

Three-dimensional shape generation via variational autoencoder generative adversarial network with signed distance function

Ebenezer Akinyemi Ajayi, Kian Ming Lim, Siew-Chin Chong, Chin Poo Lee

Faculty of Information, Science and Technology, Multimedia University, Melaka, Malaysia

Article Info

Article history:

Received Jun 13, 2022

Revised Sep 22, 2022

Accepted Oct 1, 2022

Keywords:

3D shape generation

Convolution neural network

Generative adversarial network

Signed distance function

Variational autoencoder

ABSTRACT

Mesh-based 3-dimensional (3D) shape generation from a 2-dimensional (2D) image using a convolution neural network (CNN) framework is an open problem in the computer graphics and vision domains. Most existing CNN-based frameworks lack robust algorithms that can scale well without combining different shape parts. Also, most CNN-based algorithms lack suitable 3D data representations that can fit into CNN without modification(s) to produce high-quality 3D shapes. This paper presents an approach that integrates a variational autoencoder (VAE) and a generative adversarial network (GAN) called 3 dimensional variational autoencoder signed distance function generative adversarial network (3D-VAE-SDFGAN) to create a 3D shape from a 2D image that considerably improves scalability and visual quality. The proposed method only feeds a single 2D image into the network to produce a mesh-based 3D shape. The network encodes a 2D image of the 3D object into the latent representations, and implicit surface representations of 3D objects corresponding to those 2D images are subsequently generated. Hence, a signed distance function (SDF) is proposed to maintain object inside-outside information in the implicit surface representation. Polygon mesh surfaces are then produced using the marching cubes algorithm. The ShapeNet dataset was used in the experiments to evaluate the proposed 3D-VAE-SDFGAN. The experimental results show that 3D-VAE-SDFGAN outperforms other state-of-the-art models.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Kian Ming Lim

Faculty of Information, Science and Technology, Multimedia University

Melaka Campus, 75450, Melaka, Malaysia

Email: kmlim@mmu.edu.my

1. INTRODUCTION

The urgent need to automatically generate new three-dimensional (3D) mesh-based shapes to populate a virtual world appeals to computer graphics (CG) and computer vision (CV) specialists. Advances in robotics, artificial intelligence (AI), game, virtual, and augmented reality have boosted the 3D model generation via 3D deep learning due to the high demand of real-time shape analysis and synthesis in CV domain. In the last decade, many approaches have emerged to alleviate this problem, especially in the computer graphics domain. However, majority of these works were assembly-based 3D modeling [1]–[3] which used a database of 3D models to synthesize new shapes from various parts of existing models. Though these techniques achieved impressive results, the model generation pipelines focused on a single mode, which caused their models not to be robust. In addition, these techniques are not able to capture more

complex variability of geometric features. Furthermore, techniques used heuristic methods and generate new 3D shapes using browsing parts from existing 3D model databases, and thereby, fail to produce realistic results.

Success of two dimensional (2D) deep generative networks [4]–[9], [10]–[12] in handling generative tasks with 2D images encourage researchers to extend generative adversarial network (GAN), variational auto-encoder (VAE) and VAE-GAN with 3D convolutional neural network (CNN) to perform 3D generative tasks. 3D deep learning data representation [13] allows researchers to take advantage of various available 3D data representations to perform end-to-end learning tasks. This has inspired researchers to explore learning object representation based on voxel [13]–[15] and point cloud [16], [17]. Furthermore, recent 3D deep learning algorithms take advantage of the advancements in deep learning proposed in [18]. Therefore, generative modelling algorithms such as GAN, VAE, and auto-decoder (AD) were implemented with 3D-CNN to generate 3D shapes.

This progress inspired Wu *et al.* [13] to extend generative adversarial nets (2D GAN) [4] to create 3D voxel-based shapes with low resolution, and large memory footprint, except for a tessellated grid that leveraged the octrees approach for high-resolution 3D voxel shapes [19]. Other works [15], [20] also follow the similar 3D shape generation pipeline of [13] to create 3D voxel-based shapes. Furthermore, point-cloud and mesh data representations were also used as input to 3D-GAN and 3D-VAE [17], [21], [22] to generate 3D shapes with high resolution, compact, and computationally less expensive. However, the irregularity in organization structure and disorderliness characteristics make it unfit for the learning process with CNN. In addition, their implementations are not as easy as voxels data representation with deep convolutional neural networks.

Later on, some works [13]–[15] infer latent vectors from observations by mapping a 2D image to the latent representation in GAN to enhance the recovery of a 3D object corresponding to the 2D image. The results obtained were impressive for shape generation tasks using deep generative networks because voxel grids fit the learning process and convolution is used for its rendering. However, the output shapes are coarse in nature. Furthermore, the current mesh-oriented algorithms cannot process voxel-based geometries [23].

Considering recent advances in CG and CV, we propose an efficient deep learning approach to generate realistic 3D mesh shapes from a 2D image leveraging the signed distance function (SDF)-based VAE-GAN framework, referred to as 3D-VAE-SDFGAN. We seek to produce an SDF field on the gridded domain similar to [23], [24]. Deconvolution layers are employed to encourage a polygon mesh surface reconstruction for a higher quality 3D shape generation. This work will give some contributions such as:

- The integration of VAE and GAN learns to encode, generate, and compare data simultaneously. Our proposed network explicitly learns the latent spaces of 2D images. These latent spaces are used to produce corresponding signed distance functions of objects, which are then reconstructed into 3D mesh-based shapes.
- The proposed 3D-VAE-SDFGAN model manages to generate a high-quality 3D shape from its corresponding 2D image.
- The performance of the proposed method is evaluated qualitatively and quantitatively. From the experiment results, the proposed 3D-SDF-VAE-GAN outperforms state-of-the-art 3D shape generative methods.

Related works are presented in section 2. The concept of the SDF, data pre-processing, and model architecture for the proposed solution are discussed in section 3. Section 4 describes the training procedure and the results of the experiment. Finally, we conclude this work in section 5.

2. RELATED WORK

2.1. 3D shapes modeling and generation

A 3D shape generation is a challenging problem in CG and CV. Wu *et al.* [13] have attempted to develop or learn 3D object representations based on meshes and skeletons. Many of these non-parametric-based synthesis algorithms create new objects by collecting and combining shapes and shape parts from the database. Chaudhuri *et al.* [1] proposed a 3D model generation system that leverages a probabilistic graphical model capable of encoding semantic and geometric links between shape parts to produce a 3D model. The 3D shapes generated semantically and physically resemble the objects from the database. Huang *et al.* [25] investigated the generation of 3D shapes using pre-trained templates, which produced both the object structure and surface geometry. However, these cited works are naturally incapable of creating conceptually unique shapes or providing a better representation of these shapes. In contrast, our proposed approach generates 3D mesh-based shapes without the need to collect and combine shapes and shape parts from a database in an unsupervised manner. Also, our proposed method can generate novel 3D shapes from 2D images.

2.2. Three-dimensional shape generation via deep learning approaches

Following the successes recorded in the 2D domain with deep learning, deep learning has gradually moved to the 3D domain for 3D shape generation tasks. Many researchers have investigated part-based deep learning 3D shape generation systems to produce plausible 3D shapes. For example, Li *et al.* [26] proposed the first part-based deep generative model that produces plausible 3D shapes. The work used a recursive neural network autoencoder to attain hierarchical encoding and decoding of components and relations. Li *et al.* [27] proposed a PARANet leveraging an array of per-part VAE-GANs to generate semantic parts of a complete shape. Later, transform and assemble the produced semantic parts into a plausible 3D shape using a part assembly module. Also, a recurrent neural network-based 3D shape generation system was proposed by Zou *et al.* [28] to learn sequential part creation, which only generates cuboids and is not geometrically precise. Furuya *et al.* [29] proposed HMF-Nets leveraging blocks of token-mixing layers and weighted chamfer distance (WCD) loss to train hyperplane patches to reconstruct better 3D shape details from the 2D image. However, their model requires a robust encoder network to produce richer latent 3D shape characteristics. Also, the model's output is full of holes compared to our proposed model's results. Though the cited works were deep learning-based approaches, they combined many shape parts to produce 3D shapes. Such a combination limits their results from being realistic compared to our proposed method. The generated 3D shapes were of low quality, and the results lacked detailed geometry compared to our proposed approach.

Recently, 3D data representation, classification, and generation tasks using deep learning have been extensively studied. Various 3D data representations have been used to generate 3D shapes. Examples of such data representations are voxel-based, point cloud-based, mesh-based, multi-view images, or depth-images. Balashova *et al.* [30] proposed a 3D shape generation approach that leverages a structure-aware loss function. Their framework consists of a shape encoder, a shape generator, and structure detector networks. The model incorporates structural information into its training pipeline in an end-to-end manner to impose structural limitations and provide uniformity and structure throughout the entire manifold. However, their model fails to capture complex data information compared to our proposed method. Their model only generates coarse 3D shapes. Zhirong *et al.* [18] performed 3D shape completion and recognition tasks using volumetric data representation as an input. 3D shape generation from a probabilistic latent space proposed by Wu *et al.* [13] using generative adversarial networks (GAN). PrGANs [31] used a GANs framework to train a projector. Their discriminator network was trained to discriminate projected images of a real sample from those projected samples from generative models, while their generator network learned to generate 3D models. Zhu *et al.* [14] constructed an architecture on the GANs framework and incorporated 2D image enhancer network that feeds high-level image information into a 3D model generator network for effective model training. The architecture was trained on both 2D images and 3D models simultaneously. The output is a voxel-based 3D shape, which is computationally expensive with a large memory footprint. In contrast with these models, our proposed model produces mesh-based 3D shapes instead of voxels with a similar framework. The generated 3D shape achieves better quality and is computationally less expensive with a smaller memory footprint. To date, no work has directly mapped 2D images using the VAE-GAN framework to 3D mesh-based shapes.

2.3. 3D shape generation with signed distance functions

CV and CG researchers have recently adopted the signed distance function (SDF) as an alternative for 3D data representations in 3D shape generation. SDF overcomes the limitations such as the large memory footprint, irregular nature, unstructured, and disorderliness characteristics as found in other typical 3D data representations (voxels, point-cloud, and mesh). This has made it a suitable mechanism for mesh-based 3D shape generation tasks. SDF learning is an implicit function learning that expresses the structural relationship distance on the 3D surface.

Wu *et al.* [32] leveraged sequential part assembly and proposed PQ-NET, a variant of deep neural network (DNN) for 3D shape generation. The part-features representation was used as input to the seq2seq autoencoder network to generate a fixed-length latent vector that encourages many generative tasks. The decoder network reconstructs 3D shape as an SDF data representation using the latent vector and 3D point that enables high-quality 3D shapes. However, the approach fails to learn part relations for structure understanding and does not encourage a topology-altering interpolation scheme for shapes with distinct parts. Part structure and geometry were encoded randomly and inter-twisted, affecting the learned latent space quality. Zheng *et al.* [33] proposed a deep implicit template using the spatial warping long short-term memory (LSTM) for 3D shape representation in high quality with dense correspondences to provide semantic relationship information across shapes. The model breaks up the conditional-signed distance function into a conditional spatial warping function which maps a point (p) coordinate to a new 3D coordinate and returns SDF values at the new 3D coordinate. The model achieves an ideal prototype that portrays objects with a standard structure for shape generalization. Xu *et al.* [34] proposed a deep implicit surface network (DISN)

capable of capturing holes and thin structures of 3D shapes from single-view images to generate high-quality 3D shapes. It combined global and local image features to predict an improved and accurate signed distance field for 3D shapes. Liu *et al.* [35] proposed an IMLSNet that uses an octree-based autoencoder to implement 3D shape generation. IMLSNet used the Octree structure, SDF, moving least-squares (MLS) point repulsion, projection smoothness, and radius smoothness losses to fit the sampled SDF. The methods cited above are autoencoder-based deep generative networks with autoencoder-based limitations. Hence, it does not generate a new instance of an object as our proposed approach does.

Jiang and Marcus [23] proposed a 3D-GAN-based hierarchical detail enhancing mesh shape generation with SDF data representation, and generated SDF fields on a gridded domain to reconstruct polygon mesh surfaces with higher quality. Despite the low-frequency generating and high-frequency generator networks presented in their architecture, the network was driven by uninformative random vectors and trained on only 3D data. Kingkan and Hashimoto [24] proposed a 3D-VAE-GAN-based framework that directly maps point clouds to other 3D shape representations. Unlike the two-3D mesh generation GAN-based frameworks discussed above, our proposed 3D-VAE-SDFGAN is trained on both 2D images and 3D models concurrently for enhancing the 3D mesh generation. To date, none of the existing research works use the VAE-3D-GAN framework to map 2D directly to 3D mesh-based shapes.

3. RESEARCH METHOD

In this section, signed distance function and 2D images are discussed in detail. In addition, geometry processing approaches to prepare the training data are presented. Subsequently, the background information on both the variational autoencoder (VAE) [36] and the generative adversarial network (GAN) [4] are presented. Then, to establish a mapping between 2D images and signed distance functions, we introduce our proposed network, which leverages VAE with GAN algorithm [5] for 3D shape generation.

3.1. Two-dimensional image and signed distance function generation

In the experiments, we use ShapeNet [37] as the dataset. ShapeNet consists of 55 common objects with 51,300 3D models. To create our image dataset used in training, we collected images provided in Choy *et al.* [38] work, which comprised rendered images of ShapeNet 3D models from 23 different views. The top row of Figure 1 shows some examples of 2D images from our 2D image dataset.

Also, for the 3D SDF field preprocessing, we highlighted the transformation procedure of 3D meshes into SDFs [39]. Motivated by [23], [24], the output of our network is signed distance functions. An SDF is a subset of implicit functions that assigns a 3D point to a real value rather than a likelihood, expressing the structural relationship and distance to the 3D surface. SDFs data representation is a suitable representation that uses signed values to represent a mesh object's inside-outside characteristics. SDF data representation becomes popular for mesh generation with deep convolutional neural networks because its presentation is not limited by fixed topology as applicable to mesh and point clouds. It also has a higher resolution when compared with voxel resolution. Given a spatial point $p \in R^3$, the sign distance function $k(p) \in R$, encodes the point's distance to its closest surface point, where p lies inside (-) or (+) of the object. Alternatively, given a set Ω in a 3D Euclidean space where Ω is a non-zero volume open set with an enclosed smooth piece-wise boundary $\delta\Omega$, a signed distance function k is defined as:

$$k(x) = \begin{cases} \text{dis}(x, \delta\Omega), & \text{if } x \in \Omega \\ 0, & \text{if } x \in \delta\Omega \\ -\text{dis}(x, \delta\Omega), & \text{if } x \in \Omega^c \end{cases} \quad (1)$$

where $\delta\Omega$ stands for the boundary of Ω . The distance from a point x , where x belongs to 3D Euclidean space to the boundary $\delta\Omega$ is defined as (2):

$$\text{dis}(x, \delta\Omega) = \inf_{y \in \delta\Omega} (x, y) \quad (2)$$

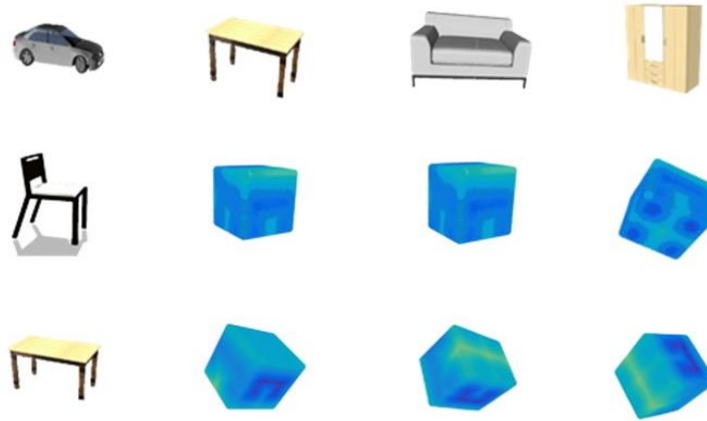
and

$$\text{sign}(x, \delta\Omega) = \begin{cases} 1 & \text{if } x \in \Omega \\ 0 & \text{if } x \in \delta\Omega \\ -1 & \text{if } x \in \Omega^c \end{cases} \quad (3)$$

The SDF in (1), which is the product of (2) and (3) can also be written as (4):

$$k(x) = \text{dis}(x, \delta\Omega) \cdot \text{sign}(x, \delta\Omega) \quad (4)$$

To produce SDF from a triangular mesh, we center and normalize the triangular mesh first. Then, we establish around the geometry a 3D unit grid with a resolution of 64^3 . Furthermore, we compute the point-to-mesh distance using an axis-aligned bounding box (AABB) tree to calculate the distance at each point in the grid. The winding number of each point is computed to provide the sign at each point in the grid.



Note:

The first row shows the 2D images of the car, table, sofa, and cabinet. The second and third rows show the *chair* and *table* meshes with their SDF views from different angles. An SDF colormap depicts the distance between each point and surfaces. The colors in this color map range from dark blue (inside) to yellow (outside)

Figure 1. Examples of 2D images and the transformation of 3D meshes into signed distance functions (SDFs)

3.2. VAE, GAN, and VAE-GAN

A VAE is an extended version of an autoencoder network that imposes additional restrictions on latent variables. The restriction turns the network into an algorithm that learns its input information from a latent variable model. VAE learns the parameters that model the data from the probability distribution [40]. The VAE network comprises encoder and decoder networks. The encoder network serves as an inference network that compresses the input data over the latent distribution $p(z)$ regulated by prior into a latent representation [24]. A decoder network, which can also be called a generator network, generates a new instance of input data from the latent representation. The VAE's weights are trained simultaneously by improving the reconstruction loss and Kullback-Leibler divergence between the latent distribution learned and a prior.

A GAN is a generative modeling algorithm used in an unsupervised manner to generate a new instance of data in an unsupervised manner from a random vector of a Gaussian probability distribution. GAN comprises a generator network and a discriminator network. The generator network is a neural network that uses random vectors drawn from a Gaussian probability distribution as input and produces a new data sample different from the training dataset but possesses the same characteristics as the training dataset. On the other hand, the discriminative neural network accepts both generated data and the training dataset as input. It evaluates if the sample data is from the dataset or generated. The generator network attempts to generate data similar to the training dataset. On the other hand, the discriminator network attempts to distinguish between the generated data and the training dataset.

Also, to improve the quality of data generation, some works combined VAE and GAN. However, it is less computationally expensive, and combining the loss functions from both networks is easy. Larsen *et al.* [5] was the first work to propose a VAE-GAN framework to acquire feature representation and similarity standards for an improved 2D image synthesis task. The combination yielded a good result compared to GAN-based and VAE-based algorithms alone. Wu *et al.* [13] combined 3D-VAE-GAN to generate a voxel-based 3D model from 2D images. Later, Kingkan and Hashimoto [24] combined 3D-VAE-GAN to generate SDF from 3D point clouds. Another work by Smith and Merger [15] combined 3D-VAE-IWGAN to perform voxel-based 3D model generation, 3D model reconstruction, and 3D shape completion from a 2D image. In view of the advantages of both VAE and GAN, we employed a similar framework to learn the latent spaces of 2D images and map 2D images to their respective SDFs.

3.3. 3D-VAE-SDFGAN framework

In this work, we propose 3D-VAE-SDFGAN to learn a 2D image's latent representation from a 2D image and produce 3D mesh-based shapes similar to 2D images. Instead of using only random vectors sampled from a Gaussian distribution, we fed the learned latent space of a 2D image from 2D-VAE to the SDF-generator network to aid better generation. In doing so, our generator network learned to generate 3D shapes from both the 2D image and SDF. The overview of the 3D-VAE-SDFGAN framework is shown in Figure 2. Our proposed 3D-VAE-SDFGAN comprises 3 components, namely a 2D-image encoder network (E) as shown in Figure 2(a), the SDF-generator network (G_{sdf}), as shown in Figure 2(b), and the SDF-discriminator network (D_{sdf}), as shown in Figure 2(c). Figure 2(a) converts a 2D image into latent spaces; Figure 2(b) produces an SDF from a 2D image's latent vector, and Figure 2(c) evaluates whether inputs are either generated or real SDFs.

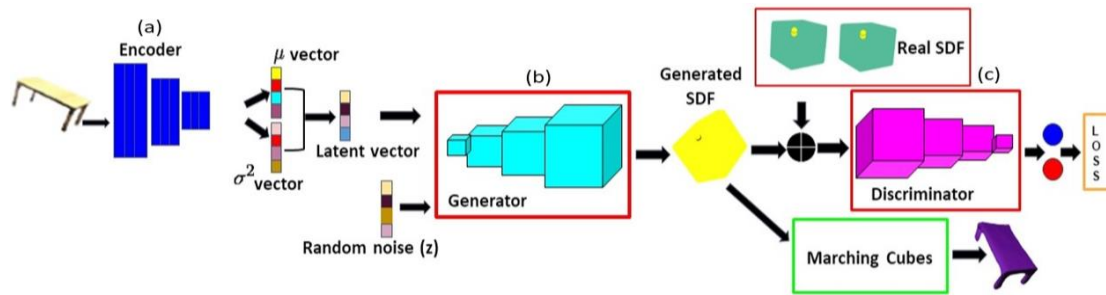


Figure 2. The proposed 3D-VAE-SDFGAN network comprises of three components (a) the encoder network converts a 2D image to latent spaces, (b) the generator network produces an SDF from a 2D image latent vector, and (c) the discriminator network evaluates whether inputs are either generated or real SDFs

3.3.1. 2D image encoder network (E)

The encoder network contains five convolution layers with the following numbers of channels {64, 128, 256, 512, 400}, a kernel with the size of {11, 5, 5, 5, 8}, and strides {4, 2, 2, 2, 1}. In between the convolution layers, the network consists of both rectified linear unit (ReLU) and batch normalization layers. The last convolution layer of the image encoder network output a latent vector of 400-dimension describing a Gaussian distribution of a 200-dimension mean latent vector and a 200-dimension of variance latent vector. A sampling layer present in the encoder network helps to sample 200-dimension latent vector input that guides the SDF-generator network to produce SDF similar to the 2D image from Gaussian distribution. The loss function of the 2D image encoder network (L_E) consists of KL divergence loss (L_{KL}) and reconstruction loss (L_R) as (5), (6):

$$L_{KL} = D_{KL} \left(q(z_{img}|img) \| p(z) \right) \quad (5)$$

$$L_R = \|G(E(img)) - sdf_{real}\|_2 \quad (6)$$

where L_{KL} is divergence loss between the prior distribution $p(z)$ from a uniform distribution over $[-1,1]$ and the learned latent distribution (z_{img}), sdf_{real} is a SDF of 3D shape from the training set, img is the corresponding 2D image, $q(z_{img}|img)$ denotes the variational distribution of latent representation. To allow the SDF-generator network to draw z_{img} from the exact distribution as $p(z)$, KL divergence is used to limit $q(z_{img}|img)$ to be as similar to $p(z)$ as possible.

3.3.2. SDF-generator and SDF-discriminator networks

SDF-generator network (G_{sdf}) comprises five transpose convolution layers with channel numbers {512, 256, 128, 64, 1} with the kernel sizes of {4, 4, 4, 4, 4}, and strides of {1, 2, 2, 2, 2}. ReLU and batch normalization layers are used between the transpose convolution layers, except the last layer that uses a Tanh function to map output into $[-1, 1]$. The generator network output is an SDF of 64^3 matrix, with values in $[-1, 1]$. Triangular mesh surfaces are then obtained from this matrix using the marching cubes algorithm (MCA) [41].

SDF-discriminator network (D_{sdf}) mirrors the SDF-generator network with leaky ReLU as activation function. Sigmoid function is used at the final layer to squash the output to $[0, 1]$. It composes of five 3D-convolution layers with channel numbers $\{64, 128, 256, 512, 1\}$ with kernel sizes of $\{4, 4, 4, 4, 4\}$, and strides of $\{2, 2, 2, 2, 1\}$. The loss function for SDFGAN is:

$$L_{SDFGAN} = \log D(sdf_{real}) + \log(1 - D(G(z))) + \log(1 - D(G(z_{img}))) \quad (7)$$

The total loss L_{total} function used in our 3D-VAE-SDFGAN framework consists of the sum of three components: reconstruction loss L_R , a cross-entropy L_{SDFGAN} , and KL divergence L_{KL} to impose a limit on the distribution of the output of the 2D encoder network.

$$L_{total} = L_{SDFGAN} + \gamma_1 L_{KL} + \gamma_2 L_R \quad (8)$$

where γ_1 and γ_2 are weights of L_{KL} and L_R respectively.

4. EXPERIMENTS

In this section, the training procedure is first discussed. Then, we compare our proposed 3D-VAE-SDFGAN with several state-of-the-art generative models. The qualitative and quantitative results are also presented.

4.1. Training procedure

The proposed 3D-VAE-SDFGAN framework is trained with a pair of $\{img_i, sdf_{real_i}\}$ drawn from the training dataset, where img_i is the 2D image, and sdf_{real_i} is the corresponding signed distance function of the 3D object. During training, the image encoder encodes a 2D image img_i into a latent representation (z_{img}) representing the image feature. The SDF generator network receives z_{img} which is a 200-dimension vector as input and generates sdf_{img} as output. We sample a random vector z from a uniform distribution $p(z)$, and fed it to the SDF-generator network to produce sdf_z . Both generated SDFs (sdf_{img}, sdf_z) with sdf_{real} serve as input to the SDF-discriminator network for classification purposes. The SDF-discriminator network distinguishes SDFs and determines if the SDF is generated from the SDF-generator network or if it is the real SDF from the dataset. Learning rates with the values of 10^{-5} , 10^{-3} , 10^{-3} are used during training for the discriminator network, the generator network, and the encoder network, respectively. Adam optimizer with $\beta = 0.5$ is used for optimization purposes. Our network is trained separately on each class of objects, with a batch size of 64. We assign 1 and 100 to γ_1 and γ_2 respectively during the experiment. The proposed 3D-VAE-SDFGAN is trained on the ShapeNet dataset with chair, table, car, lamp, sofa, and cabinet categories for 2,000 epochs. The following loss functions are used to update our encoder, generator, and discriminator networks parameters during training.

$$L_E = \gamma_1 L_{KL} + \gamma_2 L_R \quad (9)$$

$$L_G = \log(1 - D(G(z))) + \log(1 - D(G(z_{img}))) + \gamma_2 L_R \quad (10)$$

$$L_D = L_{SDFGAN} \quad (11)$$

The parameters of the discriminator network are updated when the accuracy is less than 0.8 in each batch.

4.2. Performance evaluation

In this paper, we proposed a framework which can infer a 3D shape from its associated 2D image, named 3D VAE-SDFGAN. Figure 3 shows the results of our proposed model, which learns to build SDFs of the chair, table, car, lamb, and sofa from their associated 2D images. Figure 3(a) shows an example of a 2D image used to build SDFs for our proposed model, and Figure 3(b) shows the SDFs built from its associated 2D image. We extracted the triangular surfaces from SDFs with the marching cubes algorithm [41] and the surfaces were smoothed using Laplacian smoothing. The generated chair, table, car, and sofa from its SDFs are shown in Figure 3(c). We compared our 3D-VAE-SDFGAN with several state-of-the-art models and also followed the evaluation procedure in [42] by converting the volumetric results of 3D-R2N2 using the Marching cubes algorithm to a mesh model to compare the quality of the model generated with our proposed

solution. The chamfer distance (CD) [42] between the generated and ground truth 3D mesh is used as an evaluation parameter to assess the accuracy of our model. Table 1 shows the result of the evaluation.

We computed the CD for six categories in Table 1 to evaluate the performance of generated objects. From the experiment results, our proposed 3D-VAE-SDFGAN outperforms the state-of-the-art methods in all categories except the chair category in the MeshSDF [43] model and the car category in DISN [34] model. The performance in the chair category of the MeshSDF model occurs due to the adoption of the continuous model expressed in terms of how signed distance function perturbations impact surface geometry. The MeshSDF method encourages appealing results in a reconstruction task, which does not promote the creation of a new instance of an object like our 3D-VAE-SDFGAN. Also, our 3D-VAE-SDFGAN model outperformed the MeshSDF model in other evaluated object categories. The performance in the car category of the DISN model occurred because of the combination of global image features and local features at the projected location for each 3D point used in their model. Also, the DISN model combined two decoders to generate the SDF in their work. However, the model only performed reconstruction tasks and not generation tasks. Our 3D-VAE-SDFGAN CD results are better than DISN CD results in the remaining object categories evaluated.

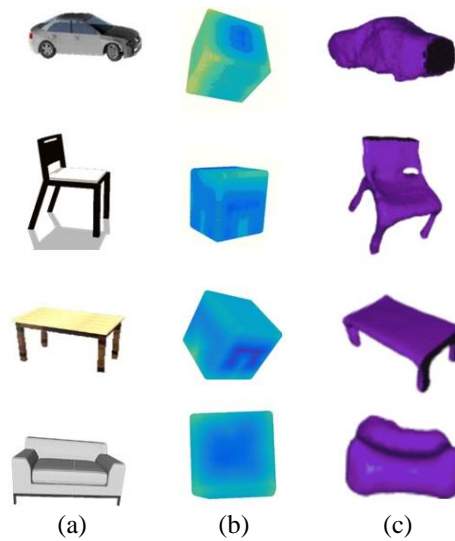


Figure 3. Results of our proposed model (a) examples of the 2D images and (b) SDFs of objects, and (c) the generated mesh-based

Table 1. Performance comparison of the proposed method with state-of-the-arts on ShapeNet dataset

Category	3D-R2N2 [38]	SIF [43]	N3MR [44]	MeshSDF [45]	DISN [35]	Ours
Chair	1.432	1.540	2.084	0.590	0.754	0.746
Table	1.116	1.570	2.383	1.070	1.329	0.710
Car	0.845	1.080	2.298	0.960	0.492	0.736
Lamb	4.009	3.420	3.013	1.490	2.273	0.767
Sofa	1.135	0.800	3.512	0.780	0.871	0.644
Cabinet	0.750	1.100	2.555	0.780	1.130	0.499
Average	1.548	1.585	2.641	0.945	1.142	0.684

Our proposed 3D-VAE-SDFGAN achieves the best average CD score because our model learns to generate new data similar to the existing data. With the generative power of our proposed model, it manages to create an object with details that are not possible with 3D-R2N2 with a different model's views [38]. In addition, our model can generate an object with various topologies that were lacking in the neural 3D mesh renderer (N3MR) approach [44]. Furthermore, our proposed model can represent the detailed structure of an object with SDF data representation which was one of the main problems associated with the structured implicit functions (SIF) method [43]. Figure 4 shows the qualitative comparison of our proposed model generated samples and learned representations with other 3D GAN-based models for visual evaluation. The chair and table in Figure 4(a), and the car, chair, and table in Figures 4(b) to (e) were duplicated and displayed for comparison. Smooth mesh surfaces recovered from signed distance function fields powered by the VAE-GAN framework and 2D image features in our work obtain higher-qualitative performance. We

compared the results of our simple but efficient proposed model in Figure 4(f) with the results of Jiang and Marcus [23] in Figure 4(d). Our proposed model, trained in an end-to-end manner, manages to produce appealing results with a low-cost 2D image, while Jiang and Marcus [23] model is a computationally expensive two-stage model implementation. The first model (low-frequency generator (LFG) network) generates SDF and passes it through a low-filter network to reduce noise in the high-frequency domain. The output was later used as an input in the second model (high-frequency generator (HFG) network) to generate high-resolution SDF. These processes are time-consuming and computationally expensive.

Also, the generated output of our proposed model in Figure 4(f) is comparable with the results of Kingkan and Hashimoto [24] in Figure 4(e), despite the complete 3D data used in their work. It implies that our low-cost 2D images, readily available with corresponding SDFs, are better dataset options for training. Moreover, the combination of 2D images with the SDFs dataset aids a better mesh-based 3D shape generation with a 3D VAE-SDFGAN model compared to more expensive 3D point-cloud data that requires specialized algorithms to make it fit for CNN-based architecture.

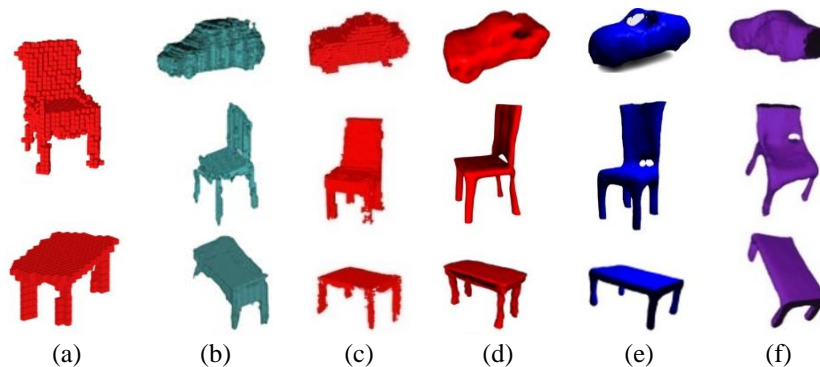


Figure 4. Qualitative comparison of our proposed model generated samples and learned representations with other 3D GAN-based models for visual evaluation (a) a sample of chair and table generated by Smith and Meger [15], and the sample of cars, chairs, and tables generated by (b) Zhu *et al.* [14], (c) Wu *et al.* [13], (d) Jiang and Marcus [23], (e) Kingkan and Hashimoto [24], and (f) by our model

5. CONCLUSION

In this paper, an architecture for learning 2D image latent spaces and mapping them to 3D shapes is proposed. The corresponding object's signed distance function (SDF) is created directly from the 2D image. We showed that the 2D images' latent space influences the network's performance and determines how much information can be transmitted from the 2D image encoder network to the SDF-generator network. Our proposed 3D VAE-SDFGAN model manages to successfully generate the corresponding SDF with smooth surface attributes using features from the 2D image. The experiment results show that the proposed model outperforms other existing state-of-the-art methods quantitatively and qualitatively.

ACKNOWLEDGEMENT

The research in this work was supported by Telekom Malaysia Research and Development under grant number RDTG/221045 and Multimedia University Graduate Research Assistant Scheme No. MMUI/190044.

REFERENCES





- [1] S. Chaudhuri, E. Kalogerakis, L. Guibas, and V. Koltun, "Probabilistic reasoning for assembly-based 3D modeling," *ACM Transactions on Graphics*, vol. 30, no. 4, pp. 1–10, Jul. 2011, doi: 10.1145/2010324.1964930.
- [2] M. Averkiou, V. G. Kim, Y. Zheng, and N. J. Mitra, "ShapeSynth: Parameterizing model collections for coupled shape exploration and synthesis," *Computer Graphics Forum*, vol. 33, no. 2, pp. 125–134, May 2014, doi: 10.1111/cgf.12310.
- [3] S. Chaudhuri and V. Koltun, "Data-driven suggestions for creativity support in 3D modeling," *ACM Transactions on Graphics*, vol. 29, no. 6, pp. 1–10, Dec. 2010, doi: 10.1145/1882261.1866205.
- [4] I. Goodfellow *et al.*, "Generative adversarial networks," *Communications of the ACM*, vol. 63, no. 11, pp. 139–144, 2020.
- [5] A. B. L. Larsen, S. K. Sonderby, H. Larochelle, and O. Winther, "Autoencoding beyond pixels using a learned similarity metric," *arXiv preprint arXiv:1512.09300*, Dec. 2015.
- [6] A. Dosovitskiy and T. Brox, "Generating images with perceptual similarity metrics based on deep networks," *arXiv preprint arXiv:1602.02644*, Feb. 2016.

- [7] C. Kim, H. Lee, and H. Jung, "Fruit tree disease classification system using generative adversarial networks," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 11, no. 3, pp. 2508–2515, Jun. 2021, doi: 10.11591/ijece.v11i3.pp2508-2515.
- [8] W. J. Hadi, S. M. Kadhem, and A. R. Abbas, "Fast discrimination of fake video manipulation," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 12, no. 3, pp. 2582–2587, Jun. 2022, doi: 10.11591/ijece.v12i3.pp2582-2587.
- [9] M. Berrahal and M. Azizi, "Optimal text-to-image synthesis model for generating portrait images using generative adversarial network techniques," *Indonesian Journal of Electrical Engineering and Computer Science (IJECS)*, vol. 25, no. 2, pp. 972–979, Feb. 2022, doi: 10.11591/ijeecs.v25.i2.pp972-979.
- [10] S. Pallavi, "Suggestive GAN for supporting dysgraphic drawing skills," *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 8, no. 2, pp. 132–143, Jun. 2019, doi: 10.11591/ijai.v8.i2.pp132-143.
- [11] A. Karthik, J. Shetty, S. G., and R. Dev, "Implementation of generative adversarial networks in HPCC systems using GNN bundle," *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 10, no. 2, pp. 374–381, Jun. 2021, doi: 10.11591/ijai.v10.i2.pp374-381.
- [12] Z. Iklima, T. M. Kadarina, and E. Ihsanto, "Realistic image synthesis of COVID-19 chest X-rays using depthwise boundary equilibrium generative adversarial networks," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 12, no. 5, pp. 5444–5454, Oct. 2022, doi: 10.11591/ijece.v12i5.pp5444-5454.
- [13] J. Wu, C. Zhang, T. Xue, B. Freeman, and J. Tenenbaum, "Learning a probabilistic latent space of object shapes via 3D generative-adversarial modeling," *Advances in neural information processing systems*, vol. 29, 2016.
- [14] J. Zhu, J. Xie, and Y. Fang, "Learning adversarial 3D model generation With 2D image enhancer," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, Apr. 2018, doi: 10.1609/aaai.v32i1.12223.
- [15] E. Smith and D. Meger, "Improved adversarial systems for 3D object generation and reconstruction," *arXiv preprint arXiv:1707.09557*, Jul. 2017.
- [16] C.-L. Li, M. Zaheer, Y. Zhang, B. Póczos, and R. Salakhutdinov, "Point cloud GAN," *arXiv preprint arXiv:1810.05795*, Oct. 2018.
- [17] G. Yang, X. Huang, Z. Hao, M.-Y. Liu, S. Belongie, and B. Hariharan, "PointFlow: 3D point cloud generation with continuous normalizing flows," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, Oct. 2019, pp. 4540–4549, doi: 10.1109/ICCV.2019.00464.
- [18] W. Zhirong *et al.*, "3D ShapeNets: A deep representation for volumetric shapes," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2015, pp. 1912–1920, doi: 10.1109/CVPR.2015.7298801.
- [19] M. Tatarchenko, A. Dosovitskiy, and T. Brox, "Octree generating networks: efficient convolutional architectures for high-resolution 3D outputs," in *2017 IEEE International Conference on Computer Vision (ICCV)*, Oct. 2017, pp. 2107–2115, doi: 10.1109/ICCV.2017.230.
- [20] M. Zhang and Y. Zheng, "Hair-GAN: Recovering 3D hair structure from a single image using generative adversarial networks," *Visual Informatics*, vol. 3, no. 2, pp. 102–112, Jun. 2019, doi: 10.1016/j.visinf.2019.06.001.
- [21] P. Achlioptas, O. Diamanti, I. Mithiagkas, and L. Guibas, "Learning representations and generative models for 3D point clouds," *arXiv preprint arXiv:1707.02392*, Jul. 2017.
- [22] T. Groueix, M. Fisher, V. G. Kim, B. C. Russell, and M. Aubry, "A papier-mache approach to learning 3D surface generation," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2018, pp. 216–224, doi: 10.1109/CVPR.2018.00030.
- [23] C. M. Jiang and P. Marcus, "Hierarchical detail enhancing mesh-based shape generation with 3D generative adversarial network," *arXiv preprint arXiv:1709.07581*, Sep. 2017.
- [24] C. Kingkan and K. Hashimoto, "Generating mesh-based shapes from learned latent spaces of point clouds with VAE-GAN," in *2018 24th International Conference on Pattern Recognition (ICPR)*, Aug. 2018, pp. 308–313, doi: 10.1109/ICPR.2018.8546232.
- [25] H. Huang, E. Kalogerakis, and B. Marlin, "Analysis and synthesis of 3D shape families via deep-learned generative models of surfaces," *Computer Graphics Forum*, vol. 34, no. 5, pp. 25–38, Aug. 2015, doi: 10.1111/cgf.12694.
- [26] J. Li, K. Xu, S. Chaudhuri, E. Yumer, H. Zhang, and L. Guibas, "GRASS," *ACM Transactions on Graphics*, vol. 36, no. 4, pp. 1–14, Jul. 2017, doi: 10.1145/3072959.3073637.
- [27] J. Li, C. Niu, and K. Xu, "Learning part generation and assembly for structure-aware shape synthesis," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 7, pp. 11362–11369, Apr. 2020, doi: 10.1609/aaai.v34i07.6798.
- [28] C. Zou, E. Yumer, J. Yang, D. Ceylan, and D. Hoiem, "3D-PRNN: generating shape primitives with recurrent neural networks," in *2017 IEEE International Conference on Computer Vision (ICCV)*, Oct. 2017, pp. 900–909, doi: 10.1109/ICCV.2017.103.
- [29] T. Furuya, W. Liu, R. Ohbuchi, and Z. Kuang, "Hyperplane patch mixing-and-folding decoder and weighted chamfer distance loss for 3D point set reconstruction," *The Visual Computer*, Sep. 2022, doi: 10.1007/s00371-022-02652-6.
- [30] E. Balashova, V. Singh, J. Wang, B. Teixeira, T. Chen, and T. Funkhouser, "Structure-aware shape synthesis," in *2018 International Conference on 3D Vision (3DV)*, Sep. 2018, pp. 140–149, doi: 10.1109/3DV.2018.00026.
- [31] M. Gadelha, S. Maji, and R. Wang, "3D shape induction from 2D views of multiple objects," in *2017 International Conference on 3D Vision (3DV)*, Oct. 2017, pp. 402–411, doi: 10.1109/3DV.2017.00053.
- [32] R. Wu, Y. Zhuang, K. Xu, H. Zhang, and B. Chen, "PQ-NET: a generative part Seq2Seq network for 3D shapes," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2020, pp. 826–835, doi: 10.1109/CVPR42600.2020.00091.
- [33] Z. Zheng, T. Yu, Q. Dai, and Y. Liu, "Deep implicit templates for 3D shape representation," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2021, pp. 1429–1439, doi: 10.1109/CVPR46437.2021.00148.
- [34] Q. Xu, W. Wang, D. Ceylan, R. Mech, and U. Neumann, "DISN: deep implicit surface network for high-quality single-view 3D reconstruction," *arXiv preprint arXiv:1905.10711*, May 2019.
- [35] S.-L. Liu, H.-X. Guo, H. Pan, P.-S. Wang, X. Tong, and Y. Liu, "Deep implicit moving least-squares functions for 3D reconstruction," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2021, pp. 1788–1797, doi: 10.1109/CVPR46437.2021.00183.
- [36] C. Doersch, "Tutorial on variational autoencoders," *arXiv preprint arXiv:1606.05908*, Jun. 2016.
- [37] A. X. Chang *et al.*, "ShapeNet: an information-rich 3D model repository," *arXiv preprint arXiv:1512.03012*, Dec. 2015.
- [38] C. B. Choy, D. Xu, J. Gwak, K. Chen, and S. Savarese, "3D-R2N2: a unified approach for single and multi-view 3D object reconstruction," in *Computer Vision ECCV 2016*, Springer International Publishing, 2016, pp. 628–644.
- [39] S. Osher and R. Fedkiw, "Signed distance functions," in *Applied Mathematical Sciences*, Springer New York, 2003, pp. 17–22.
- [40] S. J. Wetzell, "Unsupervised learning of phase transitions: From principal component analysis to variational autoencoders," *Physical Review E*, vol. 96, no. 2, Aug. 2017, doi: 10.1103/PhysRevE.96.022140.





- [41] T. Lewiner, H. Lopes, A. W. Vieira, and G. Tavares, "Efficient implementation of marching cubes' cases with topological guarantees," *Journal of Graphics Tools*, vol. 8, no. 2, pp. 1–15, Jan. 2003, doi: 10.1080/10867651.2003.10487582.
- [42] N. Wang, Y. Zhang, Z. Li, Y. Fu, W. Liu, and Y.-G. Jiang, "Pixel2Mesh: generating 3D mesh models from single RGB images," in *Computer Vision ECCV 2018*, Springer International Publishing, 2018, pp. 55–71.
- [43] K. Genova, F. Cole, D. Vlasic, A. Sarna, W. Freeman, and T. Funkhouser, "Learning shape templates with structured implicit functions," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, Oct. 2019, pp. 7153–7163, doi: 10.1109/ICCV.2019.00725.
- [44] H. Kato, Y. Ushiku, and T. Harada, "Neural 3D mesh renderer," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2018, pp. 3907–3916, doi: 10.1109/CVPR.2018.00411.
- [45] E. Remelli *et al.*, "MeshSDF: differentiable iso-surface extraction," *arXiv preprint arXiv:2006.03997*, Jun. 2020.

BIOGRAPHIES OF AUTHORS







Ebenezer Akinyemi Ajayi     is a Ph.D. student in the Faculty of Information, Science and Technology, Multimedia University (MMU), Malaysia, and a senior lecturer with the Computer Science Department, Kebbi State Polytechnic Dakingari, Nigeria. Before that, he received his Master of Technology degree from the Federal University of Technology, Akure, Ondo State, Nigeria in 2014. His research interests include machine learning, computer vision, 3D deep learning, computer graphics, and cyber-security. He can be contacted at 1161403835@student.mmu.edu.my and ebeseun@gmail.com.







Kian Ming Lim     received B.IT. (Hons) in Information Systems Engineering, Master of Engineering Science (M.Eng.Sc.), and Ph.D. (I.T.) degrees from Multimedia University. He is currently a senior lecturer at the Faculty of Information Science and Technology, Multimedia University. His research and teaching interests include machine learning, deep learning, computer vision, and pattern recognition. He can be contacted at kmlim@mmu.edu.my.



Siew-Chin Chong     is a senior lecturer at the Faculty of Information Science and Technology, Multimedia University Malaysia (MMU), and Program Coordinator of B.IT. Security Technology program. She graduated from MMU with First Class Degree honour of Bachelor of Information Science and Technology (Software Engineering) in 2003 and Master of Science (Information Technology) in 2006. She obtained her degree of Doctor of Philosophy (Information Technology) in 2018. Her research interests are machine learning and biometric security. She can be contacted at chong.siew.chin@mmu.edu.my.



Chin Poo Lee     is a senior lecturer in the Faculty of Information Science and Technology at Multimedia University, Malaysia. She completed her Master of Science and Ph.D. in Information Technology in the area of abnormal behavior detection and gait recognition. Her research interests include action recognition, computer vision, gait recognition, and deep learning. She can be contacted at cplee@mmu.edu.my.