medical domain as the majority of available datasets are 3D and the images typically have significantly more details. The computational complexity and memory issues commonly led to using an alternative approach by extracting 2D slices of the 3D images to address this issue, causing a loss of information and connectivity between the slices [11, 37]. The GANs’ capabilities in synthesizing 3D images have increased their potential in medical applications, such as generating more diverse and unseen 3D brain MRI data to address clinically difficult tasks [13, 21, 38]. Another significant application of GANs in the medical domain is generating treatment planning schemes to help with knowledge-based planning and treatment planning automation [12, 39].

Table 1 Summary of the relevant papers on 3D object generation

| Study          | Model             | Methodology                                      | Datasets                                                                 |
|----------------|-------------------|-------------------------------------------------|--------------------------------------------------------------------------|
| Wu et al. [14] | 3D-GAN, 3D-VAE-GAN | Convolutional GAN combined with VAE              | ModelNet, Ikea dataset                                                  |
| Choy et al. [15]| 3D-R2N2          | Learned mapping from 2D images to generate 3D objects | ModelNet, PASCAL VOC [16], Online Products [17] |
| Kwon et al. [13]| 3D α-GAN         | Adaptation of α-GAN plus WGAN-GP                | ADNI [18], BRATS 2018 [19]                                              |
| Babier et al. [20]| 3D GAN          | 3D pix-to-pix structure                         | Clinical radiation therapy plans for 217 patients with oropharyngeal cancer |
| Hong et al. [21]| 3D-StyleGAN      | Mapped latent noise with style vector            | Brain MR T1 images                                                      |
| Jangid et al. [22]| M-GAN           | Adaptation of StyleGAN                          | ModelNet, 3D grain volumes                                             |
| Our study      | Improved 3D α-GAN | Adaptation of 3D α-GAN with inception-layered discriminator | Synthetic connected 3D sphere/ellipsoid volumes with/without packed spheres, Synthetic connected 3D tumor volumes with/without packed spheres |

suitable representation for coarse objects.

There has been a noticeable increase in 3D object modeling and synthesis in recent years, with several studies focusing on incorporating the existing objects’ parts in CAD model libraries. These approaches generate realistic-looking samples with low novelty and diversity. Developments in deep learning and the introduction of large 3D CAD datasets such as ModelNet [25] have attracted significant attention toward learning deep representations from voxelized objects. It is important to note that using deep learning models on 3D objects is more difficult than on 2D objects because of the curse of dimensionality. Wu et al. [25] were the first to represent 3D shapes as probabilistic distribution of binary voxels, where ones represent data and zeros represent empty space. They introduced a Convolutional Deep Belief Network (CDBN) called ShapeNet, which is the first deep learning model to be trained on 3D voxel images. Drawing on this study, several methods were proposed to incorporate deep learning models in 3D image-related tasks such as 3D object recognition [26], 3D object representation and retrieval using 2D views [27–29] and 3D image reconstruction from noisy data [30].

The introduction of adversarial loss to the GAN models as well as the discriminator’s ability in object recognition inspired the incorporation of GANs for synthetic 3D object generation. Wu et al. [14] were the first to propose 3D-GAN and 3D-VAE-GAN for synthesizing diverse and high-quality 3D objects. Their trained discriminator outperformed the state-of-the-art models for image classification over ModelNet dataset by a margin of 15% in terms of accuracy. However, the authors did not offer any evaluation metrics for the quality of generated samples. Following this work, Choy et al. [15] proposed a 3D Recurrent Reconstruction Neural Network (3D-R2N2) to learn a mapping from multi-view 2D images to their 3D shapes.
Babier et al. [20] proposed a 3D GAN image-to-image translation structure adapted from Pix2pix [40] to solve a dose distribution problem. Their model receives CT images and predicts the full 3D-dose distribution. Their approach increases the satisfied clinical criteria by 11% compared to other clinical approaches. Additionally, their method outperforms a similar 2D GAN structure applied on 2D slices of the image [11], confirming that the correlations between 2D slices offer useful information. Berdyshev et al. [12] specifically focused on predicting the locations of isocenters inside the tumor volumes, which is an important step in radiosurgery treatment planning. They proposed a dimensionality reduction technique for the 3D tumor shapes to alleviate the curse of dimensionality and employed a residual neural network for the isocenter prediction task. Although they showed comparable results to the ground truth in their limited experimental results, their approach lead to the loss of voxel connectivity by moving to 2D space.

These studies inspire the investigation of the capabilities of 3D GANs in radiosurgery treatment planning. Our proposed architecture is an adapted version of 3D $\alpha$-GAN to generate connected 3D volumes. Our model differs from 3D $\alpha$-GAN [13] as we adopt an inception structure for our discriminator network, and incorporate a specifically designed loss function to ensure connectivity.

3 Methodology

In this section, we first review the basics of GAN and VAE and explain the improved 3D $\alpha$-GAN architecture in detail. Then, we discuss the specifications of four synthetic datasets used in our analysis and provide the details of our experimental setup. A notation table is provided in Appendix to describe the notation used in mathematical formulas.

3.1 Review of GAN models

A generic GAN architecture contains two networks known as the generator and the discriminator, both competing against each other (see Fig. 1). During the training process, the generator learns the distribution $p_G$ from an input noise $z$ mapped through the function $G(z; \theta_g)$ to the samples. Here, $G(z; \theta_g)$ is a differentiable function, usually defined as a neural network with a set of parameters $\theta_g$. The discriminator $D(x, \theta_d)$, which is also a differentiable function with parameters $\theta_d$, maps input samples to a probability value indicating whether the sample is real or generated. Both networks are trained simultaneously, where the discriminator and generator have maximization and minimization objectives, respectively.

Therefore, GAN is modeled as a min-max game with a value function $V(D, G)$, that is,

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

The discriminator aims to maximize the value function by generating a probability value of one for the samples from the real data, $p_{data}$, and zero for the samples from the generated distribution, $p_G$. On the other hand, generator minimizes the objective value by generating real-looking samples that the discriminator fails to detect as fake.

3.2 Variational auto-encoders

VAE is a generative model with a similar structure as a regular auto-encoder that uses variational inference for training and creates a distribution for each attribute of the latent space instead of a single value. VAE consists of two networks, namely, encoder and decoder, as illustrated in Fig. 2. The encoder maps the input $x$ to a continuous smooth latent space $z$ and decoder samples from the latent space distribution to generate diverse samples at the output. The trained decoder can be used as a generator to create diverse samples from the latent space distribution. Since the random sampling process cannot generate gradients during the backpropagation and model’s training process, a re-parameterization trick was proposed to address this issue [41]. We assume that the prior distribution $p(z)$ follows a Gaussian distribution, thus the latent space presents two vectors that contain the mean and standard deviation of the encoded distribution.

Fig. 1 Vanilla GAN structure that consists of two competing networks, namely, discriminator and generator.

![Vanilla GAN structure](image-url)
The VAE’s loss function is shown in (2), which consists of a reconstruction error between the input and generated image and a regularization term for latent space samples $j$ that follows a Gaussian distribution.

$$L = L(x, \hat{x}) + \sum_j [KL(q_j(z|x)\|p(z))]$$  \hspace{1cm} (2)

Since VAEs use the encoded space of real samples, unlike GANs, they do not suffer from mode collapse.

### 3.3 3D $\alpha$-GAN architecture

We leverage the 3D $\alpha$-GAN structure proposed by Kwon et al. [13], which is an adaptation of $\alpha$-GAN architecture [35]. $\alpha$-GAN is a combination of VAE and GAN to solve the mode collapse issue in GANs and overcome the blurry images generated by VAEs. It consists of four networks, namely, generator $G(z; \theta_g)$, discriminator $D(x; \theta_d)$, encoder $E(x; \theta_e)$ and code discriminator $D_c(x; \theta_c)$. Code discriminator network replaces the variational inference in VAE. The encoder receives the real data samples and encodes them to a latent vector $z_e$. The encoded vector $z_e$, as well as the noise vector $z_r$, are both passed to the generator and code discriminator. The generator in this approach generates two sets of fake data from encoded noise vector $z_e$ and predefined noise vector $z_r$ and passes them to the discriminator. The discriminator receives two generated samples and one real sample and differentiates between fake and real samples. On the other hand, the code discriminator distinguishes between the real noise vector $z_r$ and the encoded one $z_e$. The code discriminator encourages the encoder to fully encode the real distribution to $z_e$. When the code discriminator fails to differentiate between fake and real noise, it shows that the generator has covered the entire decoded space, and the network has successfully prevented the mode collapse. Moreover, when the discriminator fails to recognize the fake from real samples, the network is fully trained, and the generated samples are representative of the training distribution. Figure 3 illustrates the model architecture and networks’ structures.

Kwon et al. [13] showed that replacing the basic GAN’s loss function using Wasserstein loss with gradient penalty
Improved α-GAN architecture for generating 3D connected volumes

[36] stabilizes the model’s training process. Equation (3) presents the discriminator’s loss function of a general GAN plus the gradient penalty factor formulated as $L_{GPD} = E_z[1(||\nabla_{\theta}D(\hat{x})||_2 - 1)^2]$. 

$$L_D = E_{\varepsilon}[D(G(\varepsilon))] + E_{\varepsilon}[D(G(z))] - 2E_{xreal}[D(xreal)] + \lambda_1L_{GPD}$$

(3)

Loss functions of generator and encoder are equivalent, and represented in (4), where the reconstruction loss is added using the $L_1$ distance between the real and generated image, that is,

$$L_G = -E_{\varepsilon}[D(G(\varepsilon))] - E_{\varepsilon}[D(G(z))] + \lambda_2||x_{real} - G(z)||_1$$

(4)

The gradient penalty of code discriminator is defined by $L_{GPD}$ and formulated as $E_z[1(||\nabla_{\theta}D(\hat{x})||_2 - 1)^2]$. We can obtain the code discriminator loss and its gradient penalty term as

$$L_C = E_{\varepsilon}[D_c(\varepsilon)] - E_{\varepsilon}[D_c(z)] + \lambda_1L_{GPD}$$

(5)

Note that $\lambda_1$ and $\lambda_2$ are predefined hyperparameters, which we set to 10 in our implementations.

### 3.4 3D α-GAN with connection loss

The existing α-GAN architecture does not provide any insights regarding the connection of elements in 3D objects. We make a small alteration in the generator’s loss function that can significantly improve the model’s performance in generating connected components. Let the number of connected components in an object be $CC$, which should ideally be equal to one. We define the connection loss as

$$L_{\text{connect}} = \begin{cases} 
\frac{1}{N}\sum_{i=1}^{N}((CC - 1) - \lambda_3[CC = 1]) \\
\frac{1}{N}\sum_{i=1}^{N}\left(\sqrt{\frac{1}{K}\sum_{i=1}^{K}(CC[i] - 1)^2} - \lambda_3\sum_{i=1}^{K}[CC[i] = 1]\right) 
\end{cases}$$

#connected objects = 1

#connected objects $\geq$ 2

(6)

Equation (6) shows an additional penalty term, which is added to the generator’s loss function to encourage the generation of more connected elements. For samples with one connected 3D object, loss for $N$ samples is calculated over the number of connected elements. On the other hand, when the shape includes $K$ connected elements, the root mean square error between one and the number of connected components is calculated and added to a hyperparameter $\lambda_3$ multiplied by the number of connected components that are equal to one. Here, we set the initial value for $\lambda_3$ as one, but the hyperparameter tuning process could provide a better alternative value. The first element of the loss penalizes the generator for disconnected components, whereas the second part of the loss provides a large reward for outputting fully connected components. Note that the second term is included to not overwhelm the generator at the initial training epochs. Samples of connected components are illustrated in Fig. 4 to provide a more detailed explanation on connection loss computation. For Fig. 4a with one connected object and two samples, the connection loss is computed as $\frac{1}{2}[(2 - 1) - 1 \times 0] + (1 - 1 - 1 \times 1)]$. On the other hand, for Fig. 4b with two connected objects and two samples, the connection loss is estimated as $\frac{1}{2}[(\sqrt{\frac{1}{2}[(1 - 1)^2 + (1 - 1)^2]} - 1 \times 2) + (\sqrt{\frac{1}{2}[(1 - 1)^2 + (2 - 1)^2]} - 1 \times 1)]$.

### 3.5 Improved 3D α-GAN architecture

We design an improved version of 3D α-GAN by altering the discriminator network’s architecture. Specifically, the new discriminator architecture illustrated in Fig. 5 incorporates an inception network structure [42]. The inception block shown in Fig. 5b, consists of four parallel convolutional blocks that affect the input. The output of convolutions are merged through channel dimension and passed to
Improved 3D $\alpha$-GAN structure consists of four networks. The discriminator structure includes inception blocks in the next layers. Three of convolutional blocks include $1 \times 1$ kernels, which reduces the dimensionality of the channels.

Inception networks are designed to approximate and cover sparse local regions in a convolutional network. Therefore, it can help the network detect and extract connectivity features from the training data. The discriminator’s capability in recognizing connectivity helps with the task of differentiation between real and fake data, and thus, penalizes the generated samples with low connectivity. Additionally, other networks’ architectures are modified to generate the specific dimensions of our datasets (see Appendix). The modified GAN structures are illustrated in Fig. 5a.

### 3.6 Datasets

We consider four distinct datasets in our numerical study, which are described below.

- **Synthetic connected 3D volumes:** This is a dataset of size 40,000 consisting of random spheres and ellipsoids with radius $(r_1, r_2, r_3)$ that are created in a $[16, 16, 16]$ 3D space. Voxel and mesh representation of a sample from the training set is illustrated in Fig. 6.
- **Synthetic connected 3D volumes with packed spheres:** Extending the previous dataset, a fixed number of smaller spheres are predefined inside each 3D volume.
However, since the sparse distribution of inside volumes can impact the learning performance, a denser version of the dataset is proposed in this part. That is, the dataset from part one is modified, and, for seven predefined points that are distributed inside the volume, a random sphere with a radius of $r/4$ is created to cover a specific space in the generated original sample. We name these predefined points that are the center of subspheres as isocenters. A sample of the dataset is illustrated in Fig. 7. Note that smaller spheres inside the main volume correspond to the isocenter locations in 3D tumor volume datasets. That is, the center of these smaller spheres can be treated as an individual isocenter. We also note that representing an isocenter as a smaller volume inside the main 3D volume is intuitive since the isocenters are the focal points of radiation delivery, and radiotherapy machines deliver dose focusing on this volume to eradicate tumor voxels in the vicinity of the isocenter, often forming a spherical or ellipsoidal area of impact [43]. A sample of a 3D tumor that is located between two healthy tissues or organs at risk (OARs) is illustrated in Fig. 8. The six isocenter locations that are defined inside the tumor represent the focus points of transmitted radiations.

- **Synthetic connected tumor volumes**: We generate a sample of connected volumes following the real distribution of 14 radiosurgery cases used in [39]. Each voxel has a volume of $2 \text{ mm}^3$, which in the space of $[16, 16, 16]$ represents the largest created tumor of size $12 \text{ cm}^3$. A sample of the dataset is illustrated in Fig. 9.
- **Synthetic connected tumor volumes with packed spheres**: We use the synthetic tumor volume dataset as the basis of this dataset, and employ a hybrid of grassfire and sphere-packing algorithms proposed by [43] to identify the center voxels of packed spheres.
spheres (i.e., isocenters) inside the tumor volumes. We then return their covered volume’s radius in the target tumors. A sphere-filling approach is used to surround each point with its covered radius. A sample of the dataset is illustrated in Fig. 10.

### 3.7 Experimental setup

We evaluate our proposed 3D GAN architecture on four different datasets mentioned in Section 3.6. We experiment with 3D convolutional GAN, WGAN-GP, and $\alpha$-GAN. PacGAN [44] approach is used in GAN and WGAN-GP architectures to avoid mode collapse. Table 2 presents a summary table of the $\alpha$-GAN architecture [13]. Moreover, a detailed version of the proposed improved $\alpha$-GAN structure is reported in Appendix.

Each model is trained on 40,000 training samples, and the evaluation metrics are reported over 10,000 generated samples by the generator network. For all experiments, Adam optimizer with a learning rate of $2e^{-4}$ is used. Models are trained for 500 epochs, and each experiment is reported five times to minimize the effect of random initialization. Since 3D space causes computational complications in our experiments, to avoid memory issues, a batch size of 40 is used. It is worth mentioning that the simple structure of our datasets makes it possible to use less complex CNN structures. All the models were implemented using Python and PyTorch library, and experiments were performed on NVIDIA GeForce RTX 2070 GPU with 8 GB of GPU RAM.

### 4 Numerical study

In this section, we first present the performance metrics used to evaluate GAN architectures. Then, we compare the performances of two baseline models including GAN and WGAN-GP with three different $\alpha$-GAN models over
synthetic connected 3D volume datasets. Next, we choose the two best-performing models and resume the numerical comparison with connected tumor volume datasets where the irregularities of tumor shapes compared to fully convex sphere/ellipsoid introduce a new challenge for the GAN models.

Furthermore, we evaluate the models’ performance on connected volumes packed with subspheres. For the

| Network   | Layer | # of channels | Kernel size | Activation function | Batch normalization |
|-----------|-------|---------------|-------------|---------------------|---------------------|
| Generator | Conv  | 16            | 4           | ReLU                | True                |
|           | Upsample | –             | –           | –                   | False               |
|           | Conv  | 8             | 4           | ReLU                | True                |
|           | Upsample | –             | –           | –                   | False               |
|           | Conv  | 1             | 4           | Sigmoid             | False               |
| Discriminator | Conv  | 4             | 4           | LeakyReLU           | False               |
|           | Conv  | 8             | 4           | LeakyReLU           | True                |
|           | Conv  | 16            | 4           | LeakyReLU           | True                |
|           | Conv  | 1             | 4           | LeakyReLU           | False               |
| Code Discriminator | Linear | 4096       | –           | LeakyReLU           | True                |
|           | Linear | 4096         | –           | LeakyReLU           | True                |
|           | Linear | 1            | –           | –                   | False               |
synthetic sphere/ellipsoids, we have seven predefined filled spheres located at isocenters’ specific coordinates. On the other hand, the tumor data consists of a diverse number of subspheres with different coverage radiuses located at random coordinates. Therefore, the generation of synthetic connected volumes with packed spheres for sphere/ellipsoid can be deemed as a less challenging task compared to the tumor data.

### 4.1 Performance metrics

Most popular evaluation metrics for GAN architectures are designed to assess image quality and diversity, as the majority of previous studies are focused on generating high-resolution detailed data such as face samples [45]. However, most of these metrics are not suitable for an effective evaluation of our methodology, because of our datasets’ coarse details and the importance of pixel connectivity. Therefore, we propose novel metrics to test the generated samples. Proposed evaluation metrics can be divided into two categories, namely, shape metrics and location metrics. The shape metrics are used to evaluate the connectivity, convexity, and overall shape of the generated sample, whereas the location metrics are designed to evaluate the distribution of points and their relevant distances. More specifically, the location metrics are solely designed to evaluate the distribution of isocenters inside a connected volume. More details about these performance metrics including their range and desired values are provided in Appendix.

We note that both connected volume datasets have similar characteristics, and thus are evaluated using the same shape metrics. On the other hand, the connected volume datasets with packed spheres have differently designed sets of isocenters as the center of subspheres. The sphere/ellipsoid-shaped connected volumes have seven predefined isocenters, located at the center and in the middle of the main axis. We can extract the seven mentioned locations from the generated shapes, and compare their relative distances. On the other hand, tumor volumes’ isocenters are located using the grassfire algorithm and can vary from 4 to 20 points. Therefore, we exclude these metrics for the tumor volumes and only report the ratio of space that the generated subspheres cover inside the main volume.

Since there is no well-defined ideal value for the majority of our metrics, we report the KL divergence between the distributions of a metric for the training data vs the generated samples. We expect to get small KL divergence values to indicate the similarity of the two distributions, and how closely the generator followed the training data. Performance metrics of our study related to shape evaluation are summarized as follows:

- **Connected volumes’ sizes**: The size of the connected volumes presents the number of connected voxels that create the connected volume shape. The generated volumes’ sizes distribution should ideally match the real data distribution. Therefore, we have another evaluation metric, where we report the KL divergence between the two distributions.

- **Convexity ratio**: We process the generated samples and find the best set of generated voxels that can generate a convex object. The number of generated voxels that are inside the convex shape divided by the total number of generated voxels defines the convexity ratio. We note that only the sphere/ellipsoid connected volumes are fully convex, whereas the connected tumor volumes have a variety of shapes that can be convex or non-convex. Therefore, the convexity ratio for the former group is one, and for the latter group, it can range from 70% to 100%. Therefore, we solely report the KL divergence between the distribution of the convexity ratios.

- **Connectivity ratio**: Techniques on 3D shape generation typically involve post-processing the generated shapes and selecting the heavily clustered voxels as the final output. For this metric, we post-process the generated shapes by setting the average of points as the center and removing the points that have outlier distances to the center point. The ratio of the number of remaining

---

**Fig. 11** 2D shapes of generated samples and relative distances
Table 3 Summary performance values for the synthetic connected 3D volume dataset evaluated for 10,000 samples averaged over 5 repeats

| GAN model | Mode collapse | Connected volumes' sizes | Coverage ratio | Convexity ratio | KL divergence |
|-----------|---------------|--------------------------|---------------|----------------|--------------|
| GAN       |               | 579 ± 408               | 0.65          | 0.97 ± 0.608   | 4.82 ± 4.089 |
| WGAN-GP   | 500 ± 346     | 0.92                     | 1.16 ± 0.689  | 7.08 ± 5.062   | 34.001 ± 32.889 |
| α-GAN     | 681 ± 529     | 0.98                     | 1.16 ± 0.764  | 7.49 ± 5.636   | 59.273 ± 46.137 |
| α-GAN-CL† | 709 ± 539     | 0.98                     | 1.20 ± 0.746  | 7.69 ± 5.650   | 66.830 ± 51.385 |
| α-GAN++‡  | 748 ± 463     | 0.93                     | 1.23 ± 0.702  | 7.96 ± 5.216   | 70.799 ± 49.248 |

Coverage ratio: We define certain thresholds for connectivity and convexity ratios by investigating the training data ratios. We then report the ratio of generated samples that meet the pre-defined thresholds of convexity and connectivity ratio as the coverage ratio. Ideally, all the generated shapes should be completely connected and the coverage ratio should be equal to one.

3D moment invariants: Moment invariants (Ω₁, Ω₂, Ω₃) are spatial descriptors used to quantify the distribution of an object’s solid volume [46]. We use the distribution of 3D moments to evaluate the shape quality of the generated samples.

Shannon equitability index: We use this index defined in (7) to measure the similarity (i.e., evenness) of different isocenters. For the case of sphere/ellipsoid connected volumes, since we designed the subspheres to be approximately similar, we can expect this measure to be near 1. However, the shape of subspheres are typically quite different in connected tumor volumes, and thus, we report the KL divergence of the Shannon index, that is,

\[ E_H = \frac{H}{\log(K)} \]

Subspheres’ coverage: We measure the ratio of voxels that are covered by the packed spheres to the ratio of the entire connected volume. Ideally, we want the subspheres to cover the entire volume, and this coverage to be near 1.

Connected subspheres per connected volume: We discussed the importance of the coverage ratio for volumes, and how it represented a connected object. For this measurement, since we defined each subsphere to be connected, we calculate the ratio of subspheres per volume that satisfy the coverage ratio. This measurement shows how introducing diversity in our images affects the model performance in generating multiple connected shapes.

The following performance metrics are designed to evaluate the location and distribution of subspheres and isocenters. These are only applicable to the sphere/ellipsoid connected volumes, where the ideal isocenter locations are known and can be identified on a generated sample. Proposed performance metrics related to distance evaluation are as follows:

FD error: Discrete Fréchet distance between the ground truth location of isocenters (processed on the output) and generated isocenters.
Fig. 12 Moment invariant of the 3D connected volume training data (blue line represents the mean value)

Fig. 13 Moment invariants of generated 3D connected volume data for GAN over 40,000 samples (blue line represents the mean value)

Fig. 14 Moment invariants of generated 3D connected volume data for WGAN-GP over 40,000 samples (blue line represents the mean value)

Fig. 15 Moment invariants of generated 3D connected volume data for the improved $\alpha$-GAN over 40,000 samples (blue line represents the mean value)
Improved $\alpha$-GAN architecture for generating 3D connected volumes...  

**Fig. 16** Voxel and mesh representation of sample generated connected volumes’ shapes for the improved $\alpha$-GAN (sizes of the shapes are shown in the title of images)

- **Ratio Mean Absolute Error**: MAE of the ratio between each isocenters’ distance from the surface $D_s$ to the sum of the distance from the surface and center $D_c$ presented in (8) (excluding the isocenter located at the center). Since we have located the isocenters in the middle of the main axis, we expect this ratio to be equal to 0.5 (Fig. 11).

$$MAE = \frac{1}{N} \sum_{i=1}^{N} |0.5 - \text{Ratio}| \quad \text{where} \quad \text{Ratio} = \frac{D_s}{D_s + D_c} \quad (8)$$

- **Target distance error**: For each generated shape the target distances between isocenters are generated, and the MAE error between generated and expected target distances is calculated using (9). Target distances in the case of spheres are $\{r, r\sqrt{2}\}$ and in the case of ellipsoid are $\{r_a, r_b, r_c, \frac{1}{2}\sqrt{(r_a)^2 + (r_b)^2}, \frac{1}{2}\sqrt{(r_a)^2 + (r_c)^2}, \frac{1}{2}\sqrt{(r_b)^2 + (r_c)^2}\}$ (Fig. 11).

$$\text{Target distance error} = \frac{1}{N} \sum_{j=1}^{N} \sum_{i=1}^{N} \min_{i \neq j, i \neq \text{center}} \left( |D(\text{iso}[i], \text{iso}[j])| \right) \quad (9)$$

**Fig. 17** Moment invariant of the 3D connected tumor training data (blue line represents the mean value)
4.2 Comparison of GAN performance over different datasets

We provide detailed performance evaluation results for baseline GAN architectures and our proposed GAN over four synthetic datasets. Firstly, we report our results for 3D connected volumes of sphere/ellipsoid shape. We elaborate on our findings and provide a sample of generated shapes. Next, we choose the best performing models and apply them to generate connected tumor volumes. Finally, we provide our results for connected volumes packed with subspheres. Moreover, Appendix provides a detailed visual comparison for the training data and generated samples.

4.2.1 Results with connected 3D volumes

We report the average of generated connected volumes’ sizes, their coverage ratio, moment invariants, and KL divergence between the training data distribution of volumes’ sizes, connectivity ratio, and convex ratio in Table 3.

We observe that the vanilla GAN model performs poorly compared to the other models. This confirms the GAN’s instability and [13]’s choice of replacing the simple GAN loss function with WGAN-GP. Other models show a high coverage ratio as well as similar 3D moment invariants to the training data. We can also observe the effectiveness of α-GAN by comparing it with WGAN, which shows that the coverage ratio is increased by 6% and the convexity ratio is decreased from 3.919 to 0.105. The high coverage percentage shows the model’s capability in generating connected shapes. We note that α-GAN variants outperform the baseline GAN architectures by a significant margin. That is, the improved α-GAN represents more similarity to the training data by comparing the values of moment invariants. However, the other two variants outperform improved α-GAN in terms of KL divergence values for different evaluation metrics. This might be because the high convexity ratios of sphere/ellipsoid shapes deteriorate the models’ performance.

We illustrate the distribution of 3D moment invariants and report the mean and standard deviation for training data in Fig. 12. The 3D moment invariants of generated samples for GAN, WGAN-GP and improved α-GAN are presented in Figs. 13, 14, and 15, respectively, to provide a comparison against the corresponding distribution from the training data. The generated distributions for GAN have a similar shape to the training data. However, the values for all three moment invariants are significantly different from the target data. Similarly, the distribution and moment invariants of WGAN-GP do not represent the training data. On the other hand, the mean and standard deviation values for improved α-GAN closely follow the training data and show that the generated data consistently exhibit a higher mean and standard deviation.

| GAN model          | Connected volumes sizes | Coverage ratio | Connected volumes sizes | Coverage ratio | Connected volumes sizes | Coverage ratio | Connected volumes sizes | Coverage ratio | Connected volumes sizes | Coverage ratio | Connected volumes sizes | Coverage ratio |
|--------------------|-------------------------|----------------|-------------------------|----------------|-------------------------|----------------|-------------------------|----------------|-------------------------|----------------|-------------------------|----------------|
| WGAN-GP            | 1.841±0.277             | 0.33           | 1.96±0.110              | 0.96          | 0.945±0.171             | 0.93           | 0.978±0.160             | 0.93           | 0.957±0.166             | 0.89           | 0.93±0.138              | 0.90±0.166     |
| α-GAN              | 272±77                  | 0.96           | 4.644±1.209             | 0.96          | 0.945±0.171             | 0.93           | 0.978±0.160             | 0.93           | 0.957±0.166             | 0.89           | 0.93±0.138              | 0.90±0.166     |
| α-GAN-CL†          | 290±92                  | 0.73           | 3.508±1.096             | 0.73           | 0.719±0.130             | 0.73           | 0.719±0.130             | 0.73           | 0.719±0.130             | 0.73           | 0.719±0.130             | 0.73           |
| α-GAN++‡           | 33±106                  | 0.89           | 4.120±1.348             | 0.89           | 0.957±0.166             | 0.89           | 0.957±0.166             | 0.89           | 0.957±0.166             | 0.89           | 0.957±0.166             | 0.89           |

†: α-GAN with connected loss, ‡: Improved α-GAN
A sample of generated connected volumes’ shapes for improved $\alpha$-GAN is illustrated in Fig. 16. All three shapes are highly convex and fully connected.

The connected 3D tumor volumes dataset consists of fully connected volumes with more variable shapes, as they can reflect convex or non-convex characteristics in different parts of the shape. We illustrate the distribution of the 3D moment invariants ($\Omega_1, \Omega_2, \Omega_3$) in Fig. 17. A comparison with Fig. 12 for sphere/ellipsoid connected volumes demonstrates the difference in shapes.

The summary results for the best-performing models are reported in Table 4. The GAN model is excluded from experiments due to its instability and low performance on 3D connected volumes. The high coverage ratio, low KL divergence, and close 3D moment invariants corroborate our previous findings. However, we note that the improved $\alpha$-GAN loses its advantage in generating similar moment invariant values for all three moments. Moreover, WGAN-GP is outperformed by $\alpha$-GAN variants by a large margin. Since the datasets filled with subspheres are more complicated than 3D connected volumes and tumors, we expect the WGAN-GP to be highly unstable for those datasets and thus, exclude it from the rest of the experiments.

The distribution of the 3D moment invariants ($\Omega_1, \Omega_2, \Omega_3$) for generated samples by WGAN-GP and improved $\alpha$-GAN are illustrated in Fig. 18 and 19. The mean and standard deviation values for improved $\alpha$-GAN show that the generated data consistently exhibit a higher mean and standard deviation. The model’s ability to generate more realistic-looking samples can be attributed to the diverse shapes of connected 3D tumors and lower levels of convexity compared to the spheres. Moreover, the distribution of 3D moment invariants for WGAN-GP significantly varies from the training data and further confirms the model’s failure in generating 3D connected tumors.

A sample of generated shapes illustrated in Fig. 20 shows the difference between the two datasets. The samples provide a clear look into the irregularities of a real volume’s shape.

### 4.2.2 Results with synthetic 3D connected volume with filled subspheres

We report the shape and distance metrics for 3D connected volumes with packed spheres in Table 5. The improved $\alpha$-GAN structure shows a clear advantage in generating more connected subspheres. The low value of 1.170 for volume size KL divergence shows the capability of improved $\alpha$-GAN in generating connected shapes. This result attests to the ability of the model in generating representative 3D invariants in connected volumes as well. The connected loss...
Voxel and mesh representation of sample generated tumor volumes' shapes for the improved $\alpha$-GAN (sizes of the shapes are shown in the title of images).

A sample generated 3D connected volume shape and its subspheres are illustrated in Fig. 21. Samples represent two volumes with different sizes and their respective subspheres.

Analyzing the distance metrics shows that subspheres' coverage decreases for improved $\alpha$-GAN. The lower distance metrics show that shape and distance metrics should jointly be considered for a model's evaluation. The lower distance metrics show that the connected subspheres' coverage are more important than the subspheres' coverage. A set of unconnected points for one subsphere can cover a higher percentage of the connected volume. However, the generated shape does not qualify as suitable realistic-looking synthetic data.

Table 5: Summary of performance values for the connected 3D volumes plus subspheres averaged over 5 repeats

(a) Shape metrics

| GAN model     | Connected volumes' sizes | Connected subspheres per object | Subspheres' coverage | KL divergence |
|---------------|--------------------------|---------------------------------|----------------------|--------------|
| $\alpha$-GAN  | 278 ± 76                 | 0.001 ± 0.016                   | 0.724 ± 0.058        | 11.289 ± 0.682 |
| $\alpha$-GAN-CL† | 352 ± 109               | 0.013 ± 0.042                   | 0.675 ± 0.058        | 9.826 ± 0.662  |
| $\alpha$-GAN++‡ | 1,240 ± 257             | 0.073 ± 0.110                   | 0.389 ± 0.066        | 1.170 ± 0.171  |

(b) Distance metrics

| GAN model     | FD error | Ratio MAE | Target distance error |
|---------------|----------|-----------|-----------------------|
| $\alpha$-GAN  | 4.176 ± 0.499 | 0.072 ± 0.056 | 14.187 ± 2.075       |
| $\alpha$-GAN-CL† | 4.371 ± 0.475 | 0.083 ± 0.062 | 14.844 ± 1.785       |
| $\alpha$-GAN++‡ | 4.674 ± 0.528 | 0.122 ± 0.055 | 12.462 ± 1.986       |

†: $\alpha$-GAN with connected loss, ‡: Improved $\alpha$-GAN
Improved $\alpha$-GAN architecture for generating 3D connected volumes...

While the improved $\alpha$-GAN structure provides a higher percentage of connected subspheres per object compared to other models, the percentage is still smaller than expected. Therefore, a large portion of presented isocenters does not fit to the correct predefined coordinates.

We report the shape evaluation metrics for each generated subsphere in Table 6. The improved $\alpha$-GAN architecture shows high values of connected subspheres per connected volume, confirming the model’s ability in generating highly connected 3D objects. As expected, the
### Table 6

| GAN model          | KL divergence | Shannon index | Connectivity ratio | Convexity ratio |
|--------------------|---------------|---------------|-------------------|-----------------|
| \(\alpha\)-GAN     | 2.69 ± 0.332  | 0.616 ± 0.095 | 0.456 ± 0.138     | 0.616 ± 0.138   |
| \(\alpha\)-GAN-CL†  | 2.57 ± 0.336  | 0.641 ± 0.081 | 0.476 ± 0.120     | 0.476 ± 0.120   |
| \(\alpha\)-GAN++‡   | 1.48 ± 0.50   | 0.531 ± 0.153 | 0.446 ± 0.120     | 0.446 ± 0.120   |

\(\alpha\)-GAN with connected loss. †: \(\alpha\)-GAN with connected loss. ‡: Improved \(\alpha\)-GAN

Subspheres’ coverage ratio is lower than the other two variants. On the other hand, in contrast to the result for other datasets, improved \(\alpha\)-GAN shows great resistance to the variability of the number of isocenters and their subspheres’ irregular shapes and outperforms others in terms of the KL divergence of evaluation metrics. Note that, due to the variant sizes of subspheres in this experiment, KL divergence is reported rather than the Shannon index.

A sample 3D connected tumor volume along with its relevant subspheres are illustrated in Fig. 22. We observe that sample connected volumes present different numbers of subspheres with diverse sets of radiiuses. The location of generated isocenters shows a reasonable distribution over the tumor space.

## 5 Discussions

Our numerical study demonstrates the capabilities and limitations of 3D GAN architectures in generating 3D connected volumes. In particular, we note that the addition of predefined subspheres to the connected volumes further complicates the problem. As such, it is deemed essential to investigate custom 3D GAN architectures for this task. We observe that \(\alpha\)-GAN outperforms the baseline GAN architectures in generating 3D connected volumes. However, due to involving the subspheres and increasing the data diversity, \(\alpha\)-GAN fails to maintain the connectivity within 3D shapes and subspheres. We also find that improved \(\alpha\)-GAN is more resilient to variable subsphere sizes, and outperforms the other models in volume plus subsphere generation for tumor synthetic data. Our proposed modifications in the discriminator’s network lead to a comparable performance with the \(\alpha\)-GAN in generating connected shapes. More importantly, it shows more realistic-looking samples compared to \(\alpha\)-GAN.

Preserving the connectivity of the generated shapes is one of the main goals of this study. Therefore, the presented results reveal an important finding in this regard, as they show the impact of inception layers inside the discriminator in detecting the connectivity between the 3D voxels. Our proposed 3D \(\alpha\)-GAN architecture can also handle connected 3D volumes together with their subspheres. More importantly, the results of our proposed architecture on volumes with subspheres highlight the value of having small kernels in the inception layers to better detect the connectivity features. In this regard, the ability of our approach for generating 3D tumor volumes with designated isocenter locations (i.e., as subspheres) has the potential to significantly contribute to research on radiosurgery treatment planning as it opens the door for open sourcing realistic-looking datasets.
In this study, we limit our approach to generating one connected volume in each 3D object to simplify the 3D generation process. Therefore, the proposed methodology should be tuned further for samples of real tumor data. Moreover, we have selected a restricted set of shapes to model the tumors which simplify the generation process even further.

**Fig. 22** Sample generated tumors with subspheres and relative isocenters for the improved $\alpha$-GAN (different angles are illustrated)
6 Conclusions

In this paper, we examine two baseline GAN models along with \( \alpha \)-GAN variants on four different synthetically generated datasets. Experimental results reveal that \( \alpha \)-GAN models outperform the baseline GAN models in generating connected 3D volumes. Moreover, our proposed model, improved \( \alpha \)-GAN, reveal that incorporating inception blocks plays an important role in generating 3D connected shapes, particularly in 3D volumes packed with subspheres.

In future research, we plan to investigate the effect of additional connected volumes on the connectivity of samples. In addition, as we only focused on tumor volumes as connected 3D shapes, investigating the applications of 3D connected volumes in other domains is left for future research. The methods and results obtained in this work can benefit further research on radiation therapy treatment planning as well.

That is, the connectivity between tumor voxels is a crucial aspect that can be preserved using the 3D GAN architectures, which can help generating high-quality synthetic datasets.

Appendix

Notations

We provide a summary of mathematical notations used in the paper in Table 7.

| Notation          | Description                                                                 |
|-------------------|-----------------------------------------------------------------------------|
| \( G(z; \theta_g) \) | Generator network of GAN with \( z \) input and \( \theta_g \) set of parameters |
| \( D(x; \theta_d) \) | Discriminator network of GAN with \( x \) input and \( \theta_d \) set of parameters |
| \( D_c(x; \theta_c) \) | Code discriminator of \( \alpha \)-GAN with \( x \) input and \( \theta_c \) set of parameters |
| \( E(x; \theta_e) \) | Encoder network of \( \alpha \)-GAN with \( x \) input and \( \theta_e \) set of parameters |
| \( \hat{x} \)      | Reconstructed version of \( x \) generated by an auto-encoder               |
| \( L(x, \hat{x}) \) | Reconstruction loss between the \( x \) and \( \hat{x} \)                   |
| \( z_e \)          | Decoded noise vector from real samples                                       |
| \( z_r \)          | Real noise input vector                                                      |
| \( L_{GPD} \)      | Gradient penalty loss for the discriminator                                  |
| \( L_{GPC} \)      | Gradient penalty loss for the code discriminator                            |
| \( CC \)           | Connected components                                                         |

Model architectures

Table 8 presents the details of improved \( \alpha \)-GAN discriminator’s architecture using the inception block showed in Table 9. The other networks have a similar structure as Table 2.

| Network         | Layer | # of channels | Kernel size | Activation function | Batch normalization |
|-----------------|-------|---------------|-------------|---------------------|--------------------|
| Discriminator Inception 4 | Inception 4 | \( \frac{n_{in}}{2} \) | 1 | ReLU | True |
| Discriminator Inception 16 | Linear | \( \frac{n_{out}}{} \) | - | - | - |

Table 9 Structure of inception block used in improved \( \alpha \)-GAN structure

| Network | Layer | # of channels | Kernel size | Activation function | Batch normalization |
|---------|-------|---------------|-------------|---------------------|--------------------|
| Block 1 | Conv  | \( \frac{n_{in}}{2} \) | 1 | ReLU | True |
|         | Conv  | \( n_{out} \) | 5 | ReLU | True |
| Block 2 | Conv  | \( \frac{n_{in}}{2} \) | 1 | ReLU | True |
|         | Conv  | \( n_{out} \) | 3 | ReLU | True |
| Block 3 | Conv  | \( n_{out} \) | 1 | ReLU | True |
| Block 4 | MaxPool | - | - | ReLU | False |
|         | Conv  | \( n_{out} \) | 1 | ReLU | True |

Summary of performance metrics

Table 10 presents a summary of performance metrics, their description, the range of possible values, and the desired target value. The performance measured by some of the metrics requires comparing the corresponding data/metric distributions for training and test data, which is achieved via KL divergence. For instance, the convexity ratio is obtained for each data instance as the ratio of the number of generated voxels that are inside the convex shape and the total number of generated voxels. We expect the distribution of convexity ratios to be the same between training data and generated instances, and lower KL divergence values between these two distributions are desirable. Note that KL divergence values are in the range of \([0, \infty)\).
Table 10 Description of performance metrics

| Metric | Description | Interval | Desirable value |
|--------|-------------|----------|-----------------|
| (a) Shape metrics | | | |
| Connected volumes’ sizes | Number of connected voxels | $[0, \infty)$ | Match the training data, checked via KL divergence |
| Convexity ratio | Number of generated voxels that are inside the convex shape divided by the total number of generated voxels | $[0, 1]$ | Match the training data, checked via KL divergence |
| Connectivity ratio | Ratio of the number of clustered points to the number of processed points | $[0, 1]$ | Match the training data, checked via KL divergence |
| Coverage ratio | Ratio of generated samples that meet the pre-defined thresholds of convexity and connectivity ratio | $[0, 1]$ | 1 |
| 3D moment invariants | Spatial descriptors used to quantify the distribution of an object’s solid volume | $[0, \infty)$ | Match the training data |
| Shannon equitability index | Measures the similarity (i.e., evenness) of different isocenters | $[0, 1]$ | Match the training data, checked via KL divergence |
| Subspheres’ coverage | Ratio of voxels that are covered by the packed spheres to the ratio of the entire connected volume | $[0, 1]$ | 1 |
| Connected subspheres per connected volume | Ratio of subspheres per volume that satisfy the coverage ratio | $[0, 1]$ | 1 |
| (b) Distance metrics | | | |
| FD error | Discrete Fréchet distance between the ground truth location of actual and generated isocenters | $[0, \infty)$ | Lower values |
| Ratio Mean Absolute Error | MAE of the ratio between each isocenters’ distance from the surface $D_s$ to the sum of the distance from the surface and center $D_c$ | $[0, \infty)$ | Lower values |
| Target distance error | MAE between generated and expected target distances | $[0, \infty)$ | Lower values |

Visual comparison of training and generated samples

In this section, we present side-by-side illustrations of the training data and generated samples by improved $\alpha$-GAN to highlight the model’s performance. Figure 23 and 24 illustrate training samples and most similar generated data samples for 3D connected and tumor volumes. Generated shapes maintain the connectivity of the voxels, but the level of the convexity is slightly reduced compared in the provided sample compared to the training data.

Fig. 25 and 26 illustrate sample training and generated data instances as obtained by improved $\alpha$-GAN for 3D volumes and tumors filled with subspheres. Generated shapes for the 3D volumes have smaller subspheres and thus, more concentrated isocenters. However, isocenters’ distribution follows a uniform pattern similar to the training data.
Fig. 23  Voxel and mesh representation of the connected 3D volumes for the training data and improved $\alpha$-GAN generated data (closest generated samples to the training data are selected for presentation)

Fig. 24  Voxel and mesh representation of the connected 3D tumors for the training data and improved $\alpha$-GAN generated data (closest generated samples to the training data are selected for presentation)
Improved $\alpha$-GAN architecture for generating 3D connected volumes...

Fig. 25  Mesh representation of synthetic training data and generated connected 3D volumes with filled subspheres from four different angles.
Fig. 26  Mesh representation of synthetic training data and generated connected tumor volumes with filled subspheres from four different angles.
Data Availability All the datasets are publicly available, and can be obtained using the described methods.

Declarations

Conflict of Interests No potential conflict of interest was reported by the authors.

References

1. Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y (2014) Generative Adversarial Nets. In: Advances in Neural Information Processing Systems, pp 2672–2680
2. Lee H-Y, Tseng H-Y, Huang J-B, Singh M, Yang M-H (2018) Diverse image-to-image translation via disentangled representations. In: Proceedings of the IEEE Conference on Computer Vision (ECCV), pp 35–51
3. Wang T-C, Liu M-Y, Zhu J-Y, Tao A, Kautz J, Catanzaro B (2018) High-resolution image synthesis and semantic manipulation with conditional GANs. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 8798–8807
4. Zhong X, Qu X, Chen C (2019) High-quality face image super-resolution based on generative adversarial networks. In: 2019 IEEE 4th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), vol 1, pp 1178–1182. IEEE
5. Brock A, Donahue J, Simonyan K (2019) Large scale GAN training for high fidelity natural image synthesis. In: International Conference on Learning Representations. https://openreview.net/forum?id=B1xsgj09Fm
6. Mohammadjafari S, Ozyegen O, Cevik M, Kavurmacioglu E, Ethier J, Basar A (2021) Designing mm-wave electromagnetic engineered surfaces using generative adversarial networks. Neural Comput Appl 1–15
7. Suzuki K (2017) Overview of deep learning in medical imaging. Radiol Phys Technol 10(3):257–273
8. Chan ER, Monteiro M, Kellnhofer P, Wu J, Wetzstein G (2021) Pi-GAN: Periodic implicit generative adversarial networks for 3D-aware image synthesis. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 5709–5809
9. Or-El R, Luo X, Shan M, Shechtman E, Park JJ, Kembel-Macher-Shlizerman I (2022) Stylesdf: High-resolution 3D-consistent image and geometry generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 13503–13513
10. Han C, Kitamura Y, Kudo A, Ichinose A, Rundo L, Furukawa Y, Umemoto K, Li Y, Nakayama H (2019) Synthesizing diverse lung nodules wherever massively: 3D multi-conditional GAN-based CT image augmentation for object detection. In: 2019 International Conference on 3D Vision (3DV), pp 729–737. IEEE
11. Mahmood R, Babier A, McNiven AL, Diamant A, Chan TC (2018) Automated treatment planning in radiation therapy using generative adversarial networks. In: Machine Learning for Healthcare Conference, pp 484–499. PMLR
12. Berdyshev A, Cevik M, Alemán D, Nordstrom H, Riad S, Lee Y, Sahgal A, Ruscini M (2020) Knowledge-based isocenter selection in radiosurgery planning. Med Phys 47(9):3913–3927
13. Kwon G, Han C, Kim D-s (2019) Generation of 3D brain MRI using auto-encoding generative adversarial networks. In: International Conference on Medical Image Computing and Computer-Assisted Intervention, pp 118–126. Springer
14. Wu J, Zhang C, Xue T, Freeman B, Tenenbaum J (2016) Learning a probabilistic latent space of object shapes via 3D generative-adversarial modeling. Adv Neural Inf Process Syst 29
15. Choy C, Xu D, Gwak J, Chen K, Savarese S (2016) 3D-r2n2: A unified approach for single and multi-view 3D object reconstruction. In: European Conference on Computer Vision. Springer, pp 628–644
16. Everingham M, Winn J (2011) The pascal visual object classes challenge 2012 (voc2012) development kit. Pattern Analysis, Statistical Modelling and Computational Learning, Tech. Rep 8(5)
17. Oh Song H, Xiang Y, Jegelka S, Savarese S (2016) Deep metric learning via lifted structured feature embedding. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 4004–4012
18. Weiner MW (2004) Alzheimer’s Disease Neuroimaging Initiative (ADNI), https://adni.loni.usc.edu/
19. Bakas S, Akbari H, Sotiras A, Bilicello M, Rosycki M, Kirby JS, Freymann JB, Farahani K, Davatzikos C (2017) Advancing the cancer genome ATLAS glioma MRI collections with expert segmentation labels and radiomic features. Scientific Data 4(1):1–13
20. Babier A, Mahmood R, McNiven AL, Diamant A, Chan TC (2020) Knowledge-based automated planning with three-dimensional generative adversarial networks. Med Phys 47(2):297–306
21. Hong S, Marinescu R, Dalca AV, Bonkhoff AK, Bretzner M, Rost NS, Golland P (2021) 3D-StyleGAN: A Style-based generative adversarial network for generative modeling of three-dimensional medical images. In: Deep Generative Models, and Data Augmentation, Labelling, and Imperfections. Springer, New York, pp 24–34
22. Jangid DK, Brodnik NR, Khan A, Goebel MG, Echlin MP, Pollock TM, Daly SH, Manjunath B (2022) 3D, grain shape generation in polycrystals using generative adversarial networks. Integrating Materials and Manufacturing Innovation, 1–14
23. Xiang Y, Kim W, Chen W, Ji J, Choy C, Su H, Mottaghi R, Guibas L, Savarese S (2016) ObjectNet3D: A large scale database for 3D object recognition. In: European Conference on Computer Vision, pp 160–176. Springer
24. Zhi S, Liu Y, Li X, Guo Y (2018) Toward real-time 3D object recognition: A lightweight volumetric cnn framework using multitask learning. Comput Graph 71:199–207
25. Wu Z, Song S, Khosla A, Yu F, Zhang L, Tang X, Xiao J (2015) 3D shapenets: A deep representation for volumetric shapes. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 1912–1920
26. Maturana D, Scherer S (2015) Voxnet: A 3D convolutional neural network for real-time object recognition. In: 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp 922–928. IEEE
27. Li Y, Su H, Qi CR, Fish N, Cohen-Or D, Guibas LJ (2015) Joint embeddings of shapes and images via cnn image fusion. ACM Trans Graph (TOG) 34(6):1–12
28. Jimenez Rezende D, Eslami S, Mohamed S, Battaglia P, Jaderberg M, Heess N (2016) Unsupervised learning of 3D, structure from images. Adv Neural Inf Proces Syst 29
29. Smith E, Fujimoto S, Meger D (2018) Multi-view silhouette and depth decomposition for high resolution 3D, object representation. Adv Neural Inf Proces Syst 31
30. Sharma A, Grau O, Fritz M (2016) Vconv-dae: Deep volumetric shape learning without object labels. In: European Conference on Computer Vision, pp 236–250. Springer
31. Zhang X, Zhang Z, Zhang C, Tenenbaum J, Freeman B, Wu J (2018) Learning to reconstruct shapes from unseen classes. Adv Neural Inf Proces Syst 31
32. Wu J, Wang Y, Xue T, Sun X, Freeman B, Tenenbaum J (2017) Marrnet: 3D shape reconstruction via 2.5D sketches. Adv Neural Inf Proces Syst 30
33. Karris T, Laine S, Aittala M, Hellsten J, Lehtinen J, Aila T (2020) Analyzing and improving the image quality of StyleGAN. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 8110–8119
34. Karris T, Laine S, Aila T (2019) A style-based generator architecture for generative adversarial networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 4401–4410
35. Rosca M, Lakshminarayanan B, Warde-Farley D, Mohamed S (2017) Variational approaches for auto-encoding generative adversarial networks. arXiv preprint arXiv: 1706.04987
36. Gulrajani I, Ahmed F, Arjovsky M, Dumoulin V, Courville AC (2017) Improved training of Wasserstein GANs. In: Advances in Neural Information Processing Systems, pp 5767–5777
37. Volokitin A, Erdil E, Karani N, Tezcan KC, Chen X, Gool LV, Konukoglu E (2020) Modelling the distribution of 3D brain MRI using a 2D slice VAE. In: International Conference on Medical Image Computing and Computer-Assisted Intervention, pp 657–666. Springer
38. Chong CK, Ho ETW (2021) Synthesis of 3D MRI brain images with shape and texture generative adversarial deep neural networks. IEEE Access 9:64747–64760
39. Cevik M, Ghomi PS, Aleman D, Lee Y, Berdyshev A, Nordstrom H, Riad S, Sahgal A, Ruschin M (2018) Modeling and comparison of alternative approaches for sector duration optimization in a dedicated radiosurgery system. Phys Med Biol 63(15):155009
40. Isola P, Zhu J-Y, Zhou T, Efros AA (2017) Image-to-image translation with conditional adversarial networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 1125–1134
41. Kingma DP, Welling M (2014) Auto-encoding variational bayes. In: Bengio Y, LeCun Y (eds) 2nd International Conference on Learning Representations, ICLR, 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings
42. Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A (2015) Going deeper with convolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 1–9
43. Ghobadi K, Ghaffari HR, Aleman DM, Jaffray DA, Ruschin M (2012) Automated treatment planning for a dedicated multi-source intracranial radiosurgery treatment unit using projected gradient and grassfire algorithms. Med Phys 39(6Part1):3134–3141
44. Lin Z, Khetan A, Fanti G, Oh S (2018) Pacgan: The power of two samples in generative adversarial networks. In: Advances in Neural Information Processing Systems, pp 1498–1507
45. Schonfeld E, Schiele B, Khoreva A (2020) A U-Net based discriminator for generative adversarial networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 8207–8216
46. Kazhdan M, Funkhouser T, Rusinkiewicz S (2003) Rotation invariant spherical harmonic representation of 3D shape descriptors. In: Symposium on Geometry Processing, pp 156–164

Publisher’s note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.