Bulletin of Electrical Engineering and Informatics
Vol. 14, No. 6, December 2025, pp. 4988~5011
ISSN: 2302-9285, DOI: 10.11591/eei.v14i6.10755

Advances in artificial intelligence-driven 3D model generation: a review of GAN and VAE methodologies

Shyngys Adilkhan[1], Madina Alimanova[1], Lei Shi[2], Aiganym Soltiyeva[1]
[1]Department of Information Systems, School of Information Technologies and Applied Mathematics, SDU University, Kaskelen, Kazakhstan
[2]School of Computing, Faculty of Science, Agriculture and Engineering, Newcastle University, Newcastle upon Tyne, United Kingdom

Article Info

Article history:

Received May 27, 2025
Revised Sep 5, 2025
Accepted Sep 27, 2025

Keywords:

3D model generation
3D reconstruction
Artificial intelligence
Generative adversarial networks
Latent space
Polygon mesh
Variational autoencoder

ABSTRACT

This paper offers a comprehensive review of current developments in artificial intelligence (AI)-based 3D model creation, with an emphasis on techniques utilizing variational autoencoders (VAEs) and generative adversarial networks (GANs). 3DGAN, paired 3D model generation with GAN, conditional GAN, FaceVAE, voxel-based 3D object reconstruction, and 3D-VAE-SDFRaGAN are the six main techniques that are studied in this work. Each method is discussed, highlighting its architectural framework, data representation, and specific approach to generating 3D models. First, the paper introduces basic terms and classical 3D modeling techniques and provides a comparative analysis of them based on their workflow, purpose and field of application. In subsequent chapters, methods for generating 3D models based on the use of GANs and VAEs are reviewed, describing its methodology, experimentation technique, results, and comparison with other methods. The review outlines the strengths and limitations of each approach and their applications in object reconstruction, shape generation, and maintaining model consistency. It concludes by emphasizing how AI-driven methods can advance 3D modeling, underscoring the need for further research to enhance quality, control, and training reliability. The findings show AI's significant impact on automating complex modeling tasks and enabling new creative opportunities in 3D content development.

Corresponding Author:

Shyngys Adilkhan
Department of Information Systems, School of Information Technologies and Applied Mathematics
SDU University
Kaskelen, Kazakhstan
Email: shyngys.adilkhan@sdu.edu.kz

1. INTRODUCTION

The field of computer graphics enables the generation and manipulation of visual content using digital computing resources, facilitating the creation of interactive, immersive, and photorealistic environments across diverse domains [1], [2]. In entertainment, automated adaptive lighting enhances dramatic and aesthetic effects in real-time interactive media [3], while advanced facial animation techniques bring lifelike expression to virtual characters [4]. Mobile graphics and augmented-reality frameworks further extend these interactive experiences to handheld and wearable devices [5], and shaded-graphics methods laid the groundwork for realistic rendering in cinematic and training simulators [6].

In scientific research, high-fidelity 3D visualization and virtual environment simulations support medical diagnosis and treatment planning [7], and visualization techniques originally developed for entertainment have been repurposed for molecular and engineering analysis [8], [9]. Industrial design

applications leverage high-performance graphics workstations and computer-aided design (CAD) integration to streamline prototyping and manufacturing workflows [10], as well as immersive virtual environments for professional training and simulation [11]. Despite sharing core technologies—such as 3D visualization, real-time rendering, and virtual environments—each field adapts these tools to meet its unique performance, fidelity, and usability requirements [2].

Advances in 3D graphics enable scenes to be rendered in full three-dimensional depth, interactive virtual environments via VR headsets, imaginative storytelling through animated films, and visualization of novel objects using sophisticated visual effects. The rapid spread of artificial intelligence (AI) application demonstrates a new perspective at the process of creating 3D objects, providing opportunities to increase the efficiency of working with 3D models. Therefore, today, in addition to traditional techniques and tools for creating 3D content, there is also a need to have skills in working with AI in the process of 3D modeling.

The use of AI to the field of 3D modeling is essential because it fits with the larger trend of AI changing a number of fields, including research, education, and the arts [12]. AI is already present in many modern devices, such as cameras and smartphones, and it will drastically change how people interact with technology while also opening up new creative and technological possibilities. AI improves productivity in 3D modeling by automating tedious activities and facilitating the construction of intricate designs that would be difficult or time-consuming to accomplish by hand. Standard methods are changing as a result of this integration, creating new opportunities for creativity and innovation. Before discussing AI-based 3D modeling, it is necessary to define 3D models precisely and explain the traditional methods used to create them prior to the incorporation of AI.

The urgency of this research lies in the exponential growth in demand for automated 3D content creation across fields such as augmented reality (AR), digital twin engineering, real-time simulation, medical imaging, and virtual manufacturing. As industries accelerate toward digitization and automation, the need for scalable and accurate 3D modeling solutions has become critical. Traditional manual modeling techniques, while effective, are increasingly inadequate for handling the vast volume and complexity of 3D data required in real-time applications. The recent advancement of generative AI, particularly variational autoencoders (VAEs), generative adversarial networks (GANs), and diffusion models, has opened promising opportunities for addressing these challenges through automated and data-driven modeling frameworks [13], [14]. Several recent studies underscore the transformative impact of such deep generative approaches in reducing modeling time and improving the fidelity of 3D reconstructions [15]-[18]. From 2020–2025, several studies report concrete gains from GAN/VAE families in 3D generation, especially in medical imaging: 3D-StyleGAN variants improve FID, bMMD², and MS-SSIM at 1–2 mm resolutions [19], domain-adaptive VAE improves SSIM/PSNR/Dice over baselines [20], and VQ-VAE exceeds GAN/Disc-VAE on SSIM/PSNR for synthetic augmentation [21]. In creative engineering and gaming, hybrid GAN+implicit/NeRF pipelines show better generative quality while still facing artifacts and mesh-diversity limits [22]. Across domains, the literature flags non-standardized benchmarks (outside medicine) and limited external validation, further motivating a consolidated comparative review [23]. Therefore, this review is urgently needed to consolidate ongoing developments, evaluate comparative strengths and limitations, and guide future innovation in this rapidly evolving research area.

In this article we aim to answer the following research questions:

Q1: What are the key advancements in using GANs and VAEs for 3D model generation?

Q2: What are the key datasets and evaluation metrics used in GAN- and VAE-based 3D model generation studies, and how do they influence the performance of these models?

Q3: How does VAE-based models compare to GAN-based models in generating 3D models, particularly in terms of computational efficiency?

This paper is structured as follows: section 2 provides an overview of traditional 3D modeling techniques. In section 3, the methodology employed in this study is detailed. Section 4 offers an overview of GAN- and VAE-based approaches for 3D model generation. Section 5 delves into a comparative analysis of the reviewed methods, highlighting their strengths and weaknesses. Section 6 identifies the challenges and potential avenues for future research in this field. Section 7 contains results and discussion part. Finally, section 8 summarizes the key findings and conclusions of the study.

2. BACKGROUND

2.1. Traditional 3D modeling techniques

3D model is a 3 dimensional figure which represents a certain object or scene and consists of components such as vertices, edges and polygons [24]. These models are used in various fields such as animation, game development, architectural reconstruction, engineering, virtual reality, simulation, 3D printing, and visualization. They can represent both real-world objects and imaginary concepts. The classic method of building 3D models implies the use of special software like 3Ds Max, Maya, and Blender. At first,

user has to learn certain tools and operations, which change the structure and shape of the model and directly affect on a final result. Often 3D modelers or digital artists use reference images as a base for future 3D model. They may consider overall shape or some specific details from the image and recreate them in a 3D space. First, artists create the concept art of the character, object or a scene. After that 3D modeler recreate it. This method is used almost in every industry related to 3D graphics: for creating models for games, animated movies, visual effects, and motion design. In the age of emergence of AI, this workflow may be reinterpreted by using different tools and technologies, but keeping the main idea–creating a 3D model based on image.

2.1.1. Box modeling

Box modeling is a foundational technique in 3D modeling that involves utilizing traditional tools to construct shapes from elementary geometric items, such as cubes or spheres [25]. The process commences with selecting a base shape characterized by a low polygon count. Because it gives modelers precise control over the shape's various elements—faces, edges, and vertices—this method is highly acclaimed for its accuracy. Box modeling differs from other modeling approaches due to its mechanical character. Instead of initially concentrating on little details, it focuses on the manipulation of entire shapes and larger portions of the model. Using quads, or four-sided faces, is a crucial part of box modeling. These are preferred because they work with the majority of modeling software, which is primarily made to manage quads effectively. In spite of this choice, models are often triangulated to guarantee stability and compatibility with different rendering engines, either manually by the user or automatically by the program in the background. Hard-surface objects, like those in manufactured goods and architectural visualizations, are especially well-suited for box modeling. The tools frequently used in the process, including as extrusion, loop cuts, and beveling, are responsible for the technique's effectiveness in these areas. These tools make it easier to transform basic shapes into intricate, accurate forms. Furthermore, to increase the model's level of detail, box modeling is frequently used with subdivision surface techniques. The process of creating subdivision surfaces entails introducing new geometry between the preexisting faces, vertices, and edges. This geometry can then be altered using common modeling tools. The modeler can more precisely and flexibly mold the object's divided form thanks to this extra geometry, which serves as a control cage as shown in Figure 1.

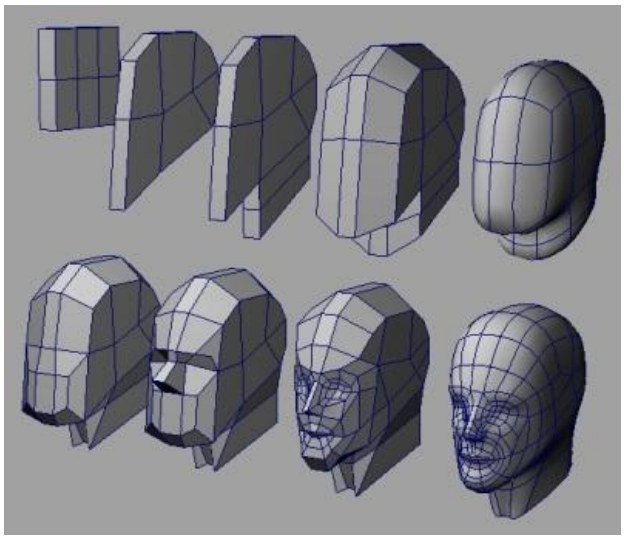

Figure 1. Box modeling of a head

2.1.2. Polygon modeling

Polygon modeling is a technique that involves starting with a vertex and gradually building the model edge by edge until the entire surface or object is complete. This process is similar to box modeling, as it often includes a subdivision process to smooth or even out the geometry of the object being modeled. It is particularly useful for visualizing 3D models of intricate designs such as statues and ornaments, where precise detail is essential [25]. The flexibility of polygon modeling allows for the creation of unique, organic-looking designs, making it a popular choice in scenarios requiring high levels of complexity and detail [25].

Like box modeling, polygon modeling typically emphasizes the use of quads in the topology, as many tools are designed to function within a quad-based framework. Using quads in 3D models is important because

quads provide predictable and even deformation, which is essential for objects that will undergo organic deformation, like moving muscles. Quads squash and stretch uniformly, making them ideal for character modeling. However, for non-deforming objects, such as video game props, quads are less critical, and factors like poly count may take precedence. Additionally, in software like Maya, using triangles as illustrated in Figure 2, can complicate adding edge loops needed to improve deformation in certain areas, making quads more advantageous for character animation.

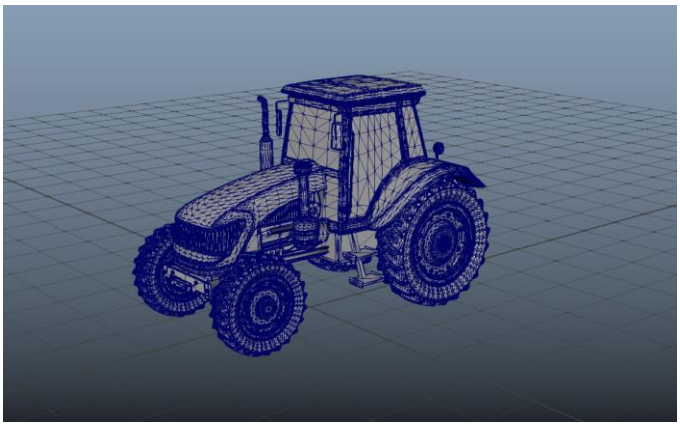

Figure 2. Triangulated polygons of the tractor model

2.1.3. Non-uniform rational b-splines and curve modeling

Figure 3 illustrates the non-uniform rational b-splines (NURBS) and curve modeling technique, which uses control points to define curves that generate surfaces rather than directly manipulating vertices and edges. Using operations like scaling, moving, and rotating the control points, this method makes it possible to create curved, smooth 3D models. The method offers more control over the model's resolution and detail and is especially beneficial for producing more consistent and organic-looking [25] characters. This technique is often employed in software and engineering design, especially in CAD settings where smooth and accurate surfaces are needed.

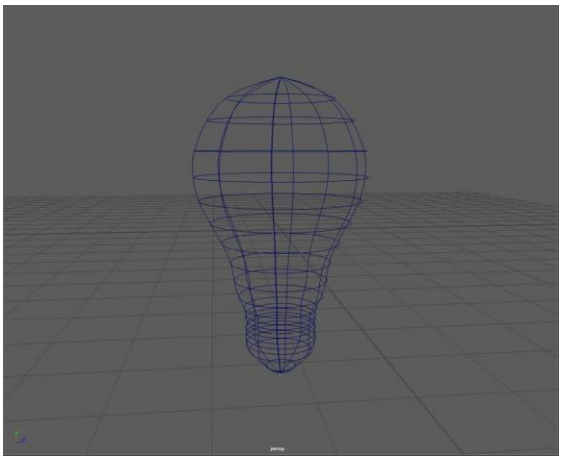

Figure 3. NURBS modeling

2.1.4. 3D sculpting

Digital sculpting offers a more fluid and artistic technique. It is similar to traditional clay sculpting method and differs from polygonal modeling, which is about the manipulation of polygons, edges, and vertices [26]. Polygonal modeling is the ideal choice for objects that need accurate measurements and defined geometries, such mechanical components, architectural designs, and hard-surfaced environments, because it

requires meticulous adjustments to each polygon, edge, and vertex. The models produced by this technique are well known for their fine features and crisp edges, which are signs of the structural integrity that polygonal modeling ensures. Digital sculpting, on the other hand, is a workflow that lets artists work with digital material in a number of ways, including pinching, pushing, pulling, and carving, to produce naturally curved shapes as shown in Figure 4. This method effectively produces complex, organic structures that would be challenging to build with traditional polygonal procedures. When making realistic humans, inventive animals, and other shapes that benefit from a flowing, organic style, it works particularly well. The sophisticated features of digital sculpting tools allow artists to add fine details and textures, such as skin pores, wrinkles, and other surface anomalies, creating models with exceptional realism and complexity.

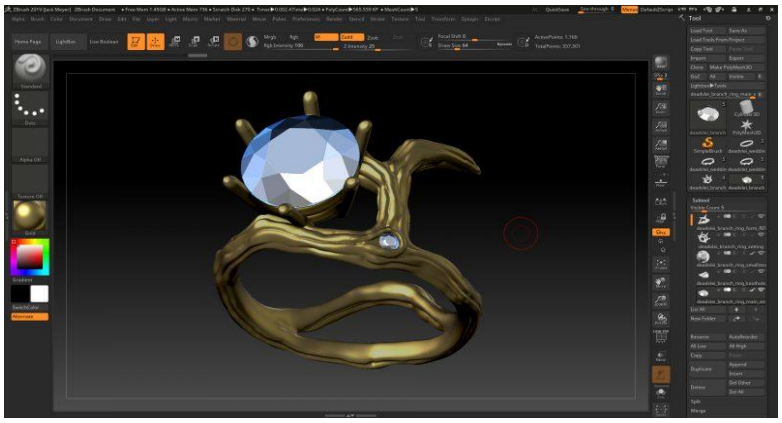

Figure 4. Sculpting in Zbrush

2.1.5. Procedural modeling

Procedural modeling is an automated technique for generating three-dimensional models, allowing for dynamic modification without manual intervention. This method simplifies the modeling process by using a predetermined set of guidelines [26], algorithms, or directives. Procedural modeling works similarly to deformation systems in that it applies changes to a basic mesh inside a layered framework. Changes to the base mesh require recalculating overlaying operations, and subsequent modifications are piled on top of this foundation. Individual modeling phases can be independently adjusted thanks to this layered architecture, which doesn't impact other model elements as illustrated in the example shown in Figure 5.

Figure 5. Procedural modeling in 3Ds Max

There are various benefits to procedural modeling compared to manual approaches. It is efficient in automating the building of complex models and saving a significant amount of time. Because of the flexibility of editing, the created meshes can be dynamically altered and animated. Its modular design encourages adaptability since various parts can be changed separately. The limited control over particular elements is a major drawback, too, as the algorithmic nature can occasionally provide surprising outcomes.

2.1.6. Boolean modeling

In Boolean modeling, new shapes are created by adding or subtracting elements from an existing model [24]. This method is frequently paired with box modeling, which creates simple shapes that are subsequently merged to build more complicated objects. As illustrated in Figure 6, this procedure relies heavily on Boolean operations such as union, intersection, and difference. Whereas intersection retains only the overlapping portions, union unites two shapes, and difference subtracts one from the other. Compared to traditional modeling techniques, Boolean modeling greatly speeds up the construction of complex designs, particularly when mixing curved and hard-edged forms.

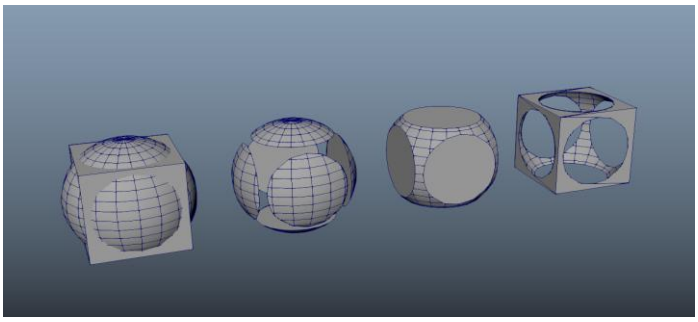

Figure 6. Boolean operations in Maya

In Table 1 the most common 3D modeling techniques are compared in terms of methodology, workflow, purpose, output, and usage. Also, specific programs were presented that most often use the indicated methods. Some of them like Maya or 3Ds Max are capable of providing an opportunity to use all the mentioned techniques in one project.

Table 1. Comparison of traditional modeling techniques comparative analysis of traditional techniques of 3D modeling

Ref	Technique	Methodology	Workflow	Purpose	Output	Usage	Software
[25]	Polygonal modeling	Creating models by manipulating vertices, edges, and faces to form polygons.	Build a base mesh, add details, and refine topology.	Creating hard-surface models, characters, and environments with precise control.	Polygonal mesh	Game development, animation, VFX, and 3D printing	Blender, Maya, 3ds Max, and Modo
[25]	NURBS modeling	Using mathematical equations to define curves and surfaces.	Create control points, manipulate curves, and refine surface.	Creating smooth, organic shapes, and precise industrial designs.	NURBS surface	Automotive design, architecture, shipbuilding, and animation	Rhino, Maya, SolidWork, and CATIA
[26]	3D sculpting	Digitally sculpting models using virtual tools.	Start with a basic shape, add details through sculpting, and refine form.	Creating organic shapes with high detail and character modeling.	High-resolution mesh	Character modeling, animation, VFX, and concept art	ZBrush, Sculptri, Blender, and Mudbox
[26]	Procedural modeling	Generating models based on rules, algorithms, and parameters.	Define rules and parameters, generate model, and iterate on parameters.	Creating complex and repetitive geometry, generating variations.	Procedural definition, generated mesh	Environment creation, game development, VFX, and architecture	Houdini, Blender, Unreal Engine, and Unity
[24]	Boolean modeling	Combining or subtracting shapes to create new forms.	Create base shapes, apply Boolean operations, and refine result.	Creating complex shapes from simple primitives and hard-surface modeling.	Combined or subtracted shapes	Industrial design, architecture, product design, and game development	Blender, Maya, 3ds Max, and SolidWorks

While traditional 3D modeling methods have been essential to fields like industrial design, gaming, and animation, novel advances in 3D generative AI are expanding the field by automating modeling tasks,

speeding up design procedures, and enabling anyone to create 3D content. To address the limitations of traditional methods in terms of time and expertise required to produce high-quality results, the integration of AI into 3D modeling has emerged as a promising solution to automate complex tasks, create highly detailed models, and explore new creative possibilities.

3. METHOD

3.1. Literature search strategy

A comprehensive literature search was conducted to identify and analyze scholarly articles focusing on 3D model generation techniques leveraging GANs and VAEs. Figure 7 demonstrates the flowchart of the literature review strategy employed in this study. The search focused on preprint repositories, conference proceedings, and peer-reviewed publications. To guarantee extensive coverage of the subject, academic databases including arXiv, IEEE Xplore, and Scopus were used. The goal of this multi-database strategy was to include both established research that was published in reputable journals and innovative work that was shared via preprints or conference proceedings.

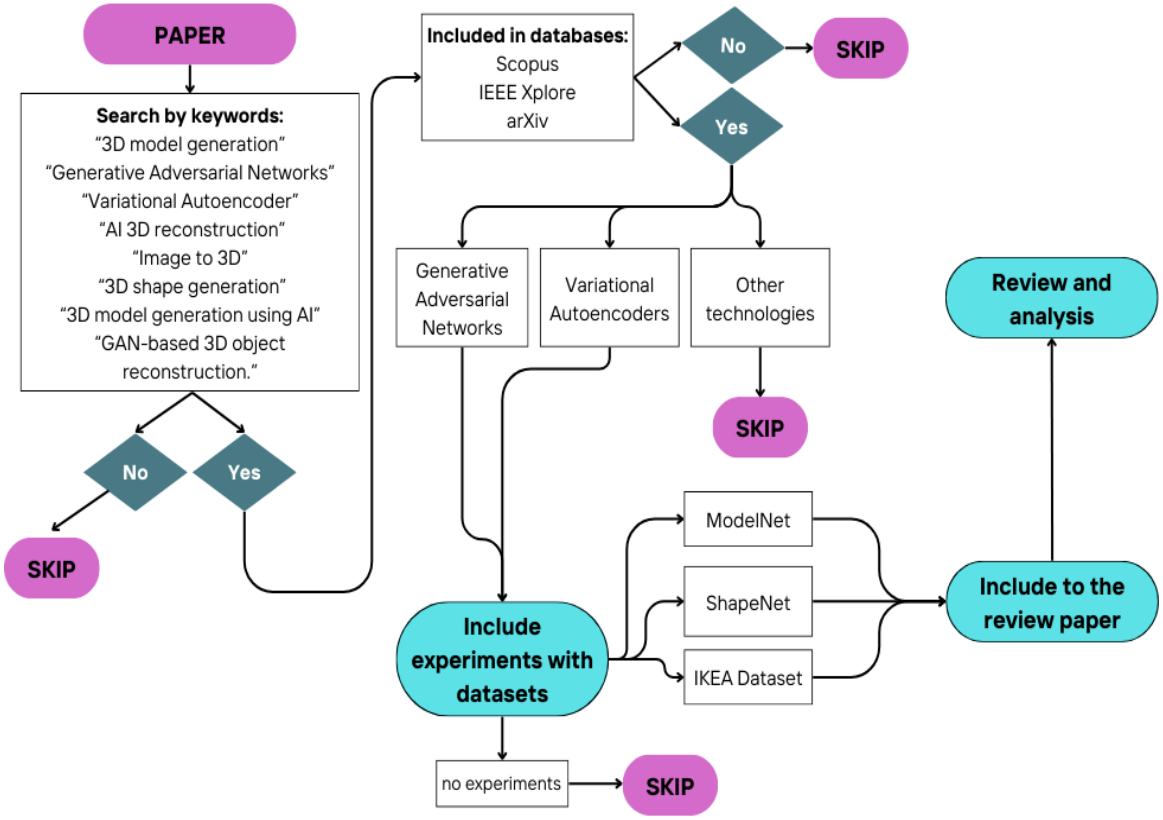

Figure 7. Flowchart of literature review strategy

Keywords like "3D model generation," "GANs," "Variational Autoencoder," "AI 3D reconstruction," "Image to 3D," "3D shape generation," "3D model generation using AI," and "GAN-based 3D object reconstruction" were carefully chosen and used both singly and in combination to retrieve relevant articles. In order to optimize the inclusion of papers pertaining to AI-based 3D model generation, these terms were modified and improved upon during preliminary searches. After the articles were first retrieved, duplicates were eliminated, and the remaining entries were filtered using abstracts and titles. At this point, studies that had no direct connection to the creation of 3D models, that only dealt with non-AI-based techniques, or that did not explicitly highlight GAN or VAE designs were excluded. A full-text review was then conducted on potentially relevant articles to confirm that they met the inclusion criteria.

3.2. Inclusion and exclusion criteria

To ensure the quality and relevance of the chosen studies, a set of predetermined inclusion and exclusion criteria was developed. Articles required to; i) investigate or suggest 3D model generation techniques utilizing GAN and VAE architectures, ii) be published in publications with a Scopus index, accessible on IEEE Xplore, or housed on arXiv, and iii) offer quantitative evaluation criteria for performance evaluation in order to be considered. Furthermore, the included research required to present or talk about tests carried out on popular 3D model datasets as the IKEA Dataset, ModelNet, and ShapeNet.

On the other hand, studies that; i) did not specifically address GAN or VAE architectures, ii) mainly dealt with non-AI-based 3D modeling techniques, and iii) did not give enough information about the training datasets, model configurations, or evaluation metrics used for performance assessment were excluded. Only the most relevant and methodologically sound studies were kept for additional assessment by implementing these criteria.

3.3. Data extraction

To assist in comparison analysis, important parameters were methodically retrieved and assessed for each chosen study. The GAN or VAE architecture, including network layers, activation functions, and other architectural modifications that can affect model performance, has to be fully described during this procedure. In order to appropriately assess the extent of each approach, information on the training datasets was also reviewed. Following that, the models' performance was assessed using established evaluation metrics such as chamfer distance, intersection over union (IoU), and other domain-specific benchmarks. Finally, reported results were compared, highlighting both advantages and limitations observed within the reviewed literature.

4. REVIEWED AI METHODS FOR 3D MODEL GENERATION

AI-driven methods that make use of GANs and VAEs have demonstrated encouraging results in producing 3D models. Rapid prototyping and modeling have been made easier by these techniques. The next sections will explore several 3D model generation techniques that use GANs and VAEs in their designs.

4.1. Generative adversarial networks

GANs are a type of deep learning model introduced by Goodfellow *et al*. [27] are designed to solve the generative modeling problem, aiming to learn the probability distribution of training data and generate new examples from it [27], [28]. This unsupervised learning technique utilizes two neural networks: a generator and a discriminator, which compete against each other [27], [29]. GANs have been used for a variety of tasks in research settings and have shown great success in producing realistic high-resolution images. In spite of their achievements, GANs pose distinct difficulties since they are game-theoretic, which sets them apart from previous optimization-based generative models [27]. GANs' architecture has changed over time, with many iterations created to cater to certain uses and boost efficiency [29].

4.2. 3DGAN

In order to create 3D objects, the authors suggest 3D-GAN [30], a framework that makes use of current developments in GANs and volumetric convolutional networks. Using an adversarial criterion is the main innovation of this concept, enabling the generator to synthesize 3D objects in a realistic and high-quality way while implicitly capturing their structure.

To represent an object in 3D voxel space, the generator G transfers a 200-dimensional latent vector z, randomly sampled from a probabilistic latent space, to a 64×64×64 cube. Then, a confidence value D(x) indicating whether a 3D object input (x) is synthetic or real is produced by the discriminator D. As a classification loss, authors used binary cross entropy and formulated adversarial loss function as (1):

$$x_{3D-GAN} = logD(x) + log(1 - D(G(z))) \quad (1)$$

In (1), *x* represents a real object in a space, while *z* is a randomly sampled noise vector from distribution p(z).

Inspired by the work of [31], the authors developed an all-convolutional neural network in which the generator is composed of five volumetric fully convolutional layers, each featuring 4×4×4 kernels and strides of 2, with batch normalization and ReLU activations included between layers, culminating in a Sigmoid layer. The discriminator mirrors this structure but substitutes Leaky ReLU for the ReLU activations.

During the training of the model authors faced with the problem of different learning speeds of the generator and discriminator. The generator was struggling with extracting signals for improvement from discriminator, since the discriminator was learning much faster and was way ahead than generator. Theoretically, this happened because of difficulty of generation of 3D object in a voxel space than distinguishing if the object is real or not. To solve this problem, the authors adapted training process, making

the discriminator updated only when its accuracy in the most recent batch is less than 80%. This approach improved training stability and yields better outcomes.

Following the idea of recovering 3D object from 2D image mapping, authors also introduced 3D-VAE-GAN [30] model, which consist of image encoder, decoder, and discriminator. Five spatial convolution layers with kernel sizes of {11, 5, 5, 5, 8} and strides of {4, 2, 2, 2, 1} make up the image encoder. In between are layers for batch normalization and ReLU, and at the end is a sampler that takes a sample of a 200-dimensional vector that the 3D-GAN uses. The loss function used is similar to VAE-GAN [32], consisting object reconstruction loss, cross entropy loss and KL divergence loss.

Table 2 demonstrates the superiority of 3D-GAN over previous state-of-the-art methods in unsupervised learning. Testing was performed using ModelNet [33] and the proposed method outperformed VConv-DAE model [34], T-L Network [35], LFD [36], and SPH [37] models learned without supervision. On ModelNet40, 3D-GAN outperformed SPH by 22.1%, LFD by 10.3%, T-L Network by 12.0%, and VConv-DAE by 10.3%. Speaking about ModelNet10, 3D-GAN outperformed SPH by 14.0%, LFD by 14.0%, and VConv-DAE by 13.0%.

Table 2. Comparison of 3D-GAN with previous methods in unsupervised learning [30]

Method	ModelNet40 accuracy (%)	Improvement over others (%)	ModelNet10 accuracy (%)	Improvement over others (%)
SPH [37]	68.2	22.1 (compared to SPH)	79.8	14.0 (compared to SPH)
LFD [36]	75.5	10.3 (compared to LFD)	79.9	14.0 (compared to LFD)
T-L Network [35]	74.4	12.0 (compared to T-L Network)		
VConv-DAE [34]	75.5	10.3 (compared to VConv-DAE)	80.5	13.0 (compared to VConv-DAE)
3D-GAN [30]	83.3	-	91.0	-

4.3. Paired 3D model generation with generative adversarial network

The paper presents a novel approach to generating paired 3D models using conditional generative adversarial networks (CGANs) [38]. The authors address the challenge of generating the same 3D object in different conditions, such as varying rotation angles, without altering the standard CGAN architecture.

The primary issue tackled is the inability of conventional CGANs to generate consistent object pairs under different conditions. CGAN offers a way to manipulate the features of generated data by providing specific conditions. However, despite being able to set these conditions, the resulting samples are still random. This makes it difficult to generate paired samples under different conditions. The core issue is that changing the condition while keeping the input constant doesn't produce consistent outputs. This is because the relationship between the input and output is complex and intertwined, meaning any alteration to the condition significantly impacts the final result.

The authors used a baseline CGAN model that they modified for 3D model production as a comparison for their suggested approach. The identical dataset and set of parameters were used to train both the baseline and the suggested approach [38]. Under varying conditions, the baseline CGAN produced completely different objects, as demonstrated by the quantitative and visual results. On the other hand, models consistent with all conditions were generated by the suggested approach.

To evaluate the results, authors introduced 2 different evaluation metrics: average absolute difference (AAD), which measures the difference between generated models and average voxel agreement ratio (AVAR), which measures the consistency of voxel grids between different conditions [38]. A lower AAD shows greater similarity between the models created under various situations and a higher AVAR reflects greater consistency and similarity in voxel structures between the models.

The results in Table 3 clearly demonstrate that the proposed method achieved significantly lower AAD values and higher AVAR values, indicating less difference and greater consistency. The suggested approach continued to provide significant advantages over the baseline CGAN even after the experiments were expanded to four conditions. The suggested approach outperformed the baseline, yielding more consistent models, even though the problem complexity increased with the number of conditions.

The authors suggested a novel approach to address this issue, which includes adding a training step to the CGAN framework. This step involves generating multiple samples with the same noise vector but different conditions and then merging these samples to enforce consistency. This additional step can be implemented without altering the CGAN design. Instead, it is incorporated as an additional training phase, which allows the method to be applied to any CGAN model.

The authors' approach, which entailed building 3D models of furniture like couches, beds, and chairs at different rotations, was validated using the ModelNet dataset [33]. 2 experiment types were conducted in this work which involved 2 conditions and 4 conditions.

Table 3. Comparison of baseline and proposed methods AAD and AVAR values of generated objects [38]

Category	Condition	Method	AAD	AVAR
Chair	2-Condition	Baseline	0.027	0.32
Chair	2-Condition	Proposed	0.009	0.79
Chair	4-Condition	Baseline	0.034	0.36
Chair	4-Condition	Proposed	0.024	0.61
Bed	2-Condition	Baseline	0.029	0.69
Bed	2-Condition	Proposed	0.012	0.89
Bed	4-Condition	Baseline	0.043	0.65
Bed	4-Condition	Proposed	0.021	0.82
Sofa	2-Condition	Baseline	0.018	0.74
Sofa	2-Condition	Proposed	0.004	0.95
Sofa	4-Condition	Baseline	0.034	0.62
Sofa	4-Condition	Proposed	0.013	0.90

4.4. Conditional generative adversarial network

The challenge of random and uncontrollable generations is the focus of another study on 3D model reconstruction using GAN, which was conducted by [39]. The authors specifically note that although GANs are capable of producing 3D models, it is difficult to produce particular models depending on desired circumstances because the outcomes are frequently random and uncontrollable. The generative process of GANs is usually uncontrollable, which leads to unanticipated model properties and frequently mode collapse, in which the generator generates a limited diversity of outputs. Furthermore, it is still difficult to effectively recreate 3D models from a single 2D image, particularly when occlusions or insufficient visual information are present.

The authors suggest a unique architecture that combines a 3D reconstruction network with CGANs in order to get through these limitations. While the reconstruction network uses a combination of conditional variational autoencoder (CVAE), GAN, and classifier components to accurately reconstruct 3D objects from single images, even under challenging conditions, the CGAN incorporates class information to produce more controlled and lifelike 3D models. Experiments were conducted using ModelNet10 dataset and they demonstrated high efficiency in generating realistic 3D models that match provided class labels [39].

The proposed method was compared with several existing approaches using the IKEA dataset [40], across categories such as “Bed,” “Bookcase,” “Chair,” “Desk,” “Sofa,” and “Table.” The comparison involved AlexNet-fc8 [35], AlexNet-conv4 [35], T-L Network [35], and both jointly and separately trained versions of 3D-VAE-GAN [30], and 3D-VAE-IWGAN [41]. The suggested approach outperformed the other methods by achieving the highest mean average precision of 70.9% across all categories. The 3D-VAE-IWGAN (separately trained) is the closest competitor, scoring 61.7%.

Table 4 summarizes the performance comparison between the proposed CGAN method and 3D-VAE-IWGAN across six furniture categories. With a remarkable mean average precision (mAP) of 86.8% in the “Bed” category, it outperformed the nearest rival, 3D-VAE-IWGAN (separately trained), which received a score of 77.7%. Likewise, with a mAP of 85.2%, the suggested approach outperformed the next best solution in the “Bookcase” category, which was just 51.8%. With a robust mAP of 60.3% for “Chair,” the approach maintained its lead over the second-best result of 56.2%. It held onto its lead in the “Desk” area, scoring 52.4%, just higher than the 50.6% of the rival approach. The suggested approach showed strong performance even though its mAP of 80.2% in the “Sofa” category was not far behind the 3D-VAE-IWGAN (separately trained) method. Lastly, the approach achieved a mAP of 60.1% in the “Table” category, surpassing the 3D-VAE-IWGAN (separately trained) method, which received a score of 52.6%. Overall, the suggested technique demonstrated its capacity to produce precise and excellent 3D models across a variety of object kinds by regularly outperforming or matching the best state-of-the-art methods.

Table 4. Comparison of the proposed CGAN and 3D-VAE-IWGAN in terms of mean average precision [39]

Category	Proposed method (mAP %)	3D-VAE-IWGAN (separately trained) (mAP %)
Bed	86.8	77.7
Bookcase	85.2	51.8
Chair	60.3	56.2
Desk	52.4	50.6
Sofa	80.2	82.0
Table	60.1	52.6

4.5. Variational autoencoders

VAEs [42] are a particular kind of autoencoder and generative model that are mostly utilized in unsupervised learning, that are often used in many different applications, including as data compression,

dimensionality reduction, and generating new data, like new images, and text. Several studies already integrated VAEs for 3D model reconstruction and demonstrated good results.

4.6. FaceVAE: generation of a 3d geometric object using variational autoencoders

FaceVAE [43] is a model that uses VAEs to build 3D geometric objects. The study focuses on face-based geometric data, especially polygons, which are frequently utilized in industrial settings. Through the direct use of polygon-based data, the research seeks to directly leverage existing 3D data production methods, which mostly rely on voxels, 2D pictures, and point clouds, to build more accurate and useful 3D models.

The absence of direct application of deep learning techniques to 3D data based on polygons, which is common in industrial settings, is the difficulty that the authors address [43]. The efficiency of traditional approaches is limited in practical applications because they concentrate on simpler data formats like point clouds and voxels, which are unable to adequately capture the complexity of 3D polygonal geometry.

The paper's methodology section outlines a methodical procedure for creating the faceVAE model. The first step in the process is data structurization, which involves employing adjacency matrices that are designed for learning to transform unstructured polygonal geometric data into a structured format. In order to prepare the data for the VAE model, this step is essential. Next, three types of adjacency matrices are proposed by the study: face adjacency (AF), which focuses on the relationships between faces that share edges with each other; vertex-face adjacency (AVF), which reflects the relationships between vertices and the faces they form; and vertex adjacency (AV), which captures the connections between vertices sharing an edge.

To help the VAE model learn, a feature matrix is built with the vertices' coordinates alongside these matrices. The vertice positions are normalized and recorded as binary data. The main component of the technique is the use of a modified VAE model for both encoding and decoding the structured data. This model consists of three hidden layers, each with 1,000 nodes. By first transforming the input into a latent space and then reconstructing it, this VAE model effectively handles the geometric data.

At last, the model reconstructs the geometric data from the latent space following the encoding and decoding procedure. In order to assess the model's effectiveness, the reconstructed data undergo voxelization, which turns them into a three-dimensional grid of voxels. This enables a comparison of the degree of similarity between the created and original geometries. Using the FaceVAE model, this structured method allows the direct production of 3D geometric data from polygonal representations.

4.6.1. Experiments

The ModelNet10 [33] dataset, which contains 3D geometric data of interior furniture, was employed in the experiments. The dataset is organized into ten categories, each of which contains geometric objects represented by vertex coordinates and face indices. Sixty-four geometric instances with fewer than 300 vertices were chosen as training data and sixteen instances as test data for the experiments. Given that every input node in the multilayer perceptron (MLP) employed in the study corresponds to a vertex or a feature, the choice was taken to match the model's capacity restrictions.

4.6.2. Training

During the training phase, the latent variable z was changed to various values (such as 2, 5, 10) and the impact on the model's capacity to learn and rebuild 3D geometry was monitored. The training losses for each kind of input data were compared. The outcomes demonstrated that learning adjacency matrices (AV and AVF) was superior to learning just vertex features (FV) in terms of effectiveness. Better performance was also seen when adjacency and feature information (AV || FV, AVF || FV) were combined, however significant overfitting was noticed, especially when using the AVF || FV combination.

4.6.3. Evaluation

Metrics including accuracy, precision, and recall were used to assess the model's performance. Since a high percentage of voxels in voxelized data are empty, precision and recall were thought to be more informative metrics than accuracy.

According to the results shown in Table 5, adjacency information was essential for efficient learning, with AV and AVF matrices considerably outperforming FV in terms of precision (72.32 and 63.48, respectively) and recall (69.86 and 59.78, respectively) on test data.

4.7. Voxel-based 3D object reconstruction from single 2D image using variational autoencoder

Another work employing VAE for 3D object reconstruction based on voxel, proposed by [44] addresses the challenge of correctly creating 3D objects out of a single 2D picture. Even with several images or substantial computer resources, traditional 3D reconstruction approaches are unable to generate smooth,

high-resolution 3D models [44]. A single 2D image contains just a limited amount of geometric information, making this challenge very difficult.

Table 5. 3D geometry generation performance [43]

G type	Training accuracy	Training precision	Training recall	Test accuracy	Test precision	Test recall
FV(z=5)	99.99	98.95	100.00	78.25	12.41	33.84
AV(z=5)	99.72	97.82	98.64	95.84	72.32	69.86
AVF(z=5)	100.00	100.00	100.00	94.08	63.48	59.78
AV‖FV(z=2)	93.20	7.44	8.24	92.54	0	0
AV‖FV(z=5)	88.92	49.01	71.27	81.80	10.91	8.63
AV‖FV(z=10)	98.61	88.26	96.91	82.06	8.63	16.28
AVF‖FV (z=2)	93.63	6.25	6.25	92.54	0	0
AVF‖FV (z=5)	93.72	61.94	80.14	80.25	11.53	27.78
AVF‖FV (z=10)	99.99	99.99	100.00	80.57	10.28	25.86

For voxel-based 3D object reconstruction (V3DOR) from a single 2D image, the authors propose two approaches: V3DOR-AE, which uses a simple autoencoder, and V3DOR-VAE, which uses a variational autoencoder. In order to create a corresponding 3D model, each model uses an encoder to extract a compressed latent representation from the 2D image. Specifically, the VAE-based method performs exceptionally well in creating new 3D models that closely resemble the features of the input data, exhibiting an improved ability to capture and reproduce minute details and variances.

The AE-based approach comprises an encoder that extracts salient features from the 2D image and a decoder that reconstructs a 3D model. The encoder's architecture incorporates convolutional layers adept at capturing intricate geometric details within the image. The decoder subsequently generates a 3D representation as a single-channel voxel volume, where each voxel encodes the presence or absence of material at a specific 3D location.

The VAE model expands upon the AE architecture by incorporating a stochastic element. Instead of producing a fixed latent representation, the VAE's encoder generates a probability distribution characterized by its mean and standard deviation. This probabilistic encoding enables the generation of diverse synthetic 3D models through random sampling from the learned distribution.

In order to increase the accuracy of the 3D reconstructions the models, the mean squared false cross-entropy loss (MSFEL) [45] function is the one authors employ. This loss function is specifically designed to address the particular challenges that come with voxel-based 3D models, since these models frequently contain sparse data (a large number of empty or unoccupied voxels).

4.7.1. Cross-entropy loss

Cross-entropy loss is commonly used in tasks involving classification, where it measures the difference between two probability distributions: the true distribution (ground truth) and the predicted distribution. In the context of voxel-based 3D reconstruction, it is used to compare the predicted voxel occupancy (whether a voxel is part of the 3D object or empty) against the ground truth.

4.7.2. False positive cross-entropy loss

FPCE focuses on penalizing the model when it predicts that a voxel is occupied (part of the 3D object) when, in fact, it is empty according to the ground truth. This approach helps reduce the number of false positives, where the model incorrectly includes parts of the space in the 3D object. The FPCE is given by (2):

$$FPCE = -\frac{1}{N}\sum_{n=1}^{N}[V_n log V'_n + (1 - V_n) log(1 - V'_n)] \quad (2)$$

In (2), N denotes the total number of unoccupied voxels in the ground truth, which should ideally be predicted as empty by the model. The variable V_n represents the ground truth value for the n-th voxel; in this context, V_n would be 0 to indicate that it is unoccupied. By contrast, V'_n is the model's predicted probability of occupancy for the n-th voxel, ranging between 0 and 1, where higher values reflect greater confidence that the voxel is occupied.

4.7.3. False negative cross-entropy loss

FNCE penalizes the model when it fails to predict that a voxel is occupied despite the ground truth indicating that it is. This penalty helps reduce the number of false negatives, where the model misses parts of the object. The FNCE is (3):

$$FNCE = -\frac{1}{p}\sum_{p=1}^{p}[V_p log V'_p + (1 - V_p) log(1 - V'_p)] \quad (3)$$

In (3), p represents the total number of occupied voxels in the ground truth, which ideally should be predicted as occupied by the model. The variable V_p denotes the ground truth value for the p-th voxel, set to 1 when the voxel is indeed occupied. Meanwhile, V'_p is the predicted probability of occupancy for the ppp-th voxel, ranging between 0 and 1, where higher values reflect greater confidence that the voxel is occupied.

4.7.4. Mean squared false cross-entropy loss

The MSFEL creates a more thorough and balanced loss function by combining the FPCE and FNCE. When creating the 3D model, it makes sure the model is penalized for both kinds of errors (false positives and false negatives).

$$MSFEL = FPCE + FNCE \quad (4)$$

This loss function tackles the problem of sparse data, which makes it very useful in the context of voxel-based 3D reconstruction. Since the actual object often occupies only a small section of the voxel grid, the model must exercise caution to avoid misidentifying empty voxels as part of the object (false positives) or missing occupied voxels (false negatives).

4.7.5. Experiments

The models were tested using a subset of the ShapeNet [46] data set, which includes categories like cars, chairs, guitars, and tables. Every object category has several photos that capture the thing's whole 360° view from various perspectives. In the tests, the matching 3D models of these things serve as ground truth. To validate their performance, the suggested V3DOR-AE and V3DOR-VAE approaches have been tested against a number of state-of-the-art methodologies such as 3D-R2N2 [47], OccNet [48], SoftRas [49], NMR [50], and 3D-Recons [51]. Using the IOU [44] metric as a reference, the comparison concentrated on the accuracy of the 3D models that were reconstructed.

4.7.6. The results

According to the experiment results, the suggested V3DOR-AE and V3DOR-VAE models consistently performed better than the current techniques for a variety of object categories. In particular, the mean IOU scores of both models were higher, showing a closer match between the reconstructed 3D models and the actual data.

The V3DOR-AE model had an IOU score of 0.713 for the car category, whilst the V3DOR-VAE model received a score of 0.708. By contrast, OccNet had the highest score of 0.731 out of all the baseline approaches. This is somewhat better than V3DOR-VAE, but it's crucial to remember that V3DOR-AE performed better than 3D-R2N2, which had an IOU of 0.661, by 0.052 points, while being within 0.018 points of OccNet.

V3DOR-AE and V3DOR-VAE both received IOU scores in the table category of 0.508 and 0.509, respectively. These scores were about 0.088 to 0.089 points higher than 3D-R2N2's performance, which had an IOU of 0.420. They also performed 0.002 to 0.003 points better than OccNet, which had an IOU of 0.506. The proposed methods also produced remarkable outcomes when used to rebuild chairs and guitars. As an example, V3DOR-AE outperformed 3D-R2N2 by 0.072 points, with an IOU of 0.511 for chairs as opposed to 0.439 for the latter. The suggested approaches outperformed the majority of baseline methods for guitars, with an IOU difference ranging from 0.03 to 0.05 points.

The V3DOR-AE and V3DOR-VAE approaches provide notable enhancements in capturing intricate geometric details when compared to the LSTM-based 3D-R2N2 approach. For instance, the suggested techniques outperformed 3D-R2N2 by about 0.342 points in the mean IOU scores (1.814 vs 1.472 for V3DOR-AE).

Table 6 compares the IOU scores of the proposed V3DOR-AE and V3DOR-VAE methods against state-of-the-art approaches across different object categories. The suggested V3DOR-AE had a modest advantage with a mean IOU of 1.814, indicating a 0.08-point improvement, even though the OccNet technique fared well with a mean IOU of 1.734. This demonstrates the enhanced capacity of the suggested models to produce precise and detailed 3D reconstructions.

Table 6. IOU comparison of proposed methods with state-of-the-art methods by object category [44]

Method	Car (IOU)	Table (IOU)	Lamp (IOU)	Chair (IOU)	Mean IOU
3D-R2N2 (LSTM)	0.661	0.420	0.281	0.439	1.472
OccNet (CNN)	0.731	0.506	0.370	0.502	1.734
SoftRas (CNN)	0.672	0.453	0.444	0.481	1.662
NMR (CNN)	0.709	0.483	0.413	0.499	1.73
3D-Recons (CNN)	0.675	0.470	0.459	0.493	1.727
V3DOR-AE	0.713	0.508	0.465	0.511	1.814
V3DOR-VAE	0.708	0.509	0.454	0.509	1.798

4.8. 3D shape generation via variational autoencoder with signed distance function relativistic average generative adversarial network

3D-VAE-SDFRaGAN [52] is a new deep learning model that combines the frameworks of VAE and GAN. The goal of the model is to convert 2D photos into high-quality 3D shapes. It uses the SDF format to capture complex 3D structures and also integrates a relativistic average GAN loss function to enhance the stability and quality of the generated 3D shapes.

4.8.1. The problem

The main objective of the study is to overcome the difficulties that arise when creating realistic 3D models from 2D images. Standard approaches based on GANs and VAEs are often unstable during training and tend to produce inaccurate results, especially when the data has a complex structure. To address these shortcomings, the authors propose a more robust and efficient model capable of generating complex 3D objects with improved surface smoothness.

4.8.2. Methods

The process consists of developing a 3D-VAE-SDFRaGAN model that combines the strengths of GAN and VAE. The method starts by encoding the input 2D image into a latent space using VAE. The resulting latent representation is then fed into an SDF generator, which produces a 3D signed distance function, which is then converted into a polygonal mesh surface. A relativistic mean loss function of the GAN is used to stabilize training, enhance the quality of the generated 3D forms, and improve convergence.

4.8.3. Architecture

The 3D-VAE-SDFRaGAN architecture consists of three main components: a 2D-image encoder network, an SDF-generator network, and an SDF-discriminator network. The 2D-image encoder network processes 2D images and encodes them into a latent space, which is a simplified internal representation that the model later uses to generate 3D shapes. The five convolutional layers that make up this encoder have filter channels that are set to 64, 128, 256, 512, and 400. The complexity of each layer increases gradually. These layers' kernel sizes begin huge and get smaller, going from 11 to 8, with steps reduced correspondingly from 4 to 1. The encoder produces a 400-dimensional vector, which is divided into two 200-dimensional vectors: a mean vector and a variance vector. These vectors are essential for encapsulating the characteristics and fluctuations of the image. The encoder uses batch normalization between layers and ReLU activation functions to guarantee efficient learning. It optimizes the learning process using a combination of Kullback–Leibler divergence, which measures how well the model's learned distribution aligns with a standard distribution, and reconstruction loss, which assesses how accurately the model can recreate the original data from the latent space.

After that, the SDF-generator network receives the latent vector that the encoder created, and it uses this latent space to create the SDF. The generator consists of five layers, each of which is a transpose convolutional layer with filter channels set to 512, 256, 128, 64, and 1. The strides are adjusted from 1 in the first layer to 2 in the subsequent layers, and the kernel sizes in each layer are always set at 4. The 64×64×64 SDF matrix that the generator produces shows the distances to the 3D shape's surface. The values in the matrix range from -1 to 1. ReLU activation functions are used across network layers to precisely capture the 3D shape, and a Tanh activation is then used at the output layer to generate an SDF. This SDF is then transformed into a triangular mesh using the marching cubes technique, producing an intricate 3D representation. Finally, to evaluate the authenticity of the SDFs created by the model, the SDF-discriminator network compares the generated SDFs with real SDFs from the dataset. This discriminator also consists of five convolutional layers, with filter channels 64, 128, 256, 512, and 1 that reflect the structure of the generator. Except for the final layer, which employs a stride of 1, all layers have strides of 2. The kernel sizes remain constant at four. At the last layer, the network uses a Sigmoid function to provide an output between 0 and 1, which indicates the probability that the SDF is generated or real. In order to help control the gradient flow, leaky ReLU activation functions are employed throughout the layers.

4.8.4. Experiments

The studies were carried out using the ShapeNet collection, which contains 3D models of a wide variety of object categories, including chairs, tables, cars, lamps, couches, and cabinets. In order to generate the 2D photos in the collection, the 3D models were rendered from 23 different angles. The corresponding 3D SDFs were then constructed using these models. For the training process, these 2D pictures were used in pairs with the corresponding SDFs. A collection of 2D photos and the matching SDFs for every object category were used to train the model. Each 2D image was processed by the 2D-image encoder network during training in order to generate a latent vector that represents the 2D image's encoded features. This latent vector was then passed to the SDF-generator network, which used it to generate a signed distance function that reflected the

three-dimensional geometry of the item. Random noise vectors drawn from a uniform distribution were also supplied into the SDF-generator network in order to add variations to the SDFs that were produced. The generated and real SDFs from the dataset were then input into the SDF-discriminator network. The discriminator's task was to distinguish between the real SDFs and the ones generated by the model. The generator's output and the 3D forms it generated were then enhanced by using the discriminator's input.

The discriminator network had a learning rate of 10^{-5}, but the generator and encoder networks both had learning rates of 10^{-3}. The model was trained using varying learning rates for each component. During training, the network weights were updated using the Adam optimizer, whose momentum was controlled by setting the β parameter to 0.5. 64 image-SDF pairings were processed concurrently in each training iteration since the model was trained with a batch size of 64. The number of epochs used to train the model varied according to the complexity of the object category: 1500 epochs were used for tables, lights, sofas, and cabinets, while 1700 epochs were used for more complex categories such as chairs and cars. Due to its complex design that combines the VAE and GAN frameworks, the model has high computing requirements, as evidenced by the training procedure taking more than 192 hours on a single Nvidia GeForce GTX 1080 GPU.

To assess the quality of the model, the chamfer distance (CD) metric was used. It determines the similarity between the vertices of the generated and real 3D shapes. The lower the CD value, the more accurate and closer the reconstruction is to the original object. The 3D-VAE-SDFRaGAN indicators were compared with the results of several current non-linear models, also using CD, for different categories of objects. These models included 3D-R2N2 [47], SIF [53], N3MR [50], MeshSDF [54], and DISN [55]. The results in Table 7 demonstrate that the proposed model achieved the lowest average CD score of 0.578, significantly outperforming other models. Visual inspection and comparison of the produced 3D shapes with those from other models showed that the 3D-VAE-SDFRaGAN produced results that were more realistic and detailed. Overall, the research showed that the 3D-VAE-SDFRaGAN model is reliable and efficient in creating high-quality 3D shapes from 2D images, suggesting that it might be used as a tool for computer graphics, virtual reality, and 3D modeling.

Table 7. Comparison of the suggested approach's performance on the ShapeNet dataset with the state-of-the-art [52]

Category	3D-R2N2	SIF	N3MR	MeshSDF	DISN	Proposed method
Chair	1.432	1.540	2.084	0.590	0.754	0.589
Table	1.116	1.570	2.383	1.070	1.329	0.672
Car	0.845	1.080	2.298	0.960	0.492	0.491
Lamp	4.009	3.420	3.013	1.490	2.273	0.662
Sofa	1.135	0.800	3.512	0.780	0.871	0.566
Cabinet	0.750	1.100	2.555	0.780	1.130	0.314
Average	1.545	1.585	2.641	0.945	1.142	0.578

5. COMPARATIVE ANALYSIS

Table 8 provides a comprehensive comparative analysis of all reviewed techniques, examining their input/output types, architectures, performance metrics, advantages, and disadvantages. The input and output type columns specify the data formats used for input and output. The architecture column details the underlying technology employed in the method's core. Performance metrics highlight the evaluation criteria used to assess a method's effectiveness. Finally, the advantages and disadvantages columns summarize the key strengths and weaknesses of the reviewed models.

All the generative models examined utilize deep learning architectures, such as GANs, VAEs, to synthesize 3D structures from various input data types. Despite their diverse output formats (voxel grids, meshes, and signed distance fields), they share a common goal of reconstructing or generating 3D shapes. However, the computational demands of these models are often substantial due to the complexity of processing high-dimensional 3D data.

In terms of input types, models like 3DGAN and CGAN rely primarily on 2D images, while voxel-based reconstruction uses point cloud data, paired 3D model GAN requires paired 2D-3D data, and 3D-VAE-SDFRaGAN utilizes SDF. Output formats vary, with 3DGAN and voxel-based reconstruction generating voxel grids, whereas 3D-VAE-SDFRaGAN and FaceVAE produce meshes or other structured polygonal representations. FaceVAE, in particular, generates 3D geometric data directly from face-based (polygonal) structures, focusing on polygon geometry, and is designed for applications where direct 3D geometry manipulation is critical. Architectural complexity is higher in hybrid models like 3D-VAE-SDFRaGAN, which combine features of VAEs and GANs, while simpler autoencoder architectures are used in FaceVAE and voxel-based reconstruction. Training difficulty is more pronounced in GAN-based models, such as 3DGAN and

conditional GAN, which struggle with issues like mode collapse and balancing the generator and discriminator, whereas VAE-based models like FaceVAE are more stable and easier to train. Hybrid models are the most challenging to train due to their combination of multiple architectures and loss functions. Regarding applications, FaceVAE is specialized for generating 3D geometric data using polygon structures, while 3DGAN and Conditional GAN offer more flexibility for broader domains like object reconstruction and shape generation. Voxel-based reconstruction is particularly useful in fields requiring precision, such as robotics, due to its ability to capture fine details.

Table 8. Comparative analysis of reviewed techniques

Technique	Input type	Output type	Architecture	Performance metrics	Advantages	Disadvantages
3DGAN	2D images	Voxel grid	GAN	IoU and chamfer distance	High-quality voxel-based 3D models	Computationally expensive and voxel resolution limited
Conditional GAN	Conditional labels	Mesh and voxel	GAN + conditional	FID and chamfer distance	Better control over output using conditional inputs	Difficult training and mode collapse risks
Paired 3D model GAN	Paired data (2D+3D)	Voxel grid and mesh	GAN	Average absolute difference	Capable of learning complex shapes	Requires paired data and which can be difficult to obtain
FaceVAE	Image (2D Fface)	3D face mesh	VAE	AVAR and IoU	Fast training and good for face meshes	Model was only trained on geometries with 300 or fewer vertices and which is a small subset of the full dataset
Voxel-based reconstruction	Point cloud	Voxel grid	Autoencoder	IoU and chamfer distance	Captures fine details of structures	Memory-intensive, high computational cost
3D-VAE-SDFRaGAN	Signed SDF	Mesh and SDF	VAE+GAN (hybrid)	Chamfer distance and SDF loss	Produces high-quality mesh and surface details	Training is complex and stability issues in training GANs

All the generative models examined utilize deep learning architectures, such as GANs, VAEs to synthesize 3D structures from various input data types. Despite their diverse output formats (voxel grids, meshes, and signed distance fields), they share a common goal of reconstructing or generating 3D shapes. However, the computational demands of these models are often substantial due to the complexity of processing high-dimensional 3D data.

In terms of input types, models like 3DGAN and conditional GAN rely primarily on 2D images, while Voxel-based reconstruction uses point cloud data, paired 3D Model GAN requires paired 2D-3D data, and 3D-VAE-SDFRaGAN utilizes SDF. Output formats vary, with 3DGAN and voxel-based reconstruction generating voxel grids, whereas 3D-VAE-SDFRaGAN, and FaceVAE produce meshes or other structured polygonal representations. FaceVAE, in particular, generates 3D geometric data directly from face-based (polygonal) structures, focusing on polygon geometry, and is designed for applications where direct 3D geometry manipulation is critical. Architectural complexity is higher in hybrid models like 3D-VAE-SDFRaGAN, which combine features of VAEs and GANs, while simpler autoencoder architectures are used in FaceVAE and voxel-based reconstruction. While VAE-based models like FaceVAE are more stable and simpler to train, GAN-based models like 3DGAN and conditional GAN have more significant training difficulties due to problems like mode collapse and balancing the generator and discriminator. Because hybrid models combine several architectures and loss functions, they are the most difficult to train. In terms of applications, 3DGAN and conditional GAN provide greater flexibility for more general domains like object reconstruction and form production, whereas FaceVAE is focused on producing 3D geometric data utilizing polygon structures. Because it can capture precise features, voxel-based reconstruction is very helpful in fields that require precision.

6. LIMITATIONS, CHALLENGES, AND FUTURE WORK

6.1. 3DGAN

The 3D-GAN framework has a number of drawbacks, such as the inability to produce high-resolution 3D things because to the quick rise in computing complexity with increasing dimensions and the production of objects with structural irregularities like holes and fragments. Because the discriminator learns more quickly than the generator, training is also difficult and calls for adaptive learning methodologies. Furthermore, the voxel-based method is better suited for basic item categories because it can't handle complex geometries with fine-grained features.

In order to preserve computational feasibility while enhancing object detail, future 3D-GAN research should concentrate on improving resolution and detail using multi-resolution techniques. Enhancing training stability could involve techniques such as curriculum learning or progressive GAN growth to balance generator and discriminator learning. Integrating mesh and point cloud representations would capture more fine-grained details and expand the model's applicability. Finally, improving single-image 2D to 3D reconstructions by incorporating multiple views or depth maps could significantly boost reconstruction accuracy.

6.2. Paired 3D model generation with conditional generative adversarial networks

The paper on paired 3D model generation using CGANs presents several limitations and gaps. One of the primary limitations is that standard CGANs generate different objects even when only the condition value (e.g., rotation) is changed, failing to create the same object across different conditions. The research addresses this by introducing a new training step, but the method requires merging samples, which is domain-specific and might not generalize well to all applications. Additionally, the process relies on voxel-based representations, which may limit the model's ability to generate high-resolution 3D objects due to voxel grid constraints. Another gap is that the research only explores 3D objects with a few fixed rotations (2 or 4), and extending the model to handle more conditions and larger object classes is left as future work.

The research highlights several challenges, particularly the high computational costs and memory demands of 3D convolutional networks compared to 2D networks. These limitations prevent the use of higher resolution models, which would significantly improve the quality of the generated 3D models. Additionally, while the method shows promise in generating diverse and realistic 3D models based on class labels and reconstructing models from images, the resolution and speed remain constrained by available computational resources. Future work will focus on adapting super-resolution techniques from the 2D image domain to 3D models, with the goal of generating high-resolution 3D models efficiently.

6.3. FaceVAE

The unresolved challenges in the FaceVAE method include difficulties in handling unstructured geometric data, where transforming such data into a structured format remains complex and inefficient. Sparse matrix representations (AF and AV) lead to computational inefficiencies, particularly with large geometric models, and the method also suffers from overfitting due to limited training data. Additionally, reconstruction of 3D geometries from adjacency and feature matrices is prone to errors, sometimes generating non-existent faces, which reduces the overall model fidelity. Lastly, the omission of normal vectors, texture UV, and other detailed surface features limits the practical applicability of the method, as the generated models lack critical surface details for industrial use.

Future research on the FaceVAE method should focus on improving matrix structuring efficiency, potentially using hierarchical techniques like octrees to dynamically adjust matrix sizes based on geometric complexity. Scaling the model to handle larger geometries, along with better data compression techniques, could address overfitting and improve generalization.

6.4. Voxel-based 3D object reconstruction from single 2D image using variational autoencoders

A number of significant drawbacks and difficulties are highlighted by the work on voxel-based 3D object reconstruction using VAEs, especially when considering dataset dependence, model complexity, and processing demands. When applied to diverse or unexplored data, the existing models, which were mostly tested on the ShapeNet dataset, show low robustness due to poor cross-dataset generalization. This limits the model's use in situations where there is a lot of data unpredictability. Additionally, the simplicity of autoencoder and VAE designs restricts the model's capacity to capture fine features, which becomes crucial when working with more complicated 3D structures, even though it has advantages like quicker training and lower computing cost.

The computational complexity of training 3D reconstruction models is still a major challenge, especially when using voxel-based techniques. Scaling output resolution requires significant processing resources, even with efforts to simplify the design. Additionally, the model has trouble reconstructing extremely complicated or irregular geometries; it does well with simpler items like chairs and vehicles but struggles to produce fine-grained details from a single 2D input image.

The authors suggest a number of ways to improve future research. In order to create a more resilient framework that can manage novel and varied data, it is crucial to improve the model's generalization skills across various datasets. Furthermore, they propose adding color and texture data to the reconstructions, which would greatly improve the models' realism and usefulness. More complex designs and training techniques might be required, nevertheless. Improving resolution is another crucial topic that needs more investigation. The reconstructed objects' smoothness and intricacy are constrained by the current voxel grid resolution of 32^3

voxels. While maximizing computational efficiency, investigating higher-resolution voxel grids or other representations like point clouds, meshes, or implicit surfaces may provide more detail.

6.5. 3D Shape generation via variational autoencoder with signed distance function relativistic average generative adversarial network

The study has a number of limitations, especially with regard to the amount of data needed, the amount of computing power needed, and the sensitivity of the model. The method is less practical for contexts with limited resources because it requires a significant amount of training data and computer power. Furthermore, the selection of hyperparameters has a significant impact on the model's performance, necessitating careful adjustment to get the best outcomes. Its applicability to increasingly complicated 3D constructions is further limited by the architecture's difficulty in producing complex shapes and geometries, especially those involving non-uniform scaling or non-rigid deformations.

The authors' recommendations for future research focus on improving the produced 3D models' accuracy and level of detail by adding more data sources or improving the current network architecture. Expanding the model's ability to manage increasingly complex shapes is another suggested approach. Lastly, to improve the model's practical utility in specialized fields, the authors propose adding semantic information or other types of prior knowledge to improve the model's capacity to produce domain-specific and application-relevant 3D models.

Table 9 arranges the examined approaches according to their shared drawbacks and difficulties. It shows that training instability affects multiple GAN-based methods including 3DGAN, Conditional GAN, and signed distance function GAN, while high computational demands are common across both GAN and VAE-based approaches.

Table 9. Methods grouped by common limitations and challenges

Common challenges and limitations	Methods
Training instability	- 3DGAN [30] - Conditional GAN for paired 3D model generation [38] - Signed distance function GAN [52]
High computational and memory demands	- 3DGAN [30] - Voxel-based 3D object reconstruction with autoencoder [44] - Signed distance function GAN [52]
Low-resolution output	- 3DGAN [30] - Voxel-based 3D object reconstruction with VAE [44] - Paired 3D model generation with CGAN [38]
Generalization issues (cross-dataset validation)	- Voxel-Based 3D object reconstruction with VAE [44] - Paired 3D model generation with CGAN [38] - Signed distance function GAN [52]
Handling complex geometries	- Paired 3D model generation with CGAN [38] - Signed distance function GAN [52]

7. RESULTS AND DISCUSSION

7.1. RQ1: What are the key advancements in using GANs and VAEs for 3D model generation?

GAN-based 3D model synthesis has centered on addressing the inherent instability of training. Novel loss functions have been proposed to enhance training stability, leading to more reliable and consistent model generation. Additionally, GANs have demonstrated their capacity to generate 3D models with high-quality geometric details and topological accuracy. This capability, coupled with the ability to control specific object attributes, makes GANs a promising tool for a diverse range of applications requiring scalable and efficient 3D model generation.

On the other hand, VAEs have made substantial contributions to 3D model generation by introducing efficient latent representations that balance computational efficiency and model performance. The ability of VAEs to generate synthetic 3D data has enriched datasets and promoted model diversity. While VAE-based 3D model reconstruction from single 2D images may exhibit certain quality limitations, their capacity to capture and generate a wide range of 3D shapes from compact latent codes remains a significant advantage.

The key challenges and potential future work associated with each GAN- and VAE-based method are summarized in Table 10.

Table 10. Unique challenges and future work for each reviewed method

Method	Unique challenges	Future work
3DGAN	Difficulty in scaling to higher resolution outputs due to computational overhead.	– Improving training stability using advanced techniques like Wasserstein GAN. – Experimenting with regularization methods to stabilize training. – Enhancing the resolution of 3D models using super-resolution techniques.
Conditional GAN for paired 3D model generation	Difficulty in generating consistent 3D models across different rotations or viewpoints.	– Focus on enhancing stability with optimization techniques and different loss functions.
	Struggles with handling complex geometries when generating paired 3D models.	– Extend the method to handle more complex object classes and rotations. – Explore more efficient cross-dataset generalization approaches.
Voxel-based 3d object Reconstruction with Autoencoder	Challenges in maintaining high-quality output when generating synthetic 3D models with the VAE method.	– Improve cross-dataset generalization for real-world applications. – Incorporate texture and color into 3D models to increase realism. – Enhance the resolution of output models using super-resolution methods.
Signed distance function GAN	Difficulty in capturing fine details for complex or irregular geometries using signed distance functions.	– Work on incorporating texture and color into 3D models. – Improve generalization across datasets, potentially using self-supervised learning techniques. – Explore techniques to capture finer geometric details for high-complexity models.
Voxel-based 3D object reconstruction with VAE	Challenges in generating realistic synthetic models when using random noise in the VAE process.	– Improve synthetic model generation with more realistic outputs. – Enhance cross-dataset validation performance. – Incorporate textures and color to improve the realism of generated 3D models.
Paired 3D model generation with CGAN	Difficulty in generating consistent object pairs across varying conditions (e.g., different rotations).	– Improve stability in training and handling of complex geometries.
	Poor performance when handling complex geometries.	– Test more conditions (e.g., higher number of rotations). – Experiment with new optimization techniques to improve model performance.

7.2. RQ2: What are the key datasets and evaluation metrics used in GAN- and VAE-based 3D model generation studies, and how do they influence the performance of these models?

A variety of evaluation metrics, including IoU, chamfer distance, AAD, and AVAR, have been extensively utilized in the literature to provide quantitative benchmarks for assessing the accuracy and fidelity of 3D reconstructions.

The choice of datasets significantly influences the adaptability of these models to new data. Models trained on more diverse datasets, like ShapeNet, tend to perform better on unseen data, whereas models tested solely on constrained datasets like ModelNet10 often exhibit dataset-specific overfitting.

7.3. Q3: How does VAE-based models compare to GAN-based models in generating 3D models, particularly in terms of computational efficiency?

Although GANs excel in generating diverse 3D models, their computational demands often pose practical limitations. Models such as 3DGAN and CGAN require significant GPU resources and are prone to training instability. The paired 3D Model GAN, despite its innovative approach to generating consistent object pairs, incurs substantial memory and computational costs, particularly when dealing with high-resolution voxel grids or large datasets. These computational demands make GANs less suitable for environments with limited hardware resources.

In contrast to GANs, VAEs offer a more stable and computationally efficient approach to 3D model generation. Models such as FaceVAE and Voxel-Based reconstruction with VAE are relatively lightweight and exhibit faster training times, making them suitable for real-time applications and resource-constrained environments. However, they often fall short of GANs in terms of output diversity and fine-grained details. The hybrid 3D-VAE-SDFRaGAN model aims to bridge this gap by combining the efficiency of VAEs with the high-quality output potential of GANs, but at the cost of increased training complexity and time.

The use of AI methods in 3D model generation, such as GANs and VAEs, signifies a significant change from earlier modeling strategies. These methods open up new possibilities for industries like 3D animation, virtual and AR, and gaming industry by automating difficult modeling jobs. Even with the encouraging outcomes, scaling these models to handle increasingly complicated forms, enhancing training stability, and mximizing computational efficiency are still difficult tasks.

The advances reviewed in this paper demonstrate a clear evolutionary trajectory in AI-driven 3D model generation. Early methods like 3DGAN [30] established the foundational voxel-based approach but struggled with resolution limitations and training instability. The field has progressively advanced through conditional architectures [39], [38] that improved control over outputs, to more sophisticated representations including polygon-based structures [43] and signed distance functions [52]. Current trends indicate a shift from single-architecture approaches toward hybrid models that combine VAE and GAN strengths, as exemplified by 3D-VAE-SDFRaGAN's superior chamfer distance score of 0.578 compared to earlier methods. These advances represent significant progress in addressing fundamental challenges: training stability has improved through relativistic average loss functions, resolution limitations are being overcome through SDF representations, and computational efficiency has been enhanced through optimized architectures like FaceVAE's adjacency matrix approach.

Methods involving GANs (3DGAN, paired 3D model generation with GAN, conditional GAN, and 3D-VAE-SDFRaGAN) employ adversarial training, where a generator network creates 3D models, and a discriminator network evaluates the realism of these models. The generator aims to fool the discriminator, resulting in more realistic outputs over time. The adversarial nature of GANs can lead to instability during training, where the generator and discriminator might not converge properly, requiring careful tuning of the learning process.

VAEs focus on learning a latent space representation of the input data. The model encodes data into a latent space and then decodes it back into the original space, aiming to minimize the difference between the original and reconstructed data. VAEs generally offer more stable training than GANs, as they do not involve an adversarial setup. The training process minimizes a combination of reconstruction loss and a regularization term.

The reviewed methods utilized several well-known datasets such as ModelNet, ShapeNet and Ikea for training, primarily focusing on 3D objects and their corresponding 2D representations. In order to evaluate the results of proposed methods, authors used varios criterias: i) IOU used in V3DOR-AE and V3DOR-VAE to to evaluate how well the generated 3D voxel-based models match the ground truth models, ii) chamfer distance was used in 3D-VAE-SDFRaGAN. It measures the geometric accuracy of the generated 3D shapes by calculating the distance between points on the surfaces of the generated and actual models, iii) the paired 3D Model Generation with GAN method uses two specific evaluation criteria: AAD and AVAR. The difference between models created under various circumstances, such as different rotation degrees, is measured using AAD. By evaluating the voxel grid consistency among these produced models, on the other hand, AVAR makes sure that the models stay consistent even when circumstances alter. When combined, these measures assess the created 3D models' stability and variability, iv) mean mAP is used by the CGAN for 3D Model Generation and Reconstruction method to assess the accuracy of the models produced for various object categories. In order to guarantee that the produced models are appropriately categorized and satisfy the desired design standards, this criterion evaluates how well the models fit into particular classes, such as chairs, tables, and other categories, v) the FaceVAE technique uses precision and recall as assessment criteria to create 3D geometric objects using VAEs. The efficiency of learning adjacency matrices and feature information during the generation process is evaluated using these metrics. Error minimization—more especially, lowering the quantity of false positives and false negatives in the produced models—is the main goal, and vi) the effectiveness of the proposed 3D-GAN method was evaluated by several criteria. First, the quality of the generated 3D objects was analyzed. Second, the ability of the model to generalize data beyond the training set was tested. The classification accuracy was also assessed on popular 3D object sets, ModelNet10 and ModelNet40. Finally, the performance of the method in 3D reconstruction from a single image was investigated using the IKEA dataset.

The comparative analysis reveals distinct advantages for different approaches: GAN-based methods (3DGAN and CGAN) excel in generating diverse, high-quality 3D models with fine geometric details but suffer from training instability and high computational costs. VAE-based methods (FaceVAE and voxel-based reconstruction) offer superior training stability and computational efficiency, making them ideal for real-time applications, though they produce less detailed outputs. Hybrid approaches like 3D-VAE-SDFRaGAN achieve the best overall performance (chamfer distance: 0.578) by combining both architectures' strengths, though at the cost of increased training complexity.

8. CONCLUSION

This review summarizes the evolution of AI-driven 3D model generation by comparing traditional modeling methods with modern GAN- and VAE-based approaches. While classical techniques such as polygonal modeling, NURBS, sculpting, and procedural workflows require significant time and professional skill, recent AI models demonstrate the potential to automate and accelerate 3D content creation. The reviewed advances—ranging from 3DGAN and CGAN to FaceVAE, voxel-based reconstruction, and hybrid architectures—illustrate clear progress in reconstruction quality, geometric representation, training stability, and computational efficiency.

The implications of these findings show that AI-based models are becoming increasingly relevant for applications in animation, gaming, VR/AR, industrial design, and simulation, offering scalable tools for complex shape generation. However, limitations remain: GANs suffer from instability and high computational costs, VAEs often lack fine detail, hybrid models introduce greater training complexity, and model performance still depends heavily on dataset characteristics. Addressing these constraints requires improved cross-domain benchmarks, more robust training pipelines, and methods that enhance stability, controllability, and generalization. Continued research following these directions will further strengthen the practical use and reliability of AI-driven 3D model generation.

FUNDING INFORMATION

Authors state no funding involved.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Shyngys Adilkhan	✓	✓				✓		✓	✓	✓	✓			
Madina Alimanova		✓		✓		✓				✓		✓		
Lei Shi	✓	✓		✓			✓		✓			✓		
Aiganym Soltiyeva						✓				✓	✓			

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

DATA AVAILABILITY

The authors confirm that the data supporting the findings of this study are available within the article [and/or its supplementary materials].

REFERENCES

[1] J. D. Foley, *Computer Graphics — Principles and Practice*, Addison-Wesley, 1996.
[2] J. J. McConnell, *Computer Graphics Companion*. Computer graphics companion, Macmillan Press Ltd, 2002.
[3] M. S. El-Nasr and I. Horswill, "Automating Lighting Design for Interactive Entertainment," *Computers in Entertainment*, vol. 2, no. 2, pp. 15–15, 2004, doi: 10.1145/1008213.1008238.
[4] F. I. Parke and K. Waters, "Computer Facial Animation," *Computer Facial Animation*, 1996.
[5] T. Capin, K. Pulli, and T. Akenine-Möller, "The state of the art in mobile graphics research," *IEEE Computer Graphics and Applications*, vol. 28, no. 4, pp. 74–84, 2008, doi: 10.1109/MCG.2008.83.
[6] F. C. Crow, "Shaded computer graphics in the entertainment industry," *Computer*, vol. 11, no. 3, pp. 11–22, 1978, doi: 10.1109/C-M.1978.218090.
[7] F. P. Vidal *et al.*, "Principles and applications of computer graphics in medicine," *Computer Graphics Forum*, vol. 25, no. 1, pp. 113–137, 2006, doi: 10.1111/j.1467-8659.2006.00822.x.
[8] R. Gallagher, *Computer Visualization: Graphics Techniques for Scientific and Engineering Analysis*, Solomon Press, 1994.

[9] G. McGill, "Molecular Movies... Coming to a Lecture near You," *Cell*, vol. 133, no. 7, pp. 1127–1132, 2008, doi: 10.1016/j.cell.2008.06.013.
[10] H. Baeverstad, "Engineering and scientific visualization using high-performance graphics workstations," in *Digest of Papers. COMPCON Spring 89. Thirty-Fourth IEEE Computer Society International Conference: Intellectual Leverage*, San Francisco, CA, USA, 1989, pp. 328–333, doi: 10.1109/cmpcon.1989.301951.
[11] J. L. Encarnação, "Edutainment and serious games - Games move into professional applications," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 4469, p. 2, 2007, doi: 10.1007/978-3-540-73011-8_2.
[12] Y. Ding, "The Impact of Artificial Intelligence on Art Research: An Analysis of Academic Productivity and Multidisciplinary Integration," *arXiv*, 2024, doi: 10.48550/arXiv.2412.04850.
[13] Z. Shi, S. Peng, Y. Xu, A. Geiger, Y. Liao, and Y. Shen, "Deep Generative Models on 3D Representations: A Survey," *arXiv*, 2022, doi: 10.48550/arXiv.2210.15663.
[14] S. P. Tata and S. Mishra, "3D GANs and Latent Space: A comprehensive survey," *arXiv*, 2023, doi: 10.48550/arXiv.2304.03932.
[15] J. Li, C. Niu, and K. Xu, "Learning part generation and assembly for structure-aware shape synthesis," in *Proceedings of the AAAI conference on artificial intelligence*, 2020, pp. 11362–11369, doi: 10.1609/aaai.v34i07.6798.
[16] J. Xie, Z. Zheng, R. Gao, W. Wang, S. C. Zhu, and Y. N. Wu, "Generative VoxelNet: Learning Energy-Based Models for 3D Shape Synthesis and Analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 5, pp. 2468–2484, 2022, doi: 10.1109/TPAMI.2020.3045010.
[17] X. Pan, B. Dai, Z. Liu, C. C. Loy, and P. Luo, "Do 2D Gans Know 3D Shape? Unsupervised 3D Shape Reconstruction From 2D Image Gans," *arXiv*, 2020, doi: 10.48550/arXiv.2011.00844.
[18] X. Zeng *et al*., "LION: Latent Point Diffusion Models for 3D Shape Generation," *Advances in Neural Information Processing Systems*, vol. 35, pp. 10021-10039, 2022.
[19] S. Hong *et al*., "3D-StyleGAN: A Style-Based Generative Adversarial Network for Generative Modeling of Three-Dimensional Medical Images," *Lecture Notes in Computer Science*, vol. 13003 LNCS, pp. 24–34, 2021, doi: 10.1007/978-3-030-88210-5_3.
[20] Q. Hu, H. Li, and J. Zhang, "Domain-Adaptive 3D Medical Image Synthesis: An Efficient Unsupervised Approach," *Lecture Notes in Computer Science*, vol. 13436, pp. 495–504, 2022, doi: 10.1007/978-3-031-16446-0_47.
[21] C. Singla, R. Bhardwaj, N. Shelke, and G. Singh, "Data Augmentation: Synthetic Image Generation for Medical Images Using Vector Quantized Variational Autoencoders," in *2025 3rd International Conference on Disruptive Technologies (ICDT)*, Greater Noida, India, 2025, pp. 1502–1507, doi: 10.1109/ICDT63985.2025.10986686.
[22] A. Schnepf, F. Vasile, and U. Tanielian, "3DGEN: A GAN-based approach for generating novel 3D models from image data," *arXiv*, 2023, doi: 10.48550/arXiv.2312.08094.
[23] A. Ferreira, J. Li, K. L. Pomykala, J. Kleesiek, V. Alves, and J. Egger, "GAN-based generation of realistic 3D volumetric data: A systematic review and taxonomy," *Medical Image Analysis*, vol. 93, 2024, doi: 10.1016/j.media.2024.103100.
[24] Z. Zabati and D. Jovanovska, "3D modelling functions and algorithms," *In Central European Conference on Information and Intelligent Systems*, 2010, pp. 327–334.
[25] A. Lamba and R. Bhalla, "A Review of various 3-D Modelling Techniques and an Introduction to Point Clouds," in *Proceedings of The International Conference on Emerging Trends in Artificial Intelligence and Smart Systems*, Jabalpur, India, 2022, doi: 10.4108/eai.16-4-2022.2318159.
[26] D. Kitsakis, E. Tsiliakou, T. Labropoulos, and E. Dimopoulou, "Procedural 3D modelling for traditional settlements. The case study of central zagori," *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives*, vol. 42, no. 2W3, pp. 369–376, 2017, doi: 10.5194/isprs-archives-XLII-2-W3-369-2017.
[27] I. Goodfellow *et al*., "Generative adversarial networks," *Communications of the ACM*, vol. 63, no. 11, pp. 139–144, 2020, doi: 10.1145/3422622.
[28] D. Paper, *Generative Adversarial Networks*, State-of-the-Art Deep Learning Models in TensorFlow, vol. 63, no. 11, Berkeley, CA: Apress, pp. 243–263, 2021, doi: 10.1007/978-1-4842-7341-8_10.
[29] D. V. Thada, M. U. Shrivastava, J. Sharma, K. P. Singh, and M. Ranadeep, "A Primer on Generative Adversarial Networks," *SSRN Electronic Journal*, 2020, doi: 10.2139/ssrn.3670831.
[30] J. Wu, C. Zhang, T. Xue, W. T. Freeman, and J. B. Tenenbaum, "Learning a probabilistic latent space of object shapes via 3D generative-adversarial modeling," *Advances in Neural Information Processing Systems*, pp. 82–90, 2016.
[31] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," *arXiv*, 2015, doi: 10.48550/arXiv.1511.06434.
[32] A. B. L. Larsen, S. K. Sønderby, H. Larochelle, and O. Winther, "Autoencoding beyond pixels using a learned similarity metric," in *Proceedings of The 33rd International Conference on Machine Learning*, 2016, pp. 1558-1566.
[33] Z. Wu *et al*., "3D ShapeNets: A deep representation for volumetric shapes," *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, MA, USA, 2015, pp. 1912-1920, doi: 10.1109/CVPR.2015.7298801.
[34] A. Sharma, O. Grau, and M. Fritz, "VConv-DAE: Deep volumetric shape learning without object labels," *Lecture Notes in Computer Science*, vol. 9915 LNCS, pp. 236–250, 2016, doi: 10.1007/978-3-319-49409-8_20.
[35] R. Girdhar, D. F. Fouhey, M. Rodriguez, and A. Gupta, "Learning a Predictable and Generative Vector Representation for Objects," *Lecture Notes in Computer Science*, pp. 484–499, 2016.
[36] D. Y. Chen, X. P. Tian, Y. Te Shen, and M. Ouhyoung, "On Visual Similarity Based 3D Model Retrieval," *Computer Graphics Forum*, vol. 22, no. 3, pp. 223–232, 2003, doi: 10.1111/1467-8659.00669.
[37] M. Kazhdan, T. Funkhouser, and S. Rusinkiewicz, "Rotation invariant spherical harmonic representation of 3D shape descriptors," *SGP03: Eurographics Symposium on Geometry Processing*, 2003, pp. 156–165, doi: 10.2312/SGP/SGP03/156-165.
[38] C. Öngün and A. Temizel, "Paired 3D model generation with conditional generative adversarial networks," *Lecture Notes in Computer Science*, vol. 11129, pp. 473–487, 2019, doi: 10.1007/978-3-030-11009-3_29.
[39] H. Li, Y. Zheng, X. Wu, and Q. Cai, "3D model generation and reconstruction using conditional generative adversarial network," *International Journal of Computational Intelligence Systems*, vol. 12, no. 2, pp. 697–705, 2019, doi: 10.2991/ijcis.d.190617.001.
[40] J. J. Lim, H. Pirsiavash, and A. Torralba, "Parsing IKEA objects: Fine pose estimation," in *Proceedings of the IEEE International Conference on Computer Vision*, Dec. 2013, pp. 2992–2999, doi: 10.1109/ICCV.2013.372.
[41] E. Smith and D. Meger, "Improved Adversarial Systems for 3D Object Generation and Reconstruction," *arXiv*, 2017, doi: 10.48550/arXiv.1707.09557.
[42] D. P. Kingma and M. Welling, "Auto-Encoding Variational Bayes," *arXiv*, 2022, doi: 10.48550/arXiv.1312.6114.
[43] S. Park and H. Kim, "Facevae: Generation of a 3D geometric object using variational autoencoders," *Electronics*, vol. 10, no. 22, 2021, doi: 10.3390/electronics10222792.

[44] R. Tahir, A. B. Sargano, and Z. Habib, "Voxel-based 3D object reconstruction from single 2d image using variational autoencoders," *Mathematics*, vol. 9, no. 18, 2021, doi: 10.3390/math9182288.
[45] S. Wang, W. Liu, J. Wu, L. Cao, Q. Meng, and P. J. Kennedy, "Training deep neural networks on imbalanced data sets," in *Proceedings of the International Joint Conference on Neural Networks*, Vancouver, BC, Canada, Oct. 2016, pp. 4368–4374, 2016, doi: 10.1109/IJCNN.2016.7727770.
[46] A. X. Chang *et al.*, "ShapeNet: An Information-Rich 3D Model Repository," *arXiv*, 2015, doi: 10.48550/arXiv.1512.03012.
[47] C. B. Choy, D. Xu, J. Y. Gwak, K. Chen, and S. Savarese, "3D-R2N2: A unified approach for single and multi-view 3D object reconstruction," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9912, pp. 628–644, 2016, doi: 10.1007/978-3-319-46484-8_38.
[48] L. Mescheder, M. Oechsle, M. Niemeyer, S. Nowozin, and A. Geiger, "Occupancy networks: Learning 3D reconstruction in function space," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2019, pp. 4455–4465, doi: 10.1109/CVPR.2019.00459.
[49] G. P.-Moll, J. Romero, N. Mahmood, and M. J. Black, "Dyna: A model of dynamic human shape in motion," *ACM Transactions on Graphics*, vol. 34, no. 4, 2015, doi: 10.1145/2766993.
[50] H. Kato, Y. Ushiku, and T. Harada, "Neural 3D Mesh Renderer," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2018, pp. 3907–3916, doi: 10.1109/CVPR.2018.00411.
[51] Y. Zhu, Y. Zhang, and Q. Feng, "Colorful 3d reconstruction from a single image based on deep learning," *ACM International Conference Proceeding Series*, 2020, doi: 10.1145/3446132.3446157.
[52] E. A. Ajayi, K. M. Lim, S. C. Chong, and C. P. Lee, "Three-dimensional shape generation via variational autoencoder generative adversarial network with signed distance function," *International Journal of Electrical and Computer Engineering*, vol. 13, no. 4, pp. 4009–4019, 2023, doi: 10.11591/ijece.v13i4.pp4009-4019.
[53] K. Genova, F. Cole, D. Vlasic, A. Sarna, W. Freeman, and T. Funkhouser, "Learning shape templates with structured implicit functions," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 7153–7163, doi: 10.1109/ICCV.2019.00725.
[54] E. Remelli *et al.*, "MeshSDF: Differentiable iso-surface extraction," *Advances in Neural Information Processing Systems*, vol. 2020-December, 2020.
[55] W. Wang, Q. Xu, D. Ceylan, R. Mech, and U. Neumann, "DISN: Deep implicit surface network for high-quality single-view 3D reconstruction," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

BIOGRAPHIES OF AUTHORS

Shyngys Adilkhan is a lecturer at SDU University in Kazakhstan. He also conducts practice lessons and works as a head of the Multimedia Laboratory in the Department of Information Systems, School of Information Technologies and Applied Mathematics. His research interests lie in 3D technologies, game development, virtual reality, and gamification. He received his M.Sc. in Computer Science at SDU University and has publications on Gamification that were indexed in Scopus. He can be contacted at email: shyngys.adilkhan@sdu.edu.kz.

Madina Alimanova is an Associate Professor at the School of Information Technologies and Applied Mathematics, SDU University, Kazakhstan. She holds a Ph.D. in Materials Science and has extensive experience in academic leadership, digital education, and interdisciplinary research. Two years ago, she completed a year-long fellowship as a Visiting Fellow in the Leaders in Higher Education (LHE) program at the Department of Computer Science and Electronics, University of Essex, UK. Throughout her career, she has served in pivotal roles such as Head of the Department of Information Systems, founder and head of the university's 3D Laboratory, and co-creator of a new bachelor's program in Multimedia Sciences. Despite her administrative positions, she has been conducting research in computer and multimedia sciences and gamification for the last ten years. She has authored over 30 scientific publications and has played a key role in organizing several national and international conferences, including the IEEE ICECCO series and other academic events in the region. She is an active member of the Women in Tech global movement, where she contributes to empowering women in science and technology through mentorship, outreach, and academic leadership. In recognition of her contributions to science and higher education, she has received multiple state awards and certificates. She is widely respected for her innovative teaching methods and is often described by her students as both an inspiring educator and a life mentor. She can be contacted at email: madina.alimanova@sdu.edu.kz.

Lei Shi is an Associate Professor in the School of Computing at Newcastle University, where he directs the Haii Lab and is a member of the interdisciplinary Open Lab. His research focuses on enhancing human-AI interaction by integrating human-computer interaction principles with advanced AI technologies, emphasizing ethical, legal, and societal considerations. He holds leadership roles as Deputy Degree Programmed Director for the MSc Digital Technology Solutions Degree Apprenticeship (Software Engineering Specialist) and as Postgraduate Research Admissions Tutor. Before joining Newcastle, he held academic positions at Durham, Liverpool, and Warwick universities. Actively engaged in the academic community, he serves on the managing committee of the European Association of Technology-Enhanced Learning (EATEL) and will be the Local Chair for ECTEL2025. He has previously contributed to conferences such as ECTEL2024, AIED2022, EDM2022, IoT2022, and AIED2023 in various organizational roles. He can be contacted at email: lei.shi@newcastle.ac.uk.

Aiganym Soltiyeva is a Lecturer at the Department of Information Systems, School of Information Technologies and Applied Mathematics, SDU University, Kazakhstan. She is a Ph.D. student in Computer Science at SDU University and also had a scientific internship at Tampere University, Finland. Her research interest is using virtual reality technology for the treatment of children with autism spectrum disorder. She holds a Bachelor's degree in "Computer Science and Software" and a Master's in "Digital Media Technology" from Kazakh-British Technical University, Kazakhstan. For her undergraduate thesis project, she created with her team a Virtual Reality training simulator for Oil and Gas Field workers. She has more than 10 years of experience working with Computer Graphics and Virtual Reality. Involved in creating graphic materials for the International Exposition "Expo Astana 2017," and participated in international conferences, in creating animated movies, video clips, and ads for Kazakhstan media. She has published papers and articles in journals in the field of Educational Technologies. She can be contacted at email: aiganym.soltiyeva@sdu.edu.kz.