Deep generative models for engineering design

Fan, J.

Citation

Fan, J. (2026, March 24). *Deep generative models for engineering design*. Retrieved from https://hdl.handle.net/1887/4298630

Chapter 1

Introduction

Human design is a ubiquitous element of the modern world and plays a crucial role in the technologies involved in creating the products we use. However, product design processes can be extremely time-consuming due to the increasing complexity of the products, e.g., the development process for one car product takes five years. Time invested in a product may thus become critical in light of rapidly evolving market needs and preferences for more demanding and customized products. Automating and accelerating the design process would yield significant cost reductions and increase productivity within industries, an immeasurable gain for global productivity and prosperity. In Section 1.1, we introduce a traditional industrial design process in detail, as well as the challenges the industry faces therein.

Generative engineering design (GED) [3, 34, 115, 138] is proposed to overcome this issue. GED represents a new trend toward generative design originating from a more function-oriented design exploration, utilizing existing designs and algorithms for assisting engineers in developing complex structures. Most of the early-stage GED methods involve the manual definition of design rules based on human experience, i.e., explicit programming of design constraints and objectives. Developing such rule-based algorithms is very time-consuming and exceedingly costly. However, comparatively, the implicit learning of existing designs holds a lot of promise in generating powerful design exploration algorithms at a plurality of times lower costs than possible. Moreover, manually-programmed GED algorithms are essentially always project-dependent and involve a significant number of predefined parameters. However, strong algorithms intended for modern industries should be highly flexible in various design requirements and scenarios. Some more insight into the impact GED will have on modern industries is provided in Section 1.2.

In recent years, machine learning has exhibited tremendous potential in surpassing the constraints of conventional algorithms across various real-world applications, including text understanding [174], image classification [81], object detection [136] and game playing [107]. Noteworthy achievements are evident in the field of deep generative models (DGMs, known as GenAI), like large language models [40, 20, 1], image synthesis [49, 72] and generation of other data formats [117, 108, 169, 130]. Since the emergence of ChatGPT [118], there has been a keen interest in deep generative models (and even large generative models) and their applications in industry. Further insights into DGMs are discussed in Section 1.3.

With the rapid development of DGM, utilizing it for generative engineering design becomes an intuitive decision due to its impressive ability to create new data. Regenwetter et al. have conducted a comprehensive survey [138], delving into this area and revealing the potential of DGMs in facilitating industrial design processes. In Section 1.4, we also demonstrate the most recent applications of DGMs for engineering designs. Despite the impressive results achieved in modern DGM researches, there is still a significant gap between current DGM technology and industrial applications. Below, we list the primary challenges faced by generative engineering using DGMs:

1. Poor plausibility of the designs generated. For the purpose of assisting the industrial design process, the generated designs must observe certain semantic constraints (referred to as plausibility, e.g., no floating material or missing parts). While the samples generated by current DGMs show outstanding visual quality (referred to as fidelity, e.g., no background noise and clear details), they often fail in generating plausible results. This directly results in these DGM-driven solutions being inefficient and unreliable.

2. Lack of metrics for measuring plausibility. Model evaluation is essential in the development of DGMs, as it allows for the ranking of models during the development phase and assesses the efficiency and reliability of the final model. When using DGM to enhance engineering design processes, evaluating the model's performance in terms of plausibility becomes critical. However, unlike metrics for fidelity or diversity, plausibility is an inherently subjective criterion, making it challenging to measure automatically. Current evaluation metrics often neglect the model's performance regarding plausibility. Relying solely on these existing metrics can lead to misguided decisions, such as selecting a DGM that ultimately fails to generate plausible designs.

3. Computation inefficiency for 3D objects. Advances in DGMs have made them proficient in generating regular data formats, such as images. However, 3D shapes (such as meshes and B-Reps) are not so straightforward for recent DGMs due to various reasons. These processes are sometimes energy and computation intensive, translating to prohibitive costs. Again, 3D object generation often relies on large-scale datasets; hence, performance obtained from modestly sized datasets hinders practical applications.

4. Inability to directly generate CAD-native representations of geometric objects. In general, DGMs used for generating engineering designs typically rely on cross-sections or meshes to represent source designs. Although widely used, such representations do not correspond to the most natural form of CAD data, which is primarily based on B-Rep solids described using parametric geometries. This discrepancy presents a significant limitation, as the conversion from images or meshes to B-Reps remains challenging at present. Learning DGMs directly from B-Reps is therefore a more logical and effective approach, as it allows for the generation of designs that are inherently compatible with CAD systems. By focusing on B-Reps representations, DGMs can produce more accurate and usable geometric models, facilitating smoother integration into engineering workflows.

5. Lack of historical successful application in industrial cases. The successes of DGMs are often showcased in controlled environments where the data is well-curated, characterized by a rich data pool and high quality. However, industrial applications frequently encounter a range of challenges that hinder the effective deployment of DGMs. These challenges include limited sample sizes, a significant amount of invalid or noisy data, and data formats that are not directly learnable by current models. As a result, the performance of state-of-the-art DGMs in these industrial contexts may not match their effectiveness in more general tasks. This gap highlights the need for further research and development to adapt DGMs for real-world industrial applications, ensuring they can handle the complexity and variability of engineering data. Addressing these issues is crucial for unlocking the full potential of DGMs in driving innovation and efficiency in industrial design processes.

While the above-mentioned limitations prevent engineering processes from taking advantages of the benefits of DGM, they indicate challenges with great industrial interest and research value. This thesis summarizes these challenges into corresponding research topics and solves them one by one by providing general solutions, which will not only benefit engineering design processes, but also contribute to the DGM research community. The research questions are as follows:

RQ1: How to prioritize plausibility in generation with DGMs? Prioritizing plausibility ensures that the designs generated by DGMs are practical and can be realistically implemented in real-world scenarios. This is crucial because generating implausible designs can lead to wasted resources and inefficiencies. By focusing on plausibility, engineers and designers can rely on DGMs to produce viable and innovative solutions, accelerating the design process and reducing trial-and-error phases. Therefore, we improve the noise scheduling of the stat-of-the-art diffusion model (EDM [71]) and propose plausibility-oriented diffusion model (PoDM) [167], details can be fond in Chapter 2.

RQ2: How to automatically evaluate the plausibility of designs generated by DGMs? Metrics are an important assistant in evaluating and ranking models during the development of DGM, but they also reveal an unsolved challenge: due to the lack of ground truth in generation, it is difficult to automatically evaluate the synthetic samples while aligning with human judgments. After reviewing existing research on developing DGMs, we note that plausibility is overlooked not only in DGM-driven generation (as questioned in **RQ1**) but also in model evaluation. To address this, we propose a novel metric Fréchet Denoised Distance (FDD) [164], detailed in Chapter 3, which complements other metrics and aligns with human designers in evaluating result plausibility.

RQ3: How to use DGMs to generate high-dimension designs more efficiently? High-dimension designs often involve complex structures and multiple variables, so training DGMs is time-consuming and challenging. DGM developers tend to generate high-dimensional data via training an autoencoder, while the DGM is trained in the learned latent space. This can be computational costly and challenging in engineering process, where sample number is limited and structure of each sample is complicated and in high dimension. In Chapter 4, we propose a decomposition-based generation framework, SpoDify [168], for high-dimensional data, where our method enables a learning-free encoding and the decomposition outcomes can be used for DGM training.

RQ4: How to enable DGMs to directly synthesize CAD native representation? Directly synthesizing CAD native representations allows for seamless integration of generated designs into the existing workflow and tools used by professionals.

This capability will reduce the need for manual conversion and adjustments, thereby enhancing productivity and ensuring that the generated designs can be immediately utilized in practical applications. We delve into this in Chapter 5 and propose NeuroNURBS [166] that is able to learn and generate directly B-Rep (the CAD native representations) and NURBS (the native representation of parametric geometries in B-Reps).

RQ5: How can DGMs be beneficial for real-world industrial design applications? Understanding the practical benefits of DGMs in real-world applications is essential for their adoption and integration into industry practices. This question addresses the tangible impact of DGMs on the field of industrial design. Demonstrating the benefits of DGMs—such as improved design efficiency, cost reduction, and enhanced innovation—will encourage industry adoption, leading to widespread improvements in design practices and outcomes. In Chapter 6, we take car rims and A-pillars as design targets and demonstrate the efficiency of DGMs in industrial cases.

Together all the above research questions are aiming to solve one primary research topic, which is:

How to enable DGMs to synthesize engineering designs?

1.1 Industrial Design Process

In modern industrial design processes, a design consists of a digital or virtual 3D object that defines the product of interest. For instance, a car or a subset of a car (such as a breaking system) is assembled from many connected components. The field of computer-aided design (CAD) encompasses all topics related to such virtual models. By nature, this virtual model or "CAD model" contains geometrical and topological information about the object. To some extent, material information can also be defined in the CAD model.

In industry, engineers conduct design, simulation, and manufacturing using CAD software such as `CATIA` of `Dassault Systems`, where the standard is to represent a solid model with boundary representation (B-Rep). In a B-Rep, the solid boundaries are defined using a set of surfaces [177, 39], which are, by default, parameterized by non-uniform rational B-splines (NURBS) [128]. A NURBS surface leverages weights assigned to the control points and non-uniform knot vectors to represent more complex shapes compared to Bézier or B-splines. It is worth noting that NURBS is often used as an intermediate data format when exchanging data between CAD software. NURBS allows to define shapes with perfect precision due to their continuity.

Often, CAD software allows the generation of alternative representations of the CAD model, such as images (cross-section or snapshot) or meshes, which can then be easily used outside of the CAD framework. However, converting these design representations back into CAD models can be challenging. In sum, industrial CAD models use relatively complex data representations with the following characteristics:

- A mixture of parametric data (material properties, geometric parameters such as thickness or radius, 3D geometry described using NURBS) and non-parametric data (3D geometry described using a point cloud or image).
- High-dimensionality. The virtual car model is full-scale and consists of numerous (up to hundreds) connected parts.
- Due to the lack of fixed design conventions (different sizes, various parameters and constituent objects), there is a lack of unique definitions when considering variants of the same design.

Within the framework set by predefined requirements, the designer can modify the geometry by adjusting geometric parameters or reshaping certain components. The designer may also modify the topology of the model by adding holes to the design. These design modifications are triggered by the designer's creativity and feedback from the engineers, who collaborate closely during the design process.

The product design process [60] begins with creating a preliminary CAD model, which evolves as designers incorporate creative ideas and perform functional evaluations. As design progresses, details are added and changes are made based on feedback from engineers to meet predefined requirements. These requirements, which range from quality and safety to manufacturability, are essential to ensuring compliance with legal and industry standards [51]. Traditionally, functional performance was gauged by experts and sometimes involved physical experiments. However, digital methods like numerical simulations have become more prevalent, offering cost-effective and faster evaluations in a partially automated way. This is the essence of computer-aided engineering (CAE), which employs various numerical techniques based on the physical aspect being assessed. Finite-element methods (FEM) are used for structural problems, while finite-volume simulations deal with fluid mechanics. For simulations, CAD models must be transformed into CAE-compatible models, involving meshing (discretizing the continuous geometry) to handle complex geometries—a task requiring expertise. Although once set up, a simulation can run autonomously, substantial expert input is necessary for initial data preparation. Additionally, simulations demand expensive software licenses, considerable computing power, and time, sometimes taking minutes to days depending on model complexity. As such, simulation-based design evaluations remain resource-intensive. In early design stages, experts often rely on their judgment for functional assessments to conserve time and resources, introducing certain bias and communication challenges. To mitigate costs and inefficiencies, early identification of underperforming design variants is key, so that numerical simulations are reserved for promising designs only. This would enhance objectivity in functional assessments and reduce the experts' reliance on them when simulations are overly costly.

1.2 Generative Engineering Design

In the traditional engineering design process, designs need to be modified repeatedly to meet predefined requirements. In this design cycle, engineers are tasked with manually adjusting the design based on their expertise due to the lack of intelligent tools, making it a trial-error process. This involves constructing and simulating designs until achieving an optimal outcome, often without knowing how to improve the design exactly to reach the desired performance without adversely affecting other aspects. As a result, the engineering design process tends to be very time-consuming and expensive, indirectly contributing to the slow development of modern products and the high cost of new designs. Consequently, this approach is often very time-consuming and costly, which contributes to the slow pace of modern product development and the high expenses associated with new designs.

Recently, advanced algorithms either employ explicit programming or implicit learning to explore design possibilities that meet engineer predefined requirements [138]. This concept has been introduced as the term "generative design" [77] and yielded a tons of form-finding tools, which shows a great potential in complementing the previous "form-making" engineering process [153]. Generative design is proposed to support designers/engineers in developing complex structures, in which algorithms explore possible design alternatives based on input constrains. In practice, designers or engineers pre-determine several technical constraints and then develop algorithms to generate or optimize the design to meet these requirements. Generative design here is further referred as generative engineering design (GED) and is a subsequent step to parametric modeling in product development [153]. More specifically, GED helps develop designs that better or best meet the predefined requirements, whereas the current product development relies heavily on the construct-evaluate cycle of human designers [34].

Early computational tools that enable GED tend to be manually-defined pipeline. For instance, parametric modeling of a structural component allows designers to focus on shape optimization with limited parameters. These tools help with developing novel solutions that bypass fixation of human designers, and automate some routine design processes to improve efficiency [25]. However, these rule-based frameworks rely heavily on human design expertise and require significant cost in terms of financial investment and time in the development process. Later, with the rapid development of artificial intelligence (AI), the advantages of implicit leaning on information and knowledge encoded in the vast expanse of existing designs has been seen. This part has been introduced in detail in Section 1.4.

1.3 Deep Generative Modeling

Machine learning is a field of artificial intelligence that enables computers to learn and improve from experience. It involves the development of algorithms and statistical models that allow systems to perform specific tasks effectively without using rule-based programming. Prior to the rise of deep learning, a variety of applications have introduced machine learning techniques, including linear regression [54], decision trees [132], support vector machines (SVMs) [37], etc. These traditional machine learning techniques were widely used before the advent of deep learning, which has since become the dominant approach in many areas of artificial intelligence and data analysis.

Deep learning is capable of learning meaningful distributions from extensive quantities of data through a deep neural network with millions of trainable parameters. Starting with the compelling results achieved using autoencoder [59] to reduce the dimensionality of large-scale data with neural networks, the era of deep learning begins. With the introduction of convolutional layers, deep convolutional neural networks (DCNNs) can better capture patterns from the locality of pixel dependencies in images compared to fully-connected neural networks and hereby excel in the vision understanding tasks, e.g., the ImageNet classification task [38, 55, 81]. Despite the powerful learning capacity of deep neural networks, it is not trivial to train such deep architectures with a surge of parameters [47]. More precisely, training deep neural network suffers from long training time, early saturation, overfitting. To improve the training stability, a set of approaches are proposed, e.g., rectified linear units (ReLU) [81], dropout [160], residual block [55], batch normalization [65], etc.

Meanwhile, a new subfield that is making astonishing breakthroughs is the deep generative models (DGMs) — deep neural networks (DNNs) that can learn the complex probability distribution of large datasets and generate new samples. Unlike previous generative models, deep Boltzmann machines [142], generative stochastic networks [12] and variational autoencoders (VAE) [75], generative adversarial networks (GANs) [49, 133] have first yielded convincing results in image generation. Within a short period of time after that, a series of derivatives have been proposed to further improve the performance of the vanilla GAN, such as Wasserstein GAN with gradient penalty (WGAN-GP) [6] to stabilize the training, progressively growing GAN (ProGAN) [70] for producing images of large resolutions, conditional GAN (CGAN) [104] to enable conditional generation and StyleGAN [72] that allows for intuitive, scale-specific control of the synthesis. Note that SyleGAN is the first work observes that the level of the affected feature depends on the scale of the noise input. For example, tiny noises can edit the curves of the hair, while coarse noises can change gender. Our work in Chapter 2 is greatly inspired by their observation.

Diffusion models [155] present a novel idea for capturing the data distribution and generation. Diffusion models did not attract much attention until the convincing implementation of the denoising diffusion probabilistic model (DDPM) [61], which leverages tremendous sampling time to generate images with quality comparable to GANs. A further developed diffusion model, denoising diffusion implicit model (DDIM) [156], improves generation speed by trading off image quality. Meanwhile, in the domain of score-based generation, Song et al. [159] propose to use a stochastic differential equation (SDE) for the forward process and a corresponding reverse-time SDE for sampling, which allows continuous diffusion processes. SDE derives a deterministic sampling process based on a corresponding ordinary differential equation (ODE), that enables identifiable encoding-decoding and more importantly flexible data manipulation via latent space. Then, Karras et al. [71] clean the design space of diffusion-based generative models and propose a novel framework, denoted as EDM. EDM achieves a new state-of-the-art performance on the generation of CIFAR-10 [80] and ImageNet-64 [38]. Diffusion-based generative models have been introduced in controlling generation and data manipulation, e.g., interpolation via latent space [61, 159, 156, 41], free-form inpainting [159, 97] and point-based dragging [150]. These image editing methods tend to be applied on natural images, e.g., CelebA [94], LSUN bedroom images [192] and ImageNet [38].

A novel backbone, Transformer (consists solely of attention layers), has achieved compelling results in text translation task [174], which changes the dominant situation of convolutional and recurrent neural networks. Inspired by this, researchers start introducing attention layer in DGMs for computer vision tasks, e.g., self-attention GAN [186], which demonstrates superior performance than its contrastive version (i.e., GANs without attention layer). As an explanation to this, self-attention mechanism is able to capture long-range dependencies across image regions, whereas convolutional layers are restricted due to their limited receptive fields. Inspired by this, we use Self-Attention Adversarial Latent Autoencoder (SA-ALAE) [165] developed by ourself to generate cross-section images of car A-pillars, more details can be found in Chapter 6. Soon after this, introducing attention layers into the DGMs becomes a new standard. Most recently, Vision Transformer (ViT) [78] that applies a standard Transformer model directly to images, outperforms CNNs in image recognition tasks and exceeds the state of the art on many image classification datasets. This breaks the dominance of convolutional architectures and also the boundaries between the field of neuro-linguistic processing (NLP) and computer vision. While standard diffusion models utilize a U-Net architecture [61] to perform the denoising for each step, right after ViT showing its excellent performance in computer vision, transformer-based diffusion models (DiTs) [125] are invented.

While diffusion models have efficiently addressed the unstable training issues associated with GANs, they face a significant drawback: slow sampling speed. This limitation becomes particularly pronounced when working with high-resolution images, as both the model size and evaluation times increase exponentially. The latent diffusion model (LDM) introduced by Rombach et al. [139] mitigates this challenge by conducting diffusion and denoising processes in a lower-dimensional latent space, which is encoded from the source data using a pretrained autoencoder. Additionally, LDMs leverage cross-attention layers, making them versatile generators capable of handling various conditioning inputs, including text, semantic maps (masks for in-painting), and images. Another approach to improve the sampling speed of diffusion models involves distilling a pretrained diffusion model into a new model that requires significantly fewer steps for sampling, all while maintaining sample quality. Recent advancements in this area include progressive distillation [144], consistency models [157], adversarial diffusion distillation [145], distribution matching distillation [191].

1.4 Deep Generative Models for Engineering

Deep generative models (DGMs) have been successfully employed to synthesize general images, e.g., animals, human faces, and landscapes. This promising advancement leads to the idea of utilizing DGMs to generate novel structural designs, thereby facilitating industrial engineering processes. However, industrial design data, e.g., blueprints or engineering drawings, is fundamentally different from the images of natural scenes. They contain rich structural patterns and long-range dependencies, which are challenging for convolution-based DGMs to generate. Recently, some models (PaDGAN [27] and BézierGAN [28]) are able to generate UIUC Airfoil shapes [170], however, when it comes to more complicated design like BIKED bicycle designs [137], DGMs fail to synthesize feasible results [137]. In order to tackle this issue, Fan et al. [165] propose a novel model self-attention adversarial latent autoencoder (SA-ALAE), which allows generating feasible design images of complex engineering parts, but still suffers from unstable training due to the implementation of adversarial training.

Moreover, learning from 2D geometric designs is not sufficient for the rapidly evolving industry. Huge progress has been made in using modern neural networks to tackle 3D learning tasks. It starts with representing 3D solids in voxels for its compatibility with convolutional mechanism, hereby yielding a set of early 3D generation models, e.g., 3D-GAN [181] and 3D U-Net [33]. To avoid voxel's unbearable memory and computation consumption, researchers have turned their attention to point clouds [129]: the generation of point clouds can be done by combining an point-cloud autoencoder and a GAN that is trained in the latent space of the autoencoder [2]; Luo et al. [98] enable the generation of high-quality 3D point clouds using diffusion models. In fact, point-based DGMs are extremely inefficient at modeling sparse data, and the lack of neighborhood information leads to poor model performance. Point-voxel CNN (PVCNN) [93] breaks the isolation wall between point cloud and voxel, the proposal of point-voxel diffusion (PVD) can significantly improve the fidelity of generated shapes. Most recently, LION [194] combines various advancements, such as PVCNN and latent diffusion model (LDM) [139], and achieves the novel state-of-the-art quality of generation results.

However, the usability of point clouds in industrial applications is limited. More precisely, point clouds often fail to accurately represent the complex shapes and geometries encountered in industrial settings. The rendered meshes derived from these point clouds can not be used for simulations, limiting their practical utility. In contrast, meshes [96, 154] are much more commonly used form in industry, as the native representation of many CAE software and Finite Element tools. Nevertheless, due to the non-uniform representation of meshes (which consist of nodes and edges), it's non-trivial to leverage CNNs. To tackle this, MeshCNN [53] has been designed to enable the use of CNN on irregular mesh data forms, and has achieved initial success in direct mesh learning. Afterwards, substantial progress has been made in directly learning on meshes, e.g., convolutional mesh autoencoder (CoMA) [135], MeshGAN [30] and MeshingNet [199]. Most recently, researchers [32, 151, 64] in this field have greatly facilitated the solid generation task with the implicit representation signed distance field (SDF) [162].

Despite the considerable progress made in learning from various solid representations, it is more natural and advantageous to learn directly from Boundary Representation (B-Rep) [177]: B-Rep is more precise, easier to manipulate, and takes up less memory [56]. However, directly learning B-Rep entities remains challenging because B-Rep consists of parametric surfaces, parametric curves and vertices. These entities need to be trimmed and sowed to become solid models. Also, B-Reps store adjacency information for neighbouring edges and vertices, allowing the structure to provide a complete description of the final entity shape [187]. DeepCAD [182] is able to generate CAD source data, i.e., STEP files, by learning each CAD model as a sequence of modeling operations such as sketching and extruding. Following this idea, a set of works have been done, e.g., Datasets (DeepCAD [182] and Fusion 360 gallery [180]), CAD generative models [85, 188] and STEP-based learning models [102]. However, these works mainly focus on CAD models, the modeling operations of which are limited to sketching and stretching, and they are difficult to scale [66].

To directly consume the geometric and topological information from B-Rep data, recent attempts convert the B-Rep into a graph, then pass through a graph neural network (GNN), hereby performing learning tasks such as face segmentation or solid classification [5, 67, 36]. Prediction accuracy has been significantly improved by increasing the information added to the graph, i.e., from using face normal and distance to origin as node feature in CADNet [36] to additionally introducing UV-grids, trimming mask and curve geometries into the graph in UV-Net [67]. While converting B-Rep to graphs performs well in shape classification and segmentation tasks, in the field of solid generation, B-Rep is often represented as a predefined hierarchical structure and generated by autoregressive prediction of B-Rep entities. This approach has given rise to some convincing methods, such as SolidGen [66] and BrepGen [187].

It is worth noting that current solid generation research is avoiding the generation of parametric surfaces with their natural representations — non-uniform rational B-splines (NURBS) [128]. However, as a structured form of a point cloud, UV-grids suffer from inefficient memory consumption, inaccurate surface representation, and cubic growth of computing costs with the complexity of the surface.

1.5 Publications

Most of the work in this dissertation has been previously published. The content of this dissertation includes the following papers:

[167] **Jiajie Fan**, Laure Vuaille, Thomas Bäck, and Hao Wang. On the noise scheduling for generating plausible designs with diffusion models. *CoRR*, abs/2411.10848, 2023.

This work addresses the issue of poor plausibility of results generated by diffusion models by proposing a novel noise scheduling method, which is demonstrated in Chapter 2. Upon completion of this dissertation, this paper has been submitted to the *Computer Graphics Forum* and is currently under review.

[164] **Jiajie Fan**, Amal Trigui, Thomas Bäck, and Hao Wang. Enhancing plausibility evaluation for generated designs with denoising autoencoder. In Aleš Leonardis, Elisa Ricci, Stefan Roth, Olga Russakovsky, Torsten Sattler, and Gül Varol, editors, *Computer Vision – ECCV 2024*, pages 88–105, Cham, 2025. Springer Nature Switzerland.

To address the challenge of automatically evaluating result plausibility for assessing DGMs, we designed a new metric FDD in this work. See detailed description in Chapter 3.

[168] **Jiajie Fan**, Amal Trigui, Andrea Bonfanti, Felix Dietrich, Thomas Bäck, and Hao Wang. A mesh is worth 512 numbers: Spectral-domain diffusion modeling for high-dimension shape generation. *arXiv preprint arXiv/2503.06485*, 2025.

This work enables a training-free encoding method for high-dimensional shape data, where the encoded features are proven to be learnable with DGMs. We introduce this work in Chapter 4. Upon completion of this dissertation, this paper has been accepted for *2026 IEEE Conference on Artificial Intelligence (CAI)*. Following our presentation of the paper at the conference, it will be included in the IEEE Xplore.

[166] **Jiajie Fan**, Babak Gholami, Thomas Bäck, and Hao Wang. NeuroNURBS: Learning efficient surface representations for 3D solids. *CoRR*, abs/2411.10848, 2024.

In this work, we addressed the challenge of learning directly on parametric geometries (i.e., NURBS surfaces) and successfully applied it to various downstream tasks, such as solid generation and solid segmentation. We demonstrate this work in Chapter 5. This work spawned numerous subsequent peer-reviewed papers, making significant contributions to the field of AI-assisted computer-aided design, such as: from NURBS to neural network [147], TPDLF [189], GEOM-GNN [17], NURBGen [172] and BrepARG [89]. Upon completion of this dissertation, this paper has been submitted to the *ACM Transactions on Graphics* and is currently under review.

[165] **Jiajie Fan**, Laure Vuaille, Thomas Bäck, and Hao Wang. Adversarial latent autoencoder with self-attention for structural image synthesis. In *2024 IEEE Conference on Artificial Intelligence (CAI)*, pages 119–124, 2024.

As an early work in this thesis, we proposed SA-ALAE in this paper and evaluated it on real-world engineering designs — 2D blueprints of car A-pillar design. More details can be found in Section 6.2.3.

1.5.1 Other Work

Besides the papers that form the content of this thesis, I have also been involved in the conceptualization, writing, and review process of the following paper:

[110] Phillip Mueller, Talip Uenlue, Sebastian Schmidt, Marcel Kollovieh, **Jiajie Fan**, Stephan Günnemann, and Lars Mikelsons. Geodiffusion: A training-free framework for accurate 3d geometric conditioning in image generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6374–6384, October 2025. In this work, we addressed the challenge of conditioning generative models on 3D parametric geometries (via our “training-free” framework) and demonstrated its efficacy across accurate geometry alignment and downstream editing tasks.