*applied sciences*

MDPI

*Article*

# A Generative Deep Learning Approach for Improving the Mechanical Performance of Structural Components

**Nurullah Yüksel * and Hüseyin Rıza Börklü**

Department of Industrial Design Engineering, Gazi University, 06500 Ankara, Türkiye; rborklu@gazi.edu.tr
* Correspondence: nurullahyuksel@gazi.edu.tr; Tel.: +90-3122028502

**Abstract:** This study aimed to improve the mechanical properties of 3D concept designs by combining the design capability of a generative adversarial network with finite element analysis. This approach offers an innovative perspective on the conditioning of generative models while improving design properties and automation. A new design and evaluation framework has been developed for GAN models to generate 3D models with improved mechanical properties. The framework is an iterative process that includes dataset generation, GAN training, and finite element analysis. A "joint" component used in the aerospace industry is considered to demonstrate the proposed method's effectiveness. Over six iterations, an increase of 20% is recorded in the average safety factor of the designs, and the variety of designs produced is narrowed in the desired direction. These findings suggest that the direct generation of structural components with generative models can expand the potential of deep learning in engineering design. Another innovative aspect of this study is that it provides a new option for the conditioning of data-dependent generative design models.

**Keywords:** generative adversarial networks (GANs); finite element analysis (FEA); parametric design; mechanical properties; additive manufacturing (AM); 3D printing



**Citation:** Yüksel, N.; Börklü, H.R. A Generative Deep Learning Approach for Improving the Mechanical Performance of Structural Components. *Appl. Sci.* **2024**, *14*, 3564. https://doi.org/10.3390/app14093564

Academic Editor: Marek Krawczuk

Received: 29 March 2024
Revised: 17 April 2024
Accepted: 18 April 2024
Published: 23 April 2024



## 1. Introduction

Machine learning models have recently been used in product and engineering design in almost every field [1–3]. In particular, developing and using deep-learning-based generative design methods has accelerated design processes such as inspiration, idea generation, concept generation, evaluation, and optimization [4,5]. By using these methods, better and comprehensive designs can be created in a very short time and with minimal designer intervention. Generative deep learning models can include all design properties within a feature space, a simple representation of the design space. The feature space makes it possible to generate new concepts by capturing the critical points of the design [6].

A generative adversarial network (GAN) is one of the current data-driven generative design systems. The ability of the GAN to generate new data (text, images, 3D models, etc.) is utilized in design studies. One of the aims of this study was 3D concept generation. Since 3D concepts contain more visual and physical features than 2D concepts, the level of detail is higher. This makes data-driven generative design research that generates 3D models critical [7]. Classical GAN models can discover form features depending on the design visualization. However, a good design representation should consider form and function features. The form includes the geometric and topological properties of the product. Function refers to the design purpose of the product. Designers can understand the relationship between form and function. However, it is difficult for artificial intelligence to grasp such complex relationships. For this reason, generative design studies generally aim to produce new forms. The concepts created here are far from being functional/structural workpieces and can only be used with additional modifications and improvements. These extensive challenges constitute a new and broad area of research.

In this study, a generative deep learning model and finite element analysis are combined to evaluate the form and function properties of the generated designs. While 3DGAN generates new data, finite element analysis (FEA) is used to evaluate the mechanical properties of the generated data. At the end of the evaluation, design options that are good in terms of form, weight, and strength are added back to the GAN training set. Thus, better designs improve the quality of the training set, and the new 3D models produced have enhanced functional properties. The main objective of this study was to develop an autonomous approach to improving the functional properties of 3D models generated by deep-learning-supported generative design method. In that context, all design components that transfer load, force, or power were evaluated by FEA and redesigned if necessary. Unlike other studies, this study combined data-driven generative design and FEA methods for the first time. Thus, the mechanical properties of structural components generated by 3DGAN were analyzed.

Another innovative aspect of this study is that it provides a new option for the conditioning of data-dependent generative design models. These models are known to generate data that reflect the distribution of the training dataset. With repeated design and evaluation steps, the training dataset's general properties are improved, and high mechanical properties are achieved for the generated data. By conditioning the mechanical properties of generative models, this can create a potential area for deep learning studies in engineering design.

## 2. Literature Review

Idea generation, which starts with problem identification in the early design phase, is the process of creating new, creative, and useful solutions [8,9]. Using deep learning methods in this process reduces design costs and enhances innovation [10–12]. In the literature, the number of studies involving idea and concept generation and 3D model generation based on deep learning models is rapidly increasing [13,14]. In one of these studies, Raine et al. proposed an approach to imitating human design strategies using deep learning in the idea generation process [15]. This image-processing-based approach aims to predict the next design step in problems where rules and strategy are ambiguous. Li and McComb [16] introduced a modified super-resolution generative adversarial network (SRGAN) model incorporating a physics-informed loss function to boost multiphase turbulent fluid flow simulations. This model shows that both traditional and physics-informed models outperform standard sampling methods, but the additional complexity of the physics-informed approach does not significantly increase accuracy. In another generative design study, Yu et al. [17] developed the DesignGAN model inspired by biological objects. This model generates new biologically inspired images by combining target object and biological object images. Yuan and Moghaddam created the design-attribute GAN (DAGAN) model using a new loss function and a different discriminative loss mechanism. The DAGAN model allows for modifying only the desired design features, thus enabling efficient fashion product creation [10,18]. In another study, Chen et al. [19] presented the Generative Adversarial Network-based Design Under Uncertainty Framework (GAN-DUF), for efficiently modeling and quantifying geometric uncertainties in engineering designs. This framework has effectively learned design variability without assuming specific distributions, demonstrating improved design optimization and robustness in real-world applications.

On the other hand, generative models can create designs with balanced engineering properties [20–23]. Oh et al. [20] integrated topology optimization into boundary equilibrium GANs (BEGANs) to obtain aesthetic product designs with high engineering performance. The training data required for the success of the GAN models were obtained by topology optimization to increase the diversity and quality of the training set. The framework, which consists of exploration and evaluation steps, aims to achieve aesthetic and feasible product designs. In a similar study, Yoo et al. [24] propose a deep-learning-based framework that automatically generates 3D CAD models at the conceptual design stage and evaluates their engineering performance. This framework allows the engineering performance of the generated 3D concepts to be predicted at the early design stage. In another engineering design study, Chen and Fuge [25] presented a generative model for generating real-world designs bounded by one or more parts. This model generates part designs individually through a low-dimensional representation based on specific master parts. The model was verified on various design examples. Lee et al. [26] presented a method that uses GANs to simplify 3D CAD models of mechanical parts using a training dataset. This method effectively simplified the models while preserving essential features for the target domain. In another study, Chen and Ahmed [27] developed a model called Multi-Objective Performance Augmented Diverse Generative Adversarial Network (MO-PaDGAN) to solve multi-objective optimization problems in engineering design. This model has shown a 180% better performance than other methods in real-world applications thanks to adding a special loss function to the GAN to increase design diversity and quality. Nobari et al. [28] proposed the Continuous Conditional Diverse Generative Adversarial Network (PcDGAN) model to generate new designs that meet the targeted performance requirements. The authors stated that the design space coverage capability of PcDGAN is higher than that of standard GAN models. In another study, Nobari et al. [29] proposed a Range-Constrained Generative Adversarial Network model to meet engineering design constraints. This proposed model aims to generate data that satisfy the targeted design constraints with a self-expansion approach and a new loss function. In another training success improvement study, Giannone et al. tried to find a solution to the problem of generative models generating faulty geometries. The proposed new training method aims to outperform the generative model by utilizing erroneous data points [30].

Another field where generative design methods have been applied is that of design options' evaluation and optimization process [31,32]. In one of these studies, Behzadi and Iliesh [33] proposed a convolutional neural network (CNN)-based transfer learning model to overcome the computational costs of topology optimization. The authors aimed to process high-resolution 3D design space in real-time with fewer samples than traditional deep learning methods. In another study, Rawat and Shen [34], addressing high computational costs, created a model that can generate 3D geometries with an approach based on Wasserstein generative adversarial networks (WGANs). This model demonstrates the potential of deep learning to significantly reduce iterative optimization processes in design. Shu et al. [6] developed a functional generative design model using GANs and computational fluid dynamics (CFD) methods. The authors introduce an iterative process of CFD evaluation of 3D aircraft models generated by GANs. This results in good aircraft designs with a low drag coefficient. Although the presented approach obtained coarser results than topology optimization methods, it converged faster. In another design evaluation study, Wu et al. [35] proposed a CNN model that predicts the success of a product from product sheets generated for design competitions. In another study, Oh et al. [20] aimed to produce aesthetic wheel designs with high engineering performance by integrating topology optimization and boundary equilibrium GANs (BeGANs). In this study, the data obtained from topology optimization were incorporated into the training data to produce high-performance models. The authors claim this approach provides lower-cost optimization with minimal design and engineering domain knowledge.

There have been many studies in the literature on incorporating generative deep learning methods into the engineering design process. These studies offer various applications, from idea and concept generation to design optimization. The potential of GANs in the generative design process is remarkably emphasized in these studies. However, these studies focus predominantly on the aesthetic aspects of design, often overlooking the integration of functional or mechanical properties. This creates a gap in the application of these systems to fields where functionality is as crucial as design, such as in engineering and industrial design. Additionally, while these papers demonstrate the capabilities of generative models like GANs, they often do not fully leverage the potential of these models for improving the mechanical performance of the generated designs. Therefore, more studies are needed that aim to improve the structural strength performance of workpieces produced with GANs. Accordingly, the design framework proposed in this study verifies and enhances the structural strength of innovative and creative design options created with GANs using FEA. An autonomous approach has been developed to add functional features to deep-learning-supported generative design methods. Within the study context, generative design and FEA methods are combined with data-dependent generative design to perform long/laborious analysis–redesign processes in a shorter time and without human intervention. Another difference in this study from the existing studies in the literature is that the concepts generated are real structural components. Generating structural components using the GAN model provides direct applicability in engineering design. This increases the efficiency and effectiveness of engineering design processes by accelerating the translation of theoretical and experimental studies into practical applications. Finally, the fact that the dataset to be used in this study was generated by a parametric design process provides a new way for the proposed framework to cover other engineering problems.

## 3. Materials and Methods

Figure 1 presents a flowchart outlining the path to be followed and the methods to be used to automate functional 3D model generation. The process begins with the generation and preparation of a diverse synthetic dataset through parametric modeling. Following this, the GAN model is trained to generate realistic 3D structures, with extensive testing to optimize performance. Subsequently, the models undergo finite element analysis to validate their mechanical integrity. The final stage involves physically producing the most promising designs using additive manufacturing techniques. The aim of this cycle is to improve mechanical properties through the design, production, and evaluation cycle in the first three stages.

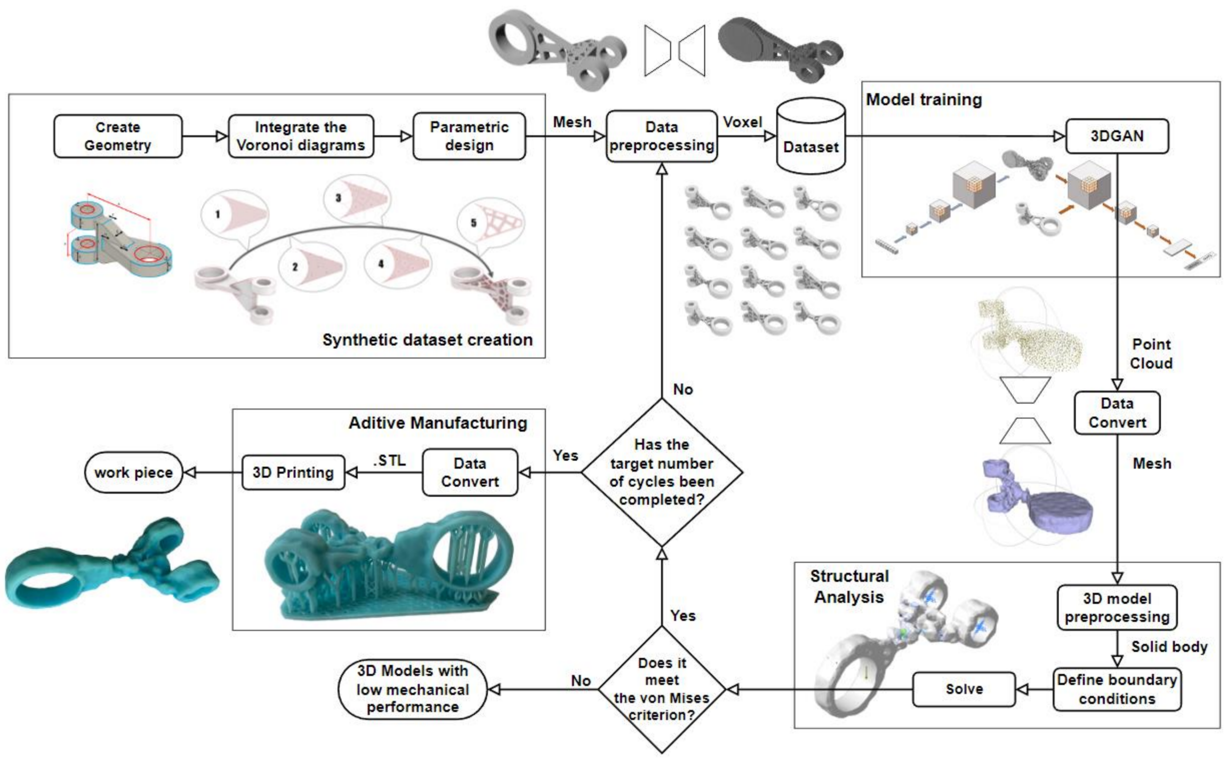



**Figure 1.** Flowchart depicting the process of creating functional design options.

### *3.1. Creating a Dataset and Performing Preparation*

Deep learning methods are AI algorithms that learn patterns from the training set and use them for image processing, object recognition, and generating new data. Therefore, creating and compiling training sets is the main factor in deep learning model success.

However, 3D model sets are limited for reasons such as differences in model representation methods [35].

Three basic representation methods are used in 3D object recognition and model generation processes: voxel, point cloud, and polygon mesh. Volumetric pixels used to represent a point in 3D space are called voxels. Point cloud is a data representation method formed by the points on the surface of a 3D structure. Each point is represented by coordinates. Point cloud model representations are often preferred in deep learning model training due to their easy modeling and low computational cost [36,37]. Finally, polygon mesh is a collection of vertices, edges, and surfaces that define the surfaces of 3D models. It is widely used in applications requiring a realistic appearance and detailed representation [38]. These representation methods have various advantages and limitations. For example, point clouds offer low computational cost and realistic representations but do not provide complete surface description information. Polygon meshes provide detailed surface information but may be less computationally efficient. On the other hand, while voxel-based representations perform well in internal structure descriptions, they are not compatible with applications where high surface quality is required. Therefore, the representation method to be used is determined according to the needs of the process to be applied [36]. In addition, different data representation methods can be used at various stages of an application or project.

Another limitation of 3D model recognition and generating studies is the difficulty of accessing accurate, diverse, and high-quality data. Although popular 3D data repositories such as Princeton shape benchmark, McGill 3D shape benchmark, 7 scenes, ModelNet, and ShapeNet contain 3D models of various classes and sizes, they are quite insufficient in terms of having structural component (bracket, hinge, joint, arm, valve, etc.) models [39]. The problem of data deficiency, which reduces the effectiveness of generative design models in engineering design, can be overcome with synthetic datasets created with various algorithms and software. For this purpose, a synthetic dataset was created with an algorithm combining Voronoi diagrams and parametric design. Parametric design is an interdisciplinary field that combines mathematical operations, algorithms, and artificial intelligence with design processes [40,41]. This approach provides flexible and efficient solutions by automatically and rapidly updating design variants. Although the dataset creation capability of parametric design gives successful results, additional algorithms and techniques are required to increase the quality and diversity of the generated synthetic datasets.

One of these techniques, Voronoi diagrams, uses a diagram that divides an area or volume into sections according to specific focal points. These partitions include all points closer to the focal point than any other focal point [42]. Given n focal points in metric space, the Voronoi diagram allows the space $S = \{s_1, s_2, \ldots, s_n\}$ to be partitioned into n regions. Each Voronoi cell $V(s_i)$ consists of all points in space ($S$) closer to $s_i$ than any other focal point. The formal expression of a Voronoi cell is shown in Equation (1). Here, $d(x, y)$ denotes the distance function between points $x$ and $y$, while $S$ denotes the initial set of points.

$$V(s_i) = \{x \mid d(x,\, s_i) < d(x,\, s_j)\ \forall\, s_j\ \in\ S,\ j \neq i\,\} \tag{1}$$

This study integrates Voronoi diagrams into the parametric design process to create 3D models that respond to design requirements and constraints [43]. This combination offers advantages such as increasing design efficiency, generating unique and problem-specific designs, and exploring large design spaces. Voronoi diagrams, known for their ability to generate complex, natural-looking patterns, introduce an element of organic randomness into the design process. These diagrams partition a space based on the proximity to a set of predefined points, resulting in a mosaic of polygonal cells with unique, non-repetitive patterns. When these diagrams are integrated into parametric models, they imbue the structured and deterministic nature of parametric designs with an element of unpredictability and complexity. This results in designs that are not just variations of a theme but are genuinely diverse, offering a richer training ground for GANs.

### 3.2. Generative Model Training

Generative adversarial networks (GANs) are unsupervised deep learning models introduced by Ian Goodfellow et al. in 2014 [44]. A classic GAN model contains two adversarial neural networks, a generative (G) and a discriminative (D) network. The generator network generates fake data, while the discriminator network aims to separate fake data from real data. 3DGAN models, on the other hand, can directly create 3D data representations using 3D datasets [45,46]. 3DGAN models not only improve the capabilities of generating realistic 3D data samples but also significantly contribute to deep learning by addressing the increasing importance and complexity of data representation [47,48].

3DGAN architectures consist of two main components, a generator (G) and a discriminator (D), similar to the classical GAN architecture. While the generator generates synthetic 3D data representations, the discriminator distinguishes the generated samples by comparing them to real 3D data. This process basically consists of noise sampling, generating fake data, distinguishing between real and fake data, and updating network weights. In the noise sampling step, points from a normal or uniform distribution are converted into a noise vector to obtain complex data structures. The second stage of 3DGAN is the generation of fake data with a generative network consisting of deep learning layers. Here, fully connected layers, convolutional transpose layers, normalization layers, and activation layers are used to up-sample the noise vector to produce 3D representations. The third stage of 3DGAN model training involves the discriminator network where real and fake data are distinguished. Here, convolutional, normalization, fully connected, and activation layers are used to determine the spatial relationships between the points forming the 3D data. In the final stage of the training process, to improve the performance of the generator and discriminator networks simultaneously, the network weights are updated using the conventional loss function given in Equation (2) [44,49]. Here, the $E_x$ term measures the discriminator's success in classifying data, while the $E_z$ term evaluates the generator's ability to produce synthetic data. Thus, it aims to maximize the discriminator network's discrimination ability and minimize the generator error. In this equation, $x$ represents the real images, and $z$ represents the noise input.

$$Minmax(D,\ G) = E_x[log(D(x))] + E_z[log(1 - D(G(z)))] \tag{2}$$

### 3.3. Design Evaluation

A vital shortcoming of data-driven generative design methods (e.g., GANs) is the inability to test the suitability of the resulting designs for the design purpose. Generative models are trained using the probabilistic distribution of the training set. Therefore, the generated data reflect the general characteristics of the training dataset in terms of form and function. For generative models to generate designs with superior functional properties, the training data must have superior design properties. The concept of functionality in structural components can be defined as the load carrying capability of components during operation. Physics-based approaches such as finite element analysis (FEA) are used to determine a component's functional (strength) properties in engineering design. FEA is a numerical analysis method that attempts to predict the mechanical behavior of the part and assembly under operating conditions [50].

In this study, a static structural analysis was performed on the workpieces using CATIA. The models were discretized with tetrahedral elements using the patch confirming technique, ensuring that the mesh size did not affect the results, and the mesh was refined and validated until variations were reduced to less than 8%. The analysis includes a comprehensive elasto-plastic analysis of strain behavior beyond the elastic limit using the Newton–Raphson method detailed in Equation (3):

$$\left[K^T\right]\{\Delta u\} = \{F\} - \{F^{nr}\} \tag{3}$$

where $K^T$ is the stiffness matrix, $\Delta u$ is the displacement matrix, and $\{F\} - \{F^{nr}\}$ is the force instability matrix.

This study focused on the post-yield behavior of AlSi10Mg materials modeled with a bilinear stress–strain relationship to reflect hardening after the yield point. Material properties included a yield strength of 270 MPa and a tangent modulus of 5 GPa. Boundary conditions fixed the workpieces in small holes, and a 150 N bearing force was applied to the large hole.

In this study, we aimed to develop design concepts with a high strength-to-weight ratio using a data-driven generative design method. The generated designs were evaluated in a design-simulation environment, and their mechanical properties were determined. The von Mises values of all concept designs under the same conditions (boundary conditions, mesh size, material properties, load, etc.) were determined autonomously. Von Mises is the equivalent stress, which shows how much stress a material can carry without exceeding the elastic limit.

A preliminary mesh optimization study was performed on a subset of approximately 10 samples to determine the optimal mesh size. This preliminary study involved repeating the analysis with different network sizes and observing the results until a point was reached where the analysis results were independent of network size. The mesh size that met this criterion was then used equally across all designs in our automated analysis process.

In this study, the design concepts analyzed were ranked according to their von Mises values, and the designs that fell in the 50% percentile were determined. High-performance concept designs were converted into voxel format to be added to the training set. A random 3D model was removed from the training set as much as the data were added to the initial dataset. Thus, the average functional performance of the training dataset was increased. Creating the training set, training the GAN model, evaluating the generated design concepts, and adding high-performance concepts to the training set constituted the functional generative design cycle. This cycle did not diminish the diversity of the dataset; rather, it enhanced the dataset by incorporating designs that not only exhibited aesthetic and geometric variety but also demonstrated improved mechanical properties. The cycle continued until part of the training set consisted of GAN-generated models. Thus, design concepts with high strength could be generated.

### *3.4. Additive Manufacturing (AM)*

Advances in additive manufacturing (AM) technology have significantly spurred on the development of generative design algorithms. AM techniques enable the production of parts with complex geometries that are beyond the capabilities of conventional manufacturing methods. Notably, design concepts generated by data-driven generative design methods can incorporate complex geometries. However, the manufacturability of such intricate designs often poses challenges when using traditional manufacturing techniques. As a result, these advanced design concepts are typically only feasible through AM technologies, which provide the necessary flexibility and precision to accommodate their complexity.

Within the scope of this research, in the last step of the generation and evaluation cycle, the concept designs were classified according to their weight and strength properties. Then, we aimed to manufacture the ten design concepts with the best performance. In this process, the defects (if any) on the parts were adjusted in the CAD environment. The voxel model was converted into the STL format and made suitable for printing. Concept designs can be manufactured by choosing printer settings according to the desired precision and print quality. Although the production of parts in layers is a familiar working principle in AM techniques, there are differences between the methods according to the materials and printing technology used [51,52]. In this study, the performance of AM techniques in manufacturing parts generated by generative design was evaluated by the stereolithography (SLA) and fused deposition modeling (FDM) methods.

SLA is a 3D printing method in which 3D objects are manufactured layer by layer using special photopolymer resins that solidify when exposed to ultraviolet light. In this method, the scanning system creates layers by directing the light to the surface to form a specific cross-section. With SLA, it is possible to produce exact and detailed parts. The other method, FDM, is one of the most commonly used methods in 3D-printing technology. This method involves creating a 3D object by layering material in filament form through a heated nozzle following a special design [53].

## 4. Application

A case study was carried out to verify the validity of the presented functional generative design approach. In this context, a special joint component that helps to steer a helicopter by providing freedom of movement to the rotor was considered. The location, mobility, and original design of this component in the assembly are shown in Figure 2. In the first step of the application, a synthetic dataset was created, covering a wide design space by integrating the Voronoi diagram into the parametric design. A 3DGAN model was trained using this synthetic dataset in the second step. Various data preparation procedures were applied to determine the mechanical properties of the pseudo data generated by this model. In the third step, CATIA V5, a design and analysis software, was used to evaluate the mechanical performances of the 3D components. As a result of this evaluation, designs that were superior according to the von Mises performance criterion were included in the training dataset. The same amount of data from the initial dataset was removed from the process, and the training and analysis cycle was restarted. After the process was completed, the manufacturability of the generated designs was evaluated by an AM method.

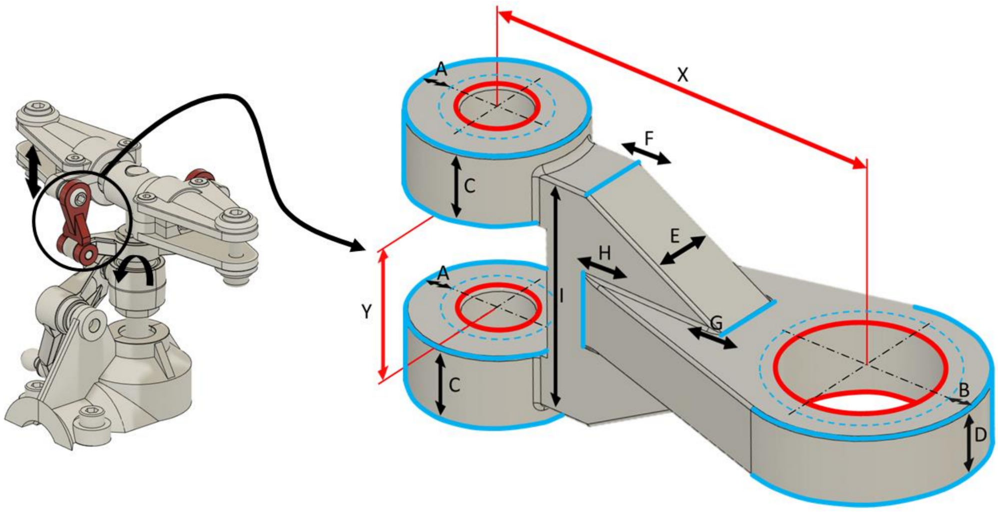



**Figure 2.** Position of the considered structural component in the helicopter rotor assembly and parametric model.

### *4.1. Creating a Dataset and Performing Preparation*

3DGAN model training requires datasets that are quite different and contain a large amount of data. This study used a synthetic dataset created by the authors. The main reason for using synthetic data is the insufficiency of real 3D model data for structural components. A parametric model of the structural component was assembled to create the synthetic dataset. In this model, parameter value ranges were determined by considering geometric boundary conditions. Figure 2 presents a parametric model of the structural components, showing the design parameters and boundary conditions. While the features shown in red on the part have fixed values, the features in blue can change parametrically. The determined parameters and boundary conditions are shown in Table 1. With the nine parameters used here, over 60 billion ($8 \times 11 \times 16 \times 8 \times 9 \times 16 \times 51 \times 76 \times 11$) design variants can be created. However, these concepts are insufficient for deep learning training regarding data diversity. Although each of these designs is different from the other, their general geometries are quite similar to each other. Therefore, Voronoi diagrams were included in the parametric design process to increase data diversity. Applying Voronoi diagrams to the areas determined on the component created 4635 unique design variants. This dataset consists of models with different wall thicknesses, element sizes, Voronoi cell sizes, and Voronoi cell numbers. These variables help the dataset to mimic real data diversity. The parametric design process and some of the data obtained are shown in Figure 3. The dataset creation process was described in detail in a previous study prepared by the authors. The general process is summarized here.

**Table 1.** Parameter value ranges.

| Parameter Name | Min. Value (mm) | Max. Value (mm) | Range of Change |
|---|---|---|---|
| Parameter A | 8 | 15 | 8 |
| Parameter B | 5 | 15 | 11 |
| Parameter C | 10 | 25 | 16 |
| Parameter D | 3 | 10 | 8 |
| Parameter E | 3 | 11 | 9 |
| Parameter F | 5 | 20 | 16 |
| Parameter G | 0 | 50 | 51 |
| Parameter H | 15 | 90 | 76 |
| Parameter I | 0 | 10 | 11 |

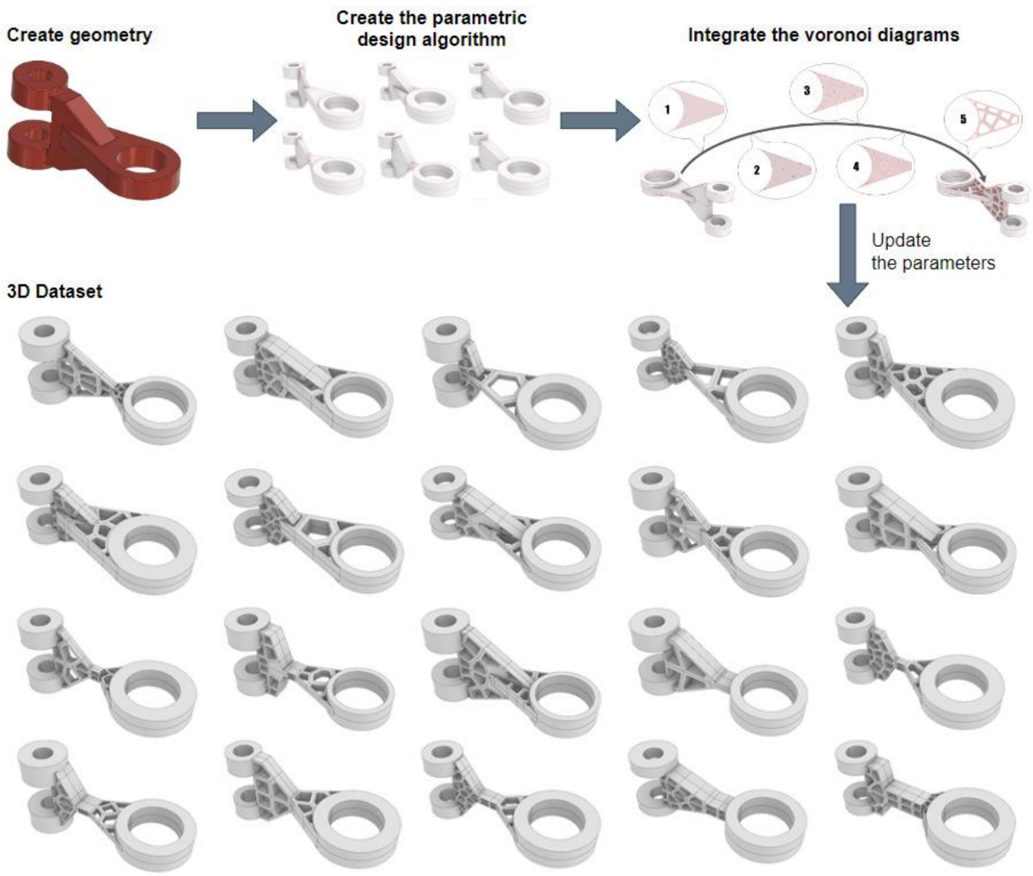



**Figure 3.** Synthetic dataset creation process and samples from the created dataset.

First of all, the area on which the Voronoi diagram would be applied was determined. In the second stage, Voronoi points were randomly placed in this predefined area. The distribution and number of points determine the size and shape of cells, creating diversity. In the third stage, Voronoi cells were created from point positions. In the fourth stage, the basic geometry for the hollow structure was created by offsetting each of the resulting Voronoi cells by a fixed value. This offset distance can be adjusted parametrically to control the detail and complexity of the design. In the final stage, the volumes between the walls of the Voronoi cells were evacuated and a unique structure with cavities was designed.

For the training of the deep learning model, the 3D model dataset was subjected to certain processes. In the first stage, faulty or missing data were identified and removed from the dataset. Then, all data were standardized to a standard size to improve the performance of the generative design. Using the "binvox" library on Python 3.10, the models in OBJ format were converted into voxel format, which expresses an object as a collection of unit cubes. Since voxels can represent complex internal structures, they were chosen as the appropriate data representation method for the structural component data generated in this study. However, part resolution depends on the amount/size of voxels. For this study, the ideal grid size was determined as 128 × 128 × 128, depending on the computer processing capacity.

Data representation methods appropriate for the specific demands of each phase of this study were strategically selected, taking into account their different advantages and limitations. Voxels were utilized during the 3DGAN training phase due to their superior ability to represent complex internal structures. Voxels, as three-dimensional units, are effective in capturing the volumetric detail necessary for modeling complex designs. However, their suitability is reduced for applications requiring high surface quality; this limitation is addressed in the next steps. Point clouds were used to reduce the computational cost after training. Point clouds are effective at compressing large volumes of data while preserving vital geometric details. Also, the mesh representation method was used to edit the part in the CAD environment. Meshes are advantageous in terms of detailed surface information and increase the accuracy of the overall design. The choice of these representation methods allowed us to efficiently balance computational efficiency, internal structural details, and surface quality.

### *4.2. Generative Model Training and Data Generation*

In this study, a voxel-based 3D deep-convolutional generative adversarial network (3D DCGAN) model developed by Wu et al. was used [45]. This model can generate realistic 3D objects from a low-probability distribution. In the first stage of training this model, the probabilistic point distribution is converted into a noise vector. Then, this noise vector is upscaled to the real object size by the generator network [38]. Meanwhile, the discriminator network tries to distinguish between the generated object and the objects coming from the dataset. The ultimate goal of the generator and discriminator network training process is first to create more realistic fake data and then to be able to distinguish them from real data [49].

In this context, a balance must be struck between optimal model quality and the requirements for computation and memory. In practice, this balance is often adjusted based on the available GPU capacity and the desired level of detail in the model. Figure 4 displays three sets of 3D model variations at different resolutions (64 × 64 × 64, 96 × 96 × 96, and 128 × 128 × 128). Each set illustrates how the geometric fidelity of the generated models improves with increasing resolution. In the first set, at the lowest resolution (64 cubic units), the models created with 15,000 data points are quite rough and lack detail. In contrast, the models in the last set, at the highest resolution (128 cubic units) with 4,600 data points, exhibit much more detailed and accurate geometries.

However, increasing the resolution significantly raises computational costs and can challenge the graphics processing unit's (RTX 3090) capacity. This is a particular constraint for large-scale GAN training as higher resolutions demand more memory and processing power. The "number of data" in the image refers to the number of unique models used at each level of resolution, where decreasing numbers reflect an increased computational load due to a higher resolution.

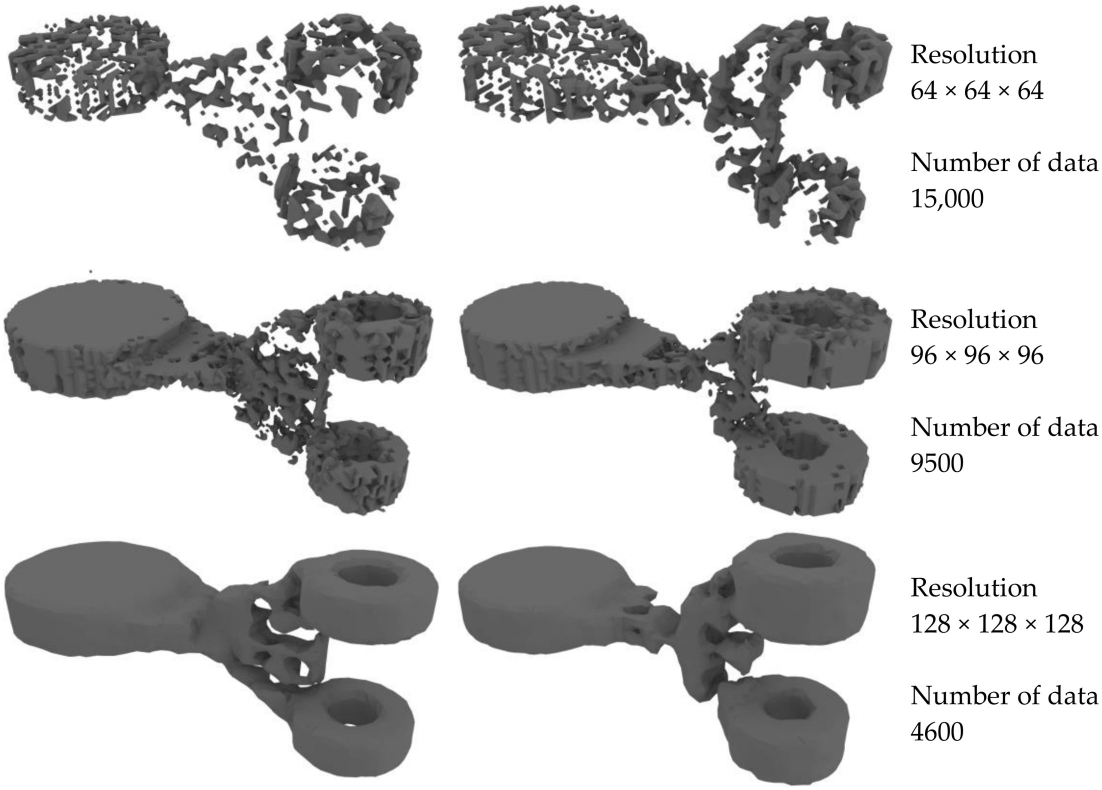



**Figure 4.** Comparison of GAN-generated 3D models between resolutions.

The determination of hyperparameters, including the number of epochs, data size, pool size, and noise vector dimensions, was achieved through a meticulous trial-and-error process. For each hyperparameter, the training process was iteratively repeated, enabling a careful evaluation of its impact on the model's output and overall performance. Similarly, the network architecture was refined based on the quality of the produced parts. Various architectural configurations were rigorously tested, with their effectiveness in generating high-quality designs thoroughly assessed. This methodical approach facilitated precise fine-tuning of the model, ensuring it was adeptly tailored to meet the specific requirements of the application. The network architecture and hyperparameter values used are given in Table 2.

In this study, the "PyTorch" library was used to create a 3D DCGAN architecture and to increase the training process efficiency. A 200-dimensional noise vector was transformed into a 3D matrix in the first stage. The size of this matrix was gradually expanded by four transpose convolution layers in the generating network. A matrix with a higher dimension than the previous layer was obtained in each layer. In the last layer of the generative network, a fake 3D model was generated by a $128 \times 128 \times 128$ matrix. The discriminator network transformed the $128 \times 128 \times 128$ matrix into a one-dimensional array using feature maps of different sizes in four convolution layers. By reducing the matrix size in each layer, the generalization capability of the network is increased while the computational cost is reduced. This matrix was converted into a one-dimensional array to train fully connected (FC) layers. In the last stage of the discriminator network, the Sigmoid activation function was used to predict the class of the input matrix (fake or real), and the process was completed. Batch normalization and activation (i.e., LeakyReLU) functions were implemented between the convolution (transpose and normal) layers in both network architectures. The parameters used in the network architecture, such as the number of convolution layers, number of parameters, batch size, hidden vector size, number of epochs, and learning rate, were determined heuristically. The main limitation in determining these parameters was the graphics processor capacity. After 250 epochs (approximately 9 h), new 3D models were obtained, as shown in Figure 5.

**Table 2.** Architecture of the 3D DCGAN: layer configuration and hyperparameters for the generator and discriminator.

| | Layer | Output Shape | Activation Function | Hyperparameter |
|---|---|---|---|---|
| Generator | Input | [200, 1, 1, 1] | - | Epoch: 250<br>Resolution: 128 × 128 × 128<br>Batch size: 64<br>Noise vector size: 200<br>Generator learning rate: 0.00015<br>Discriminator learning rate: 0.0001 |
| | Layer 1 | [256, 4, 4, 4] | ReLU | |
| | Layer 1 | [128, 12,12, 12] | ReLU | |
| | Layer 1 | [64, 28, 28, 28] | ReLU | |
| | Layer 1 | [32, 64, 64, 64] | ReLU | |
| | Output | [1, 128, 128, 128] | Sigmoid | |
| Discriminator | Input | [1, 128, 128, 128] | - | |
| | Layer 1 | [32, 64, 64, 64] | LeakyReLU | |
| | Layer 1 | [64, 32, 32, 32] | LeakyReLU | |
| | Layer 1 | [128, 16, 16, 16] | LeakyReLU | |
| | Layer 1 | [256, 4, 4, 4] | LeakyReLU | |
| | Output | [1, 1, 1, 1] | Sigmoid | |

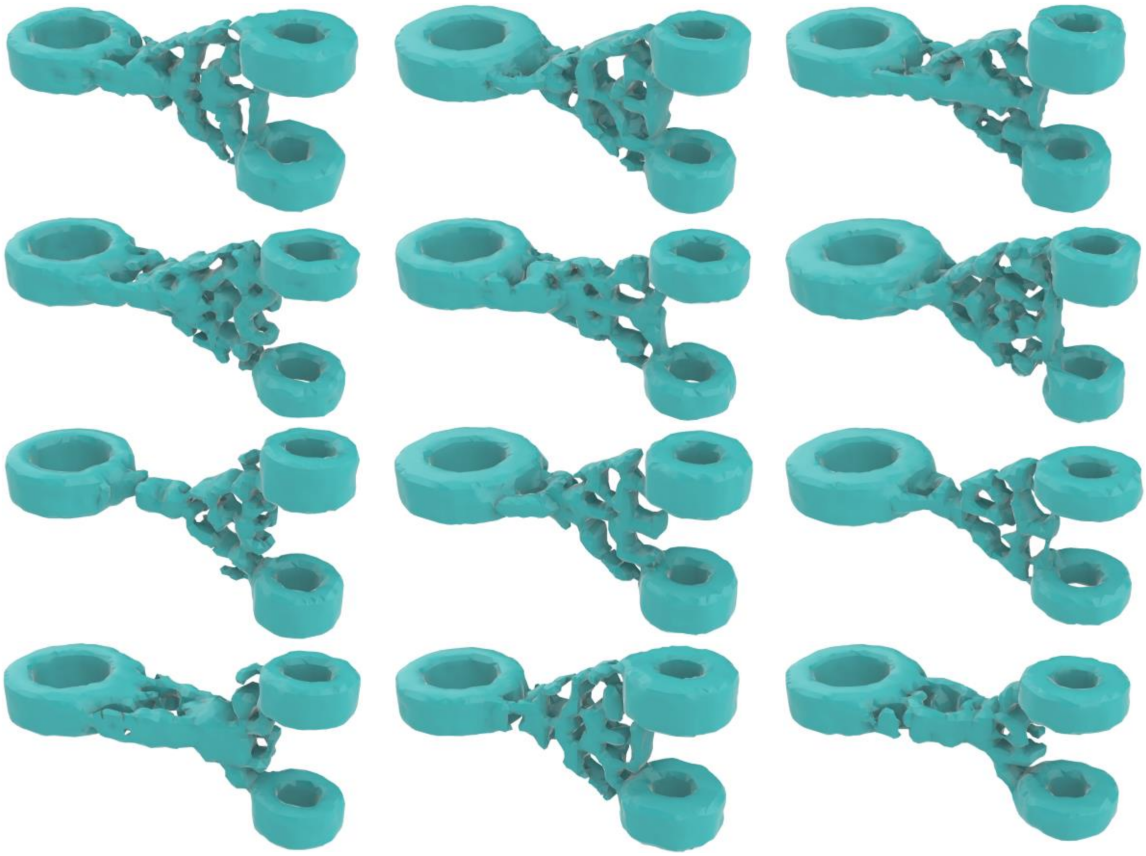


**Figure 5.** Samples of new workpiece (joint) generated by the 3DGAN model.

These data in voxel format were converted to the mesh format for editing in a CAD environment and to reduce the file size. Some editing and refinement operations, such as reducing the number of points, removing noise, merging close points, removing duplicate faces, scaling, and filling holes, were performed on the generated mesh. Then, the obtained mesh-formatted data were saved in STL format. These processes were performed with automation using the "pymeshlab" libraries on Python.

### 4.3. Design Evaluation

The functionality of structural components was evaluated concerning von Mises values. Von Mises stress is a scalar value used to predict the yields of materials under multi-axial loading conditions. If this value exceeds the yield strength of the material, plastic deformation starts. The concept components generated with the generative design model were autonomously analyzed with a script in the CATIA V5 software, and von Mises values were determined. The macro script prepared for this purpose converts the mesh format workpieces into a solid model and performs structural analysis by defining the boundary conditions.

Firstly, mesh-formatted data are taken into the CAD environment, and surface defects are removed. In the next stage, mesh structures are converted into surface and solid models, respectively (Figure 6). This solid model is transferred to the simulation module after holes are drilled in the part. At this stage, structural analysis is performed under a linear force with standard boundary conditions (fixed supports on small holes, bearing load on big hole). In the final stage, the performance score (von Mises stress) and model volume are saved in an Excel file. This structural analysis process was applied for all 3D models generated by the GAN model. The data preparation and structural analysis process took, on average, 36.2 s per component. The evaluation time of all models was approximately 5 h.

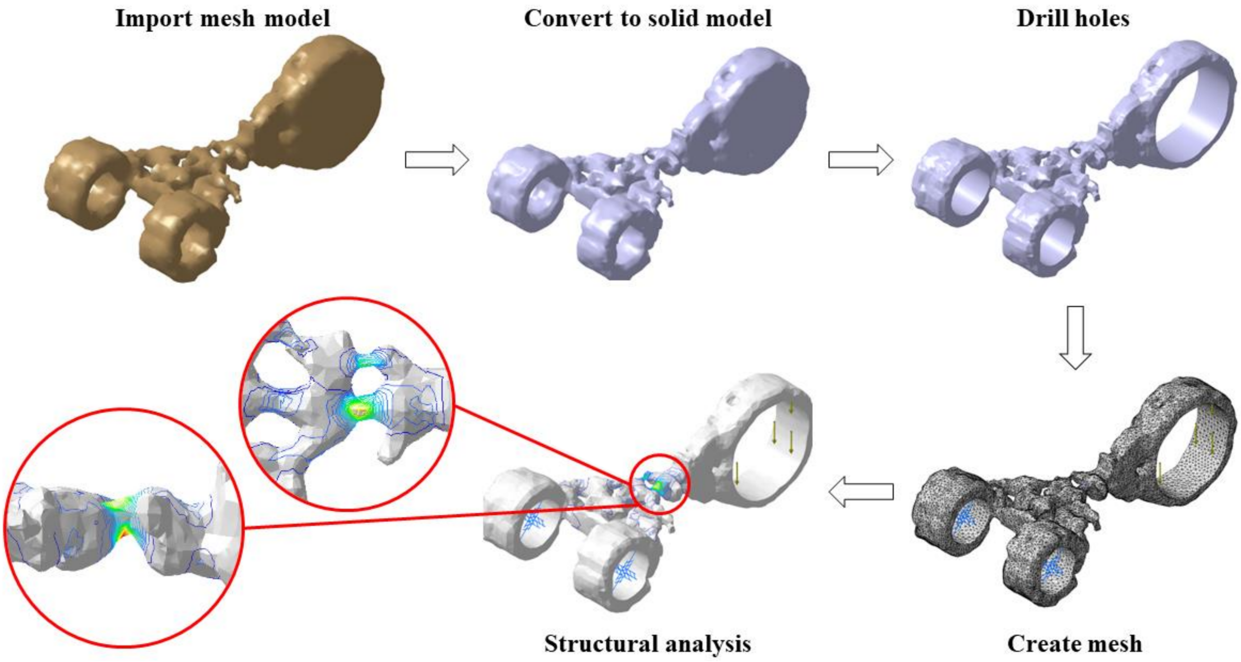
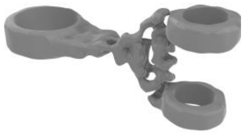
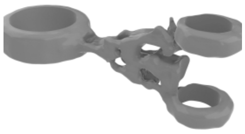



**Figure 6.** Preparation of 3D model and structural analysis process.

At each structural analysis stage, 500 3D models were evaluated, and 250 with relatively superior mechanical performance were included in the dataset. The amount of data was kept constant by randomly removing 250 3D models from the initial dataset. The GAN model was retrained using the training dataset created with the newly added data. At the end of each training step, the training data were updated by evaluating the mechanical performance of the 3D models produced. This iterative process was completed after six cycles. In the last cycle, approximately 20.3% of the training set consisted of data generated by the GAN model. Table 3 shows some of the GAN models generated by the 3DGAN model at the end of the sixth iteration. The physical and mechanical properties of these models are also in the table. The table shows that some high-volume 3D models were subjected to high stresses due to their thin sections.

**Table 3.** Examples of joints produced by the 3DGAN model at the end of the fourth iteration and their physical properties.

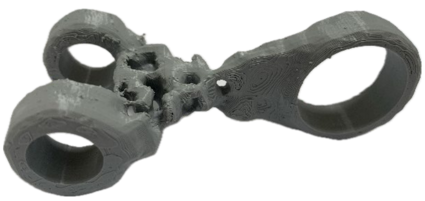
| **Generated Models** | | **Properties** | |
|---|---|---|---|
|  |  | Part | : P_98 |
| | | Material | : AlSi10Mg |
| | | Volume (cm$^3$) | : 103.23 |
| | | von Mises (MPa) | : 71.2 |
| | | Safety factor | : 3.79 |
|  |  | Part | : P_137 |
| | | Material | : AlSi10Mg |
| | | Volume (cm$^3$) | : 98.28 |
| | | von Mises (MPa) | : 61.8 |
| | | Safety factor | : 4.37 |
|  |  | Part | : P_358 |
| | | Material | : AlSi10Mg |
| | | Volume (cm$^3$) | : 95.36 |
| | | von Mises (MPa) | : 69.8 |
| | | Safety factor | : 3.86 |

### 4.4. Additive Manufacturing (AM)

Concept designs were ranked according to their weight and strength properties in the last step of the generation and evaluation cycle. According to this ranking, the design concept with the best performance was additively manufactured. In this process, the CAD model was converted to STL format and made ready for printing. The concept design was produced by adjusting the printer settings, such as layer height, infill density, and print speed, according to the desired precision and print quality.

The final iteration's concept design with the best mechanical properties was produced using two different additive manufacturing methods: SLA (stereolithography) and FDM (fused deposition modeling). Table 4 compares the production process of this part (model generated by GAN) using SLA and FDM techniques. While polylactic acid (PLA) offers an environmental advantage due to its biodegradability, the use of liquid resin in SLA provides superior surface quality. Moreover, SLA's finer layer thickness makes it more suitable for this application, which demands high detail and surface quality. In terms of production speed, SLA completes the process in 175 min, whereas FDM takes 317 min. However, FDM emerges as a more cost-effective option with a cost of USD 0.61 compared to SLA's USD 1.17.

**Table 4.** Production comparison of SLA and FDM methods.

| **Methods** | **Parts** | | |
|---|---|---|---|
| SLA |  | Material | : Resin |
| | | Quantity of Material | : 18 mL |
| | | Layer Thickness | : 0.05 mm |
| | | Production Time | : 175 min. |
| | | Cost | : USD 1.17 |
| FDM |  | Material | : PLA |
| | | Quantity of Material | : 22 mL |
| | | Layer Thickness | : 0.12 mm |
| | | Production Time | : 317 min. |
| | | Cost | : USD 0.61 |

## 5. Results and Discussion

GAN is an effective generative design tool for generating new data similar to the geometrical configurations of training data. The main shortcoming of GAN models in engineering design is that mechanical properties such as elastic modulus, yield strength, and energy absorption cannot be controlled in the generated data. Since the properties to be learned by the GAN model are not specified by the user, it is very difficult to learn these mechanical properties with existing GAN models. Instead, this study proposes an indirect approach to learning mechanical properties. Since GAN models learn from the dataset, building the dataset from only high-structural-performance workpieces ensures that the 3D models generated by the GAN model also exhibit high structural performance. Thus, mechanical features are included in the generative deep learning training process.

The proposed design and evaluation cycle aims to improve the mechanical properties of 3D models generated by GAN. In the first stage, a synthetic dataset is created using parametric design techniques to overcome the data shortage problem. The 3DGAN model is trained with this synthetic dataset in the second stage. In the last stage, 3D models generated by the 3DGAN model are analyzed in a simulation software to determine the mechanical properties. The 3D models with superior mechanical properties determined here are included in the dataset to be used in the next iteration. Random data are removed from the dataset equal to the amount of new data added. The proposed method analyzes only the effect of data quality on GAN performance without changing the amount of data and training parameters.

Strategically, we preferred to indirectly tune the training dataset rather than use a physics-informed GAN. This approach was primarily applied with the aim of exploring the potential of GANs to generate new designs that deviate significantly from standard models while maintaining performance. By editing the training dataset, an attempt was made to provide the GAN model with a variety of examples that enabled it to implicitly learn the desired performance characteristics without explicitly defining them. This method allows the GAN to learn and generate new designs that are not just variations of existing models but also potentially innovative solutions that may not be immediately noticeable with direct coding.

There are three reasons why the proposed method is not applied to the initial training dataset. Firstly, pre-filtering the limited training data may leave insufficient data for deep learning model training. Secondly, the reduction in data diversity may cause an over-learning problem. Data diversity is necessary for GANs to learn the common features of a design class. Finally, analyzing the whole dataset in a simulation environment is computationally inefficient.

The main objective of this study was not to maximize the performance of a structural component but to demonstrate the contribution of the presented method in improving mechanical properties. Although only one joint model was considered in this study, the proposed method can be applied to various structural components where mechanical properties are crucial, such as brackets, hinges, and load-bearing parts. Similarly, von Mises stress, which provides accurate predictions in multiaxial stress situations, was considered as the primary performance metric for evaluating joint models. However, many other critical material properties, such as elastic modulus, stiffness, fatigue strength, yield strength, and natural frequency, are worthy of consideration in engineering design and evaluation. Integration of other performance metrics into the proposed framework is necessary for a comprehensive evaluation.

Figure 7 shows the calculated safety factor values according to the von Mises stress obtained at each iteration step. In the graphs, the x-axis shows the volumes of the generated designs, while the y-axis represents the safety factor of the design. In the first iteration, the average safety factor ($\mu SF$) and its standard deviation ($\sigma SF$) were 0.947 and 0.645, respectively. No significant performance change was observed in the second and third iterations. However, a notable increase in the average safety factors of the designs was seen after the fourth and fifth iterations. Compared to the first iteration, there was a slight decrease of 0.01% in the safety factor at the end of the second iteration. In subsequent iterations, the average mechanical performance characteristics of the generated designs showed increases of 3.8%, 11.2%, 16.6%, and 20.3%, respectively. The findings revealed that there was a statistically significant improvement in the mechanical properties of 3D models generated with the revised dataset. In addition, the included 3D models, produced with GANs in the dataset, also affected data diversity. We found that 3D models with more organic forms increased the depth of the dataset and thus the volumetric diversity of the designs produced. On the other hand, when the safety factors were analyzed, there was a linear decrease in the standard deviation values. Figure 8 shows the average safety factors at each iteration. The circle diameters on the graph are sized according to the standard deviation values of the safety factors, expressing the data diversity. The first training iteration generated the highest data diversity, while this value was lower in the sixth iteration, as expected. Additionally, the ratios of data generated by GANs within the dataset used in each iteration are depicted on the graph. Figures 7 and 8 show the positive conditionability of the designs produced with GANs. Data diversity improves in the desired direction as the number of iterations increases.

When evaluating the initial dataset and the datasets developed through subsequent iterations in terms of geometric diversity, innovation, and quality, it was evident that the initial dataset assured both diversity and robustness. This assurance stemmed from the fact that the initial dataset was created using parametric design techniques. In repeated processes, while the design quality generally improved in the desired direction, there was a slight decrease in diversity. This suggested that repetitive processes can somewhat limit design diversity while maintaining certain quality standards. However, these processes also facilitated the emergence of more innovative and applicable designs for real-world applications. The potential of GAN models to produce designs that were dissimilar to the training dataset demonstrated their contribution to innovative design approaches and their scope for producing more original designs.

Another criterion taken into account in determining the performance of the generated 3D models was applicability, which refers to the success of the design in real-world applications. Although the GAN model could generate realistic data, in some cases, it generated incomplete or incorrect geometries. This situation revealed the potential of the GAN model to create designs that did not resemble the training dataset. On the other hand, the applicability of designs that did not fully reflect the dataset characteristics was weak. Some models could not be analyzed in the simulation environment due to fragile sections or discontinuities on the 3D models. The proportions of 3D models that could not be analyzed were 12.8%, 12.3%, 11.8%, 12.9%, 11.5%, and 13.1% for the six iterations, respectively. These statistical results showed that the reorganized data did not have a significant effect on applicability.

The mechanical properties of the best-performing workpiece generated in the final iteration were precisely validated using Ansys 2022 r2 software. The results from this assessment closely aligned with those obtained through FEA automation. Figure 9 illustrates the mechanical behavior of the component under loading conditions, with a focus on critical sections. Here, the highest stress occurs at the area farthest from the point of force application and closest to the support point, while other sections experience lower stress levels. The figure also provides information on the von Mises stresses and weights of the original, GAN-generated, and edited models. Initially, the original model has a relatively high weight (579.79 g) but its sharp corners lead to stress concentrations. The GAN-generated model, despite its lower weight (266.39 g), exhibits mechanical properties similar to the original model. This model was further enhanced by manually thickening the thin sections and removing volumes not subjected to stress, achieving better mechanical performance characteristics (43,618 MPa).

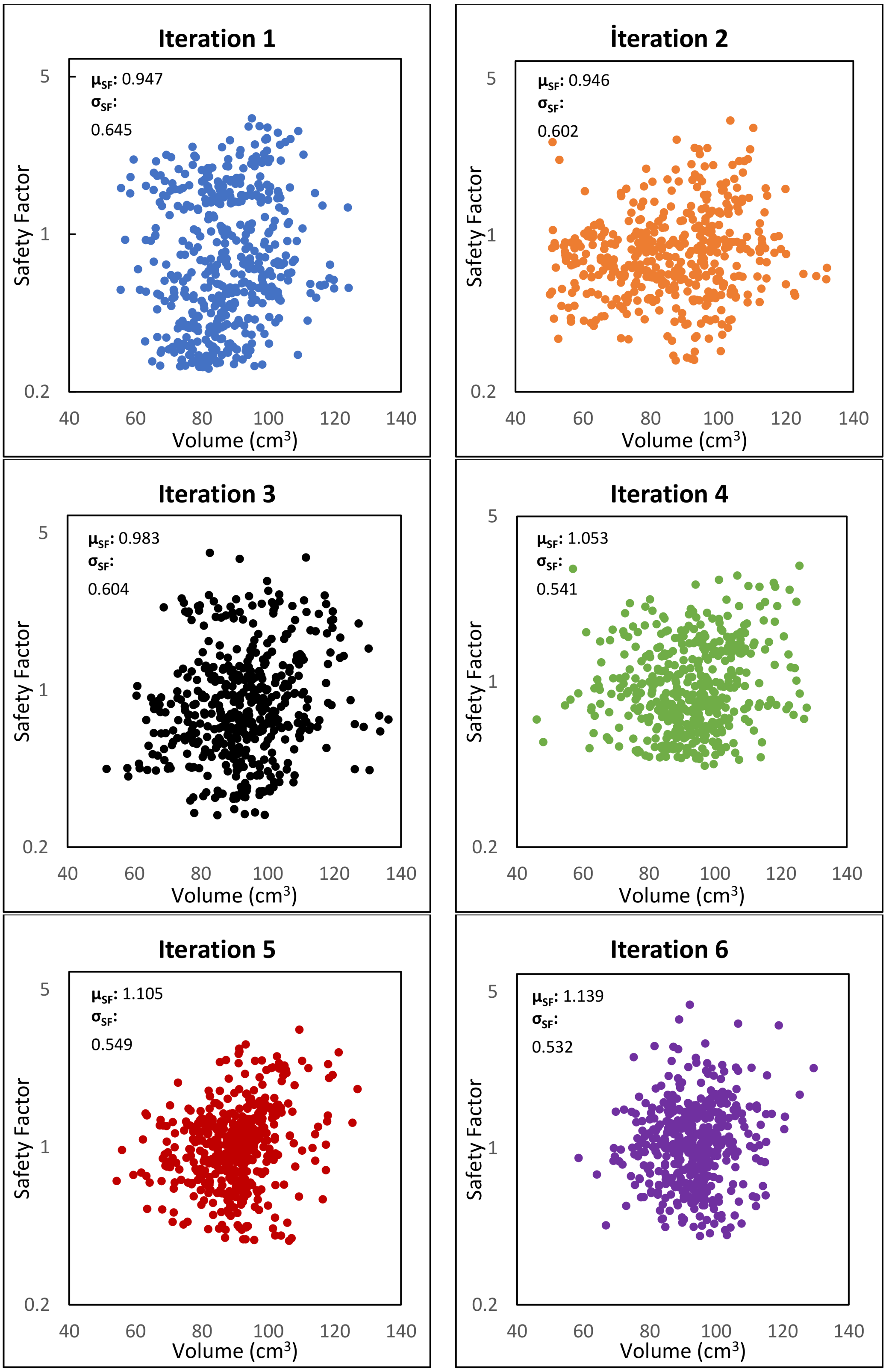



**Figure 7.** Volume and safety factors of the generated designs.

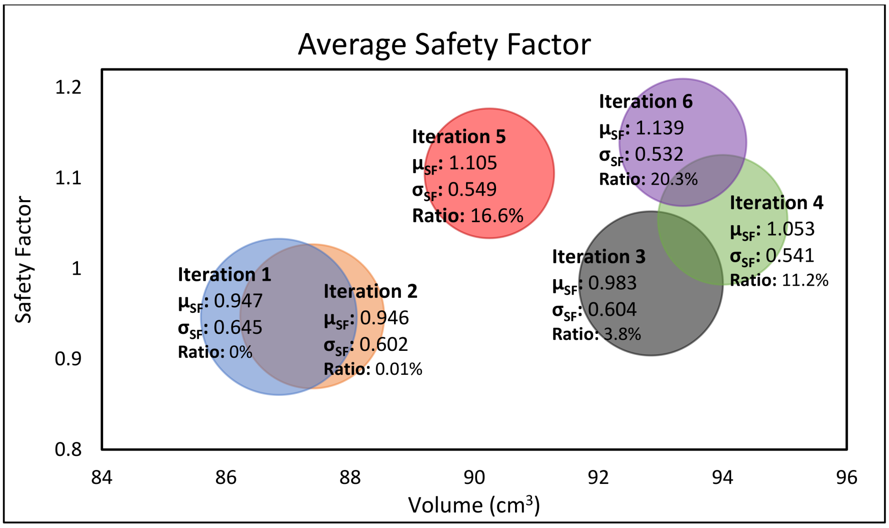



**Figure 8.** Average safety factors for training iterations.

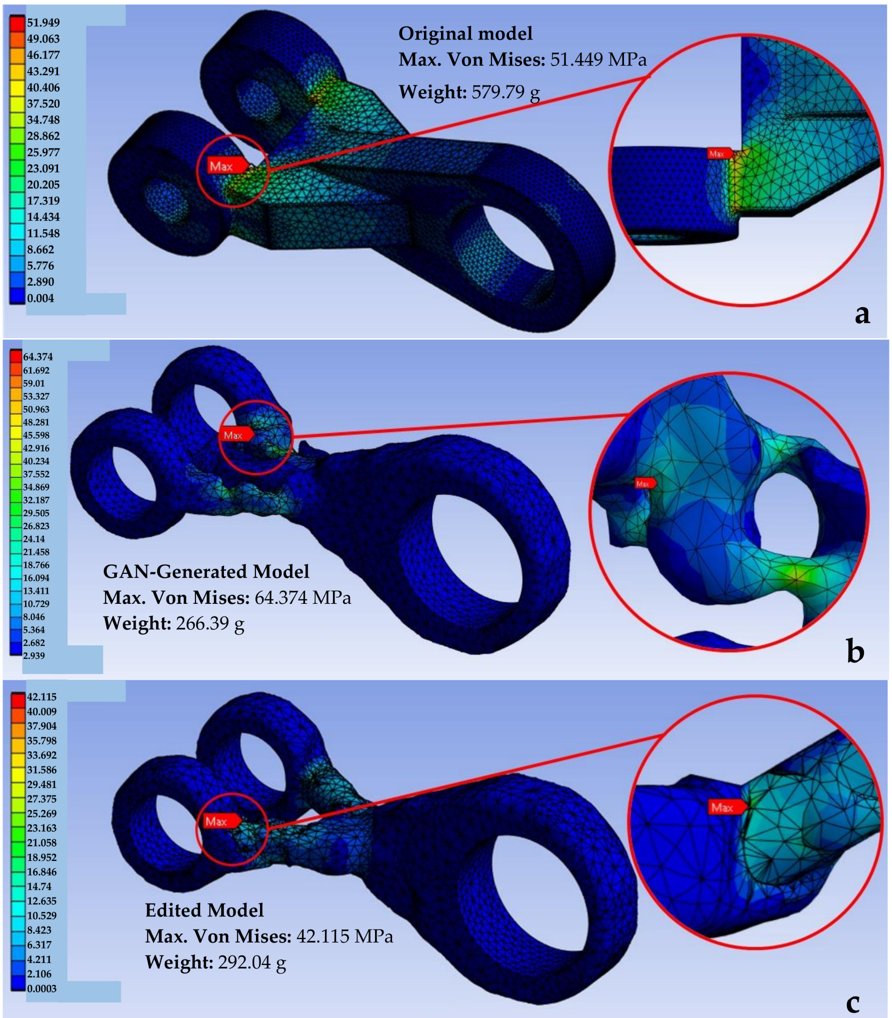



**Figure 9.** Verification of finite element analysis automation. (**a**) Original model, (**b**) GAN-generated model, (**c**) edited model.

The proposed design and evaluation framework enables the automation of the following steps carried out in the traditional design process: needs identification, idea and concept generation, modeling, evaluation, and redesign. Thus, product development processes, which are part of the product life cycle, can be performed automatically without human intervention. This minimizes human errors, shortens the design process, and saves on qualified labor. Furthermore, compared to traditional topology optimization, this method aims to generate a variety of feasible design options rather than a single ideal design option. With the network weights obtained at the end of training, designs with superior mechanical properties can be produced in seconds.

The inclusion of finite element analysis in the GAN training process enables the generation of designs with targeted mechanical performance. Moreover, the generative deep learning model, which is partially trained with a limited amount of data, can produce an unlimited number of functional 3D models. This generative design framework offers a potential solution to the problem of data shortage thanks to its capacity to generate new datasets. This feature offers a new way to create the necessary training data for deep learning studies, especially in the defense, aerospace, and automotive industries where the number of data points is insufficient and/or data are difficult to access due to confidentiality. Training 3DGANs directly with 3D model data makes them suitable for many different applications.

On the other hand, this approach presents certain potential challenges and limitations. One notable issue is that as the mechanical performance of the generated designs improves, there is a corresponding increase in the volume of these designs. In addition, it may be necessary to develop more complex GAN architectures because of the need for a high learning capacity in large datasets. This requires more computational power and time. At the same time, some targeted mechanical properties may be more difficult to learn. In addition, if factors such as real-world material properties, manufacturing methods, and conditions of use are ignored, the generated designs may not perform as expected in practical applications. Furthermore, this method cannot take into account traditional manufacturing constraints. This may limit the use and application of designs generated by GANs in real-world applications. For example, while a design may be suitable for 3D printing, it may not be suitable for casting or machining. Therefore, future work may consider incorporating manufacturing-process-specific constraints into the GAN model.

While the 3DGAN model enables the creation of organic and fine-geometry designs, these designs are difficult to manufacture with traditional manufacturing methods due to limitations in tooling and the need for support structures. The SLA method, which is prominent in additive manufacturing, was used in this study to investigate the manufacturing potential of designs obtained with 3DGAN and verified with FEA. SLA enables the production of complex geometries and fine details. The 3D prints obtained show the potential of the proposed method to be applied to real life. However, other methods, especially metal additive manufacturing methods, should be tested for more comprehensive results [54].

## 6. Conclusions

This study presents a new framework for enhancing the mechanical properties of 3D concepts generated by the 3DGAN model. The proposed framework consists of parametric design, deep learning training, and finite element analysis (FEA). The integration of 3DGAN and FEA enables the production of products with superior mechanical properties in an automated process. The proposed iterative design and evaluation framework aims to overcome the problems of lack of control of mechanical properties and data scarcity in engineering design. The framework provides an approach that attempts to optimize both design quality and design variety.

The results obtained show that 3DGANs have significant potential in engineering design. In particular, autonomous generation of components that fulfil certain mechanical properties can speed up design processes and make the product development process more efficient. Although only one synthetic dataset was used in this study, the proposed framework can be used in the design and development of many structural components used in engineering design. However, to reveal the full potential of this approach, further research may be required to overcome its current limitations and challenges. In particular, the adaptation and optimization of this approach for different engineering disciplines and application areas is an important research topic.

In future work, the application of optimization techniques to improve the stability and accuracy of the GAN model may help to minimize the potential errors. Including advanced GAN architectures in the proposed framework will increase the level of detail and applicability of the generated designs. Furthermore, the applicability of the proposed method to more complex engineering problems, such as multi-material components, functionally graded materials, and parts subjected to dynamic loading, should be investigated.

**Author Contributions:** Conceptualization, N.Y.; methodology, N.Y.; software, N.Y.; validation, N.Y. and H.R.B.; resources, N.Y.; data curation, N.Y.; writing—original draft preparation, N.Y. and H.R.B.; visualization, N.Y.; supervision, H.R.B.; project administration, H.R.B.; funding acquisition, H.R.B. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by TÜBİTAK, grant number 122E411.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The raw data supporting the conclusions of this article will be made available by the authors on request.

**Conflicts of Interest:** The authors declare no conflicts of interest.

## References

1. Panchal, J.H.; Fuge, M.; Liu, Y.; Missoum, S.; Tucker, C. Special Issue: Machine Learning for Engineering Design. *ASME J. Mech. Des.* **2019**, *141*, 110301. [CrossRef]
2. Gatys, L.A.; Ecker, A.S.; Bethge, M. Image Style Transfer Using Convolutional Neural Networks. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 2414–2423.
3. Lei, X.; Liu, C.; Du, Z.; Zhang, W.; Guo, X. Machine Learning-Driven Real-Time Topology Optimization under Moving Morphable Component-Based Framework. *J. Appl. Mech.* **2018**, *86*, 011004. [CrossRef]
4. Yüksel, N.; Börklü, H.R.; Sezer, H.K.; Canyurt, O.E. Review of artificial intelligence applications in engineering design perspective. *Eng. Appl. Artif. Intell.* **2023**, *118*, 105697. [CrossRef]
5. Salehi, H.; Burgueño, R. Emerging artificial intelligence methods in structural engineering. *Eng. Struct.* **2018**, *171*, 170–189. [CrossRef]
6. Shu, D.; Cunningham, J.; Stump, G.; Miller, S.W.; Yukish, M.A.; Simpson, T.W.; Tucker, C.S. 3D Design Using Generative Adversarial Networks and Physics-Based Validation. *J. Mech. Des.* **2020**, *142*, 071701. [CrossRef]
7. Verganti, R.; Vendraminelli, L.; Iansiti, M. Innovation and Design in the Age of Artificial Intelligence. *J. Prod. Innov. Manag.* **2020**, *37*, 212–227. [CrossRef]
8. Han, J.; Shi, F.; Chen, L.; Childs, P.R.N. A computational tool for creative idea generation based on analogical reasoning and ontology. *Artif. Intell. Eng. Des. Anal. Manuf.* **2018**, *32*, 462–477. [CrossRef]
9. Boden, M.A. *The Creative Mind: Myths and Mechanisms*; Routledge: London, UK, 2004.
10. Yuan, C.; Moghaddam, M. Attribute-Aware Generative Design With Generative Adversarial Networks. *IEEE Access* **2020**, *8*, 190710–190721. [CrossRef]
11. Zhao, T.; Yang, J.; Zhang, H.; Siu, K.W.M. Creative idea generation method based on deep learning technology. *Int. J. Technol. Des. Educ.* **2021**, *31*, 421–440. [CrossRef]
12. Abbas, A.; Rafiee, A.; Haase, M. DeepMorpher: Deep learning-based design space dimensionality reduction for shape optimisation. *J. Eng. Des.* **2023**, *34*, 254–270. [CrossRef]
13. Regenwetter, L.; Nobari, A.H.; Ahmed, F. Deep generative models in engineering design: A review. *J. Mech. Des.* **2022**, *144*, 071704. [CrossRef]
14. Zhang, W.; Yang, Z.; Jiang, H.; Nigam, S.; Yamakawa, S.; Furuhata, T.; Shimada, K.; Kara, L.B. 3D Shape Synthesis for Conceptual Design and Optimization Using Variational Autoencoders. In Proceedings of the ASME 2019 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Anaheim, CA, USA, 18–21 August 2019.
15. Raina, A.; McComb, C.; Cagan, J. Learning to Design From Humans: Imitating Human Designers Through Deep Learning. *J. Mech. Des.* **2019**, *141*, 111102. [CrossRef]
16. Li, M.; McComb, C. Using Physics-Informed Generative Adversarial Networks to Perform Super-Resolution for Multiphase Fluid Simulations. *J. Comput. Inf. Sci. Eng.* **2022**, *22*, 044501. [CrossRef]

17. Yu, S.; Dong, H.; Wang, P.; Wu, C.; Guo, Y. *Generative Creativity: Adversarial Learning for Bionic Design*; Springer: Cham, Switzerland, 2019; pp. 525–536.
18. Mirza, M.; Osindero, S. Conditional generative adversarial nets. *arXiv* **2014**, arXiv:1411.1784.
19. Chen, W.; Lee, D.; Balogun, O.; Chen, W. GAN-DUF: Hierarchical Deep Generative Models for Design under Free-Form Geometric Uncertainty. *J. Mech. Des.* **2022**, *145*, 011703. [CrossRef]
20. Oh, S.; Jung, Y.; Kim, S.; Lee, I.; Kang, N. Deep Generative Design: Integration of Topology Optimization and Generative Models. *J. Mech. Des.* **2019**, *141*, 1151. [CrossRef]
21. Yonekura, K.; Suzuki, K. Data-driven design exploration method using conditional variational autoencoder for airfoil design. *Struct. Multidiscip. Optim.* **2021**, *64*, 613–624. [CrossRef]
22. Nie, Z.; Lin, T.; Jiang, H.; Kara, L.B. TopologyGAN: Topology Optimization Using Generative Adversarial Networks Based on Physical Fields Over the Initial Domain. *arXiv* **2020**, arXiv:2003.04685.
23. Yin, B.; Chen, Z.; Tao, M. Predictive GAN-Powered Multi-Objective Optimization for Hybrid Federated Split Learning. *IEEE Trans. Commun.* **2023**, *71*, 4544–4560. [CrossRef]
24. Yoo, S.; Lee, S.; Kim, S.; Hwang, K.H.; Park, J.H.; Kang, N. Integrating deep learning into CAD/CAE system: Generative design and evaluation of 3D conceptual wheel. *Struct. Multidiscip. Optim.* **2021**, *64*, 2725–2747. [CrossRef]
25. Chen, W.; Fuge, M. Synthesizing Designs With Interpart Dependencies Using Hierarchical Generative Adversarial Networks. *J. Mech. Des.* **2019**, *141*, 111403. [CrossRef]
26. Lee, H.; Lee, J.; Kwon, S.; Ramani, K.; Chi, H.-G.; Mun, D. Simplification of 3D CAD Model in Voxel Form for Mechanical Parts Using Generative Adversarial Networks. *Comput.-Aided Des.* **2023**, *163*, 103577. [CrossRef]
27. Chen, W.; Ahmed, F. MO-PaDGAN: Reparameterizing Engineering Designs for augmented multi-objective optimization. *Appl. Soft Comput.* **2021**, *113*, 107909. [CrossRef]
28. Nobari, A.H.; Chen, W.; Ahmed, F. PcDGAN: A Continuous Conditional Diverse Generative Adversarial Network For Inverse Design. In Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining, Virtual Event, Singapore, 14–18 August 2021; Association for Computing Machinery: Singapore, 2021; pp. 606–616.
29. Nobari, A.H.; Chen, W.; Ahmed, F. Range-Constrained Generative Adversarial Network: Design Synthesis Under Constraints Using Conditional Generative Adversarial Networks. *J. Mech. Des.* **2021**, *144*. [CrossRef]
30. Giannone, G.; Regenwetter, L.; Srivastava, A.; Gutfreund, D.; Ahmed, F. Learning from Invalid Data: On Constraint Satisfaction in Generative Models. *arXiv* **2023**, arXiv:2306.15166.
31. Yu, Y.; Hur, T.; Jung, J.; Jang, I.G. Deep learning for determining a near-optimal topological design without any iteration. *Struct. Multidiscip. Optim.* **2019**, *59*, 787–799. [CrossRef]
32. Khan, S.; Goucher-Lambert, K.; Kostas, K.; Kaklis, P. ShipHullGAN: A generic parametric modeller for ship hull design using deep convolutional generative model. *Comput. Methods Appl. Mech. Eng.* **2023**, *411*, 116051. [CrossRef]
33. Behzadi, M.M.; Ilieş, H.T. Real-Time Topology Optimization in 3D via Deep Transfer Learning. *Comput.-Aided Des.* **2021**, *135*, 103014. [CrossRef]
34. Rawat, S.; Shen, M.H. *Application of Adversarial Networks for 3D Structural Topology Optimization*; SAE International: Warrendale, PA, USA, 2019.
35. Wu, J.; Xing, B.; Si, H.; Jian, D.; Wang, J.; Zhu, Y.; Liu, X. Product Design Award Prediction Modeling: Design Visual Aesthetic Quality Assessment via DCNNs. *IEEE Access* **2020**, *8*, 211028–211047. [CrossRef]
36. Achlioptas, P.; Diamanti, O.; Mitliagkas, I.; Guibas, L.J. Learning Representations and Generative Models for 3D Point Clouds. In Proceedings of the International Conference on Machine Learning, Kyoto, Japan, 2017; Available online: https://arxiv.org/abs/1707.02392 (accessed on 17 January 2024).
37. Li, C.-L.; Zaheer, M.; Zhang, Y.; Poczos, B.; Salakhutdinov, R. Point cloud gan. *arXiv* **2018**, arXiv:1810.05795.
38. Singh, R.D.; Mittal, A.; Bhatia, R.K. 3D convolutional neural network for object recognition: A review. *Multimed. Tools Appl.* **2019**, *78*, 15951–15995. [CrossRef]
39. Wu, Z.; Song, S.; Khosla, A.; Yu, F.; Zhang, L.; Tang, X.; Xiao, J. 3d shapenets: A deep representation for volumetric shapes. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015.
40. Caetano, I.; Santos, L.; Leitão, A. Computational design in architecture: Defining parametric, generative, and algorithmic design. *Front. Archit. Res.* **2020**, *9*, 287–300. [CrossRef]
41. Ahlquist, S.; Menges, A. *Computational Design Thinking*; Wiley: London, UK, 2011.
42. Drezner, T.; Drezner, Z. Voronoi diagrams with overlapping regions. *OR Spectr.* **2013**, *35*, 543–561. [CrossRef]
43. Laaroussi, F.-E.; Nouh, B.; Bakkali, M. Parametric Building Design Options Within Voronoï Diagram Based Urban Fabric. In *Intelligent Environments 2020*; IOS Press: Amsterdam, The Netherlands, 2020; pp. 278–288.
44. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial networks. *Commun. ACM* **2020**, *63*, 139–144. [CrossRef]
45. Wu, J.; Zhang, C.; Xue, T.; Freeman, W.T.; Tenenbaum, J.B. Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling. In Proceedings of the 30th International Conference on Neural Information Processing Systems, Barcelona, Spain, 5–10 December 2016; pp. 82–90.
46. Kastner, P.; Dogan, T. A GAN-Based Surrogate Model for Instantaneous Urban Wind Flow Prediction. *Build. Environ.* **2023**, *242*, 110384. [CrossRef]

47. Rahman, R. Using Generative Adversarial Networks for Content Generation in Games. In *Computer Games Technology*; Abertay University: Dundee, UK, 2020.
48. Bendale, S. A Study of 3D-GANs and their Implementation Challenges. *Int. Res. J. Eng. Technol.* **2020**, *194*.
49. Yi, X.; Walia, E.; Babyn, P. Generative adversarial network in medical imaging: A review. *Med. Image Anal.* **2019**, *58*, 101552. [CrossRef] [PubMed]
50. Yi, G.; Sui, Y.; Du, J. Application of python-based Abaqus preprocess and postprocess technique in analysis of gearbox vibration and noise reduction. *Front. Mech. Eng.* **2011**, *6*, 229–234. [CrossRef]
51. Bikas, H.; Stavropoulos, P.; Chryssolouris, G. Additive manufacturing methods and modelling approaches: A critical review. *Int. J. Adv. Manuf. Technol.* **2016**, *83*, 389–405. [CrossRef]
52. Ian Gibson, I.G. *Additive Manufacturing Technologies 3D Printing, Rapid Prototyping, and Direct Digital Manufacturing*; Springer: Berlin/Heidelberg, Germany, 2015.
53. Rahim, T.N.A.T.; Abdullah, A.M.; Md Akil, H. Recent developments in fused deposition modeling-based 3D printing of polymers and their composites. *Polym. Rev.* **2019**, *59*, 589–624. [CrossRef]
54. Frazier, W.E. Metal additive manufacturing: A review. *J. Mater. Eng. Perform.* **2014**, *23*, 1917–1928. [CrossRef]