scientific reports

Check for updates

OPEN

# Generative models struggle with kirigami metamaterials

Gerrit Felsch[1,2] & Viacheslav Slesarenko[1,2]✉

**Generative machine learning models have shown notable success in identifying architectures for metamaterials—materials whose behavior is determined primarily by their internal organization—that match specific target properties. By examining kirigami metamaterials, in which dependencies between cuts yield complex design restrictions, we demonstrate that this perceived success in the employment of generative models for metamaterials might be akin to survivorship bias. We assess the performance of the four most popular generative models—the Variational Autoencoder (VAE), the Generative Adversarial Network (GAN), the Wasserstein GAN (WGAN), and the Denoising Diffusion Probabilistic Model (DDPM)—in generating kirigami structures. Prohibiting cut intersections can prevent the identification of an appropriate similarity measure for kirigami metamaterials, significantly impacting the effectiveness of VAE and WGAN, which rely on the Euclidean distance—a metric shown to be unsuitable for considered geometries. This imposes significant limitations on employing modern generative models for the creation of diverse metamaterials.**

**Keywords** Inverse design, Machine learning, Generative models, Mechanical metamaterials, Kirigami

Mechanical metamaterials inherit unique behavior and extreme properties from their intricate internal organization. Conceptualized back at the end of the previous century[1,2], mechanical metamaterials underwent significant progress over the last thirty years[3,4], in part thanks to the revolution in additive manufacturing, enabling precise fabrication[5,6]. If classical metamaterials were mostly lattices assembled from the repeating unit cells[7], modern metamaterials often feature heterogeneous[8] or modular designs[9], exhibit multistability[10] and are even capable of performing simple computations[11–13]. Such advanced mechanical behavior relies on structure-properties relationships more complex than ever before[14]. Tailoring metamaterial architectures to achieve specific desired behavior is a challenging task, but it has become more accessible thanks to the advancements in artificial intelligence (AI) and machine learning (ML)[15,16]. While some approaches rely directly on inverse models[8,17,18], others couple forward surrogate models with iterative design improvements using, for example, genetic algorithms[19–21].

Several successful architectures, such as Variational Autoencoders (VAE)[22], Generative Adversarial Networks (GAN)[23] and the more recent Denoising Diffusion Models[24] have revolutionized image generation[25,26]. Initially developed for generation of images, these techniques have recently been adapted for the inverse design of metamaterials[27–31]. However, upon examining the application of generative models in the inverse design of mechanical metamaterials[32,33], it is apparent that they are used differently than in image generation. In image generation, the strength of these models is in their ability to narrow the design space from all possible images to a subset of admissible ones, such as generating only human faces[34]. Contrary to this, it is common to parameterize mechanical metamaterials such that every randomly chosen set of parameters is admissible[18,35], or the restrictions are trivial, such as requiring the trusses to have positive lengths. For imposing more complex restrictions on the design space, advanced generative ML techniques are required. In general, learning restrictions is not a trivial task, even within the computer science domain. In some instances, incorporating negative data or invalid designs can enhance the learning of constraints[36,37]. Another method involves directly feeding constraints as input parameters into network architectures, such as GAN[38]. Additionally, some strategies employ a dual training process, where the general generative model is trained alongside a constraint function that learns to capture structured knowledge about restrictions[39]. Despite noticeable progress in constraint learning, ML methods applied to metamaterials are still primarily used to learn the relationship between structure and properties, rather than to understand structural restrictions that should be imposed.

However, an all-admissible parameterization can come at the cost of excluding more sophisticated and sometimes optimal solutions from the design space. A good example is the metamaterials employing straight cuts in the planar sheets – often called kirigami metamaterials[40–43] – to program the desired mechanical behavior.

[1]Cluster of Excellence livMatS @ FIT – Freiburg Center for Interactive Materials and Bioinspired Technologies, 79110 Freiburg, Germany. [2]Department of Microsystems Engineering, University of Freiburg, 79110 Freiburg, Germany. ✉email: viacheslav.slesarenko@livmats.uni-freiburg.de

For instance, alternating cuts with horizontal and vertical orientations within the sheet (Fig. 1a) can lead to the manifestation of auxetic behavior via the rotating squares mechanism[44]. Further, it was shown[45] that adding random rotations to the base alternating structure (Fig. 1b) could be a method to program the behavior of the resulting metamaterial. The greater the maximum deviation from the base structure, the greater the achievable range of material properties. It is important to note that rotations for each cut were limited to prevent intersections that can cause undesired effects such as stress concentration or even lead to disconnected regions. While limiting the deviations from the initial structure helps in preventing intersections, it simultaneously disqualifies the majority of intersection-free configurations, such as the one shown in Fig. 1c.

While kirigami metamaterials have been the focus of multiple studies employing machine learning techniques, such studies were either limited to property predictions[46], or employed strict restrictions to get all-admissible parameterizations[47,48]. This prompts the question of why ML is not more extensively used to learn design restrictions in metamaterials in a manner akin to its application in image generation. In this manuscript, we show the fundamental problem in generation of kirigami metamaterials, associated with a non-trivial design space and absence of a good similarity metric. We discuss why certain classical ML algorithms can learn these design restrictions while others cannot. Through this analysis, we also aim to highlight the presence of survivorship bias in the existing literature on generative AI in metamaterial design, which arises from considering only metamaterials with favorable design spaces.

In general, kirigami metamaterials present a case where humans outperform machines in an intuitive task. It is extremely easy for a human to generate intersection-free samples after examining just a dozen valid examples. In contrast, standard generative models struggle with this seemingly simple task. This contrast underscores the importance of either adapting existing methods or developing new approaches for generating metamaterials with complex restrictions. Furthermore, we demonstrate that the challenge of generating the aforementioned kirigami metamaterials serves as an effective benchmark for assessing the ability of generative algorithms to learn general design space restrictions.

## Results

### Problem statement

Most generative design algorithms adhere to a similar core concept. A dataset filled with examples of what to generate, e.g. admissible metamaterials, is presented to the corresponding algorithm that learns from this information to generate similar data. The difficulty lies in determining what qualifies as "similar data". There are two approaches to this, which can also be combined together. The first approach views the dataset as samples from a probability distribution, where certain combinations of parameters are more or less likely than others. In this case, similarity is assessed by comparing the distribution of the generated data with that of the example data. The second approach performs a direct comparison between samples. For this, a specific sample-to-sample similarity metric must be chosen. The most commonly used one is the Euclidean distance (ED). In a n-dimensional space $\mathbb{R}^n$, the ED between two samples $x$ and $\hat{x}$ is calculated as follows:

$$D_E(x,\hat{x}) = \sqrt{\sum_{i=1}^{n}(x_i - \hat{x}_i)^2} \tag{1}$$

While other similarity measures might be more effective for image generation[49], ED has still been shown to yield good results, and it is a part of the original formulations for several generative algorithms[22,50]. For mechanical metamaterials, on the other hand, ED may not be the most appropriate choice for measuring similarity. For

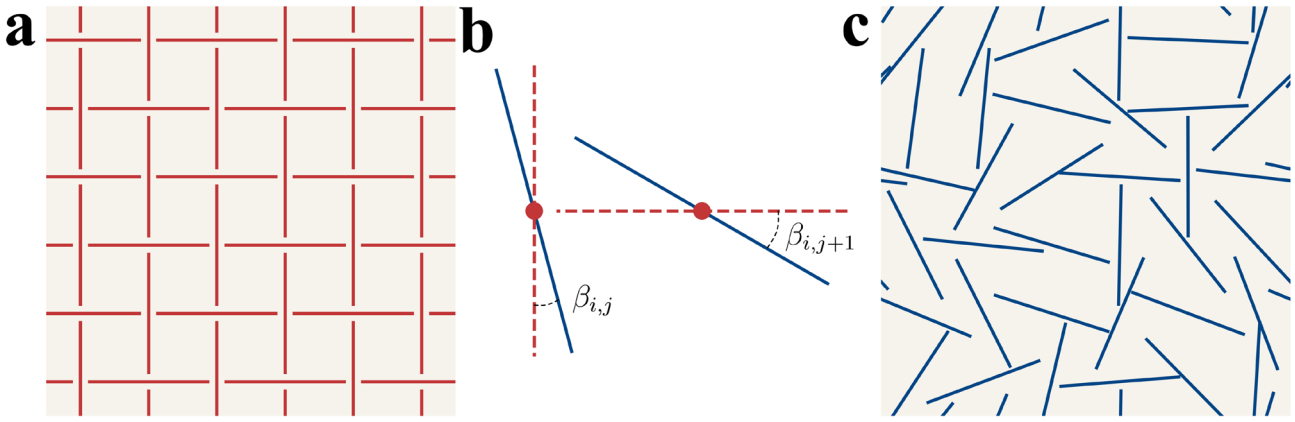



**Figure 1.** Mechanical metamaterials based on straight cuts. (**a**) Initial $6 \times 6$ pattern with alternating cuts. This architecture gives rise to auxetic behavior through the rotating squares mechanism. (**b**) Perturbation of the initial architecture is performed by adding rotations $\beta_{i,j}$ to each cut. The absolute value of rotations is capped by parameter $\beta_{max}$. (**c**) Resulting admissible design without intersections between cuts. The likelihood of obtaining intersection-free sample through random rotations ($\beta_{max} = 90°$) does not exceed 0.001%.

illustration, consider Fig. 2a, which displays three different configurations for two neighboring cuts with angles to the vertical direction as follows: **A)** [5°,4°], **B)** [−5°,−3°] **C)** [65°,−45°]. We can pose the question: which two configurations are most similar? At first glance, the answer (**A**, **B**) appears straightforward, which aligns with ED since $D_E(\mathbf{A},\mathbf{B}) < D_E(\mathbf{A},\mathbf{C}) < D_E(\mathbf{B},\mathbf{C})$, where

$$\begin{aligned} D_E(A,B) &= \sqrt{(5+5)^2+(4+3)^2} \approx 12.206 \\ D_E(A,C) &= \sqrt{(5-65)^2+(4+45)^2} \approx 77.466 \\ D_E(B,C) &= \sqrt{(-5-65)^2+(-3+45)^2} \approx 81.633 \end{aligned} \tag{2}$$

However, this conclusion overlooks a crucial point. Dark regions in Fig. 2b represent pairs of angles where two neighboring cuts intersect, forming a non-admissible zone. The transition from configuration **A** to configuration **B** through linear interpolation (as illustrated in Fig. 2c) follows the shortest path in Euclidean space, indicated by the magenta line. This path, however, clearly passes through the non-admissible zone. Consequently, not all intermediate configurations between **A** and **B** belong to the admissible design space, despite both end configurations being intersection-free. To navigate from **A** to **B** staying within the admissible design space, a considerably longer trajectory is required, as shown by the green line in Fig. 2b. Notably, the admissible path from **A** to **B** includes passing through configuration **C** (Fig. 2d). Therefore, if no intersections are allowed, the title of most similar pair belongs to (**A**, **C**). This implies that ED might not be an appropriate measure of similarity for these simplified two-cut kirigami designs. Moreover, it suggests that generative algorithms relying on ED may not effectively learn to avoid intersections, which will be shown further. This paper analyzes the ability of the four most-common generative design algorithms – VAE[22], GAN[23], Wasserstein GANs (WGAN)[50] and Denoising Diffusion Probabilistic Models[26] – to handle such geometrical challenges.

*Variational autoencoders*

The Variational Autoencoder (VAE), shown in Fig. 3a and introduced in 2013 by Kingma and Welling[22], is a modification of the traditional autoencoder[51] for generative design. Autoencoders transform input data into a usually lower-dimensional representation in so-called latent space and are comprised of two parts. The encoder creates a representation of the original data in the latent space while the decoder tries to reconstruct the original data from such a representation. Both parts are trained jointly by minimizing the error between the original and reconstructed data, which is usually referred to as reconstruction loss. While these traditional autoencoders can

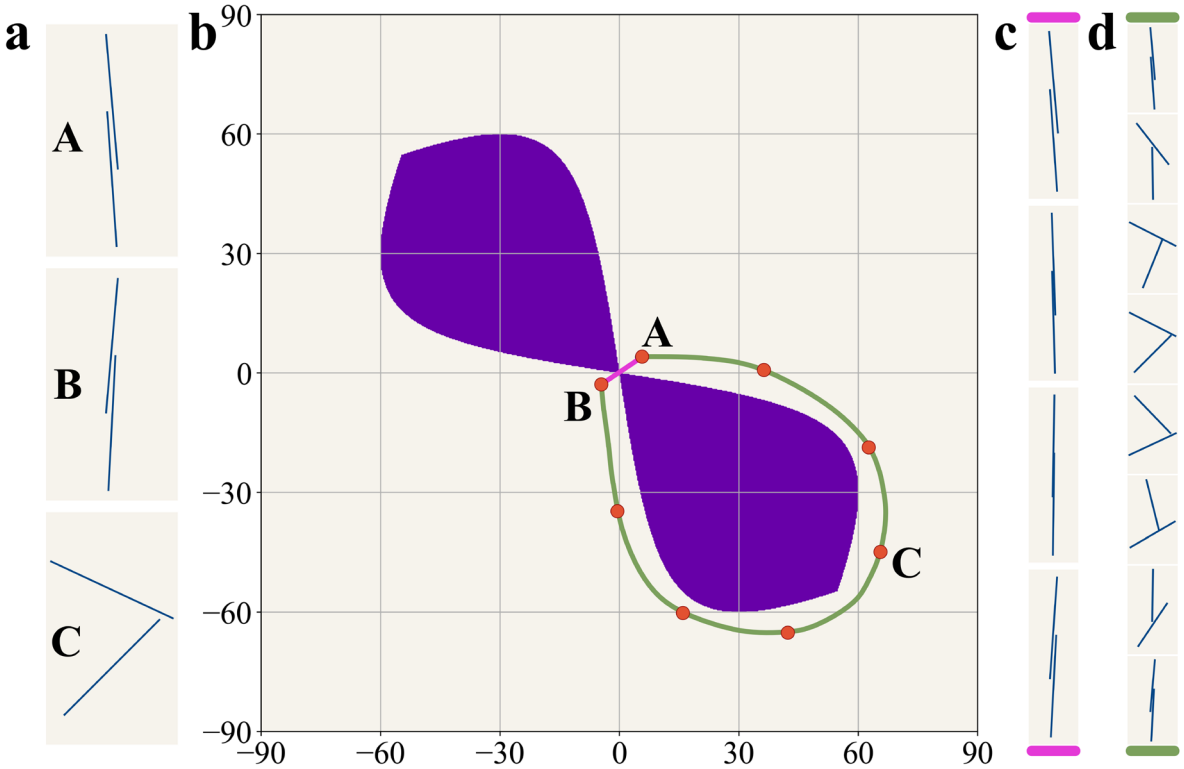



**Figure 2.** Suitability of the Euclidean Distance for two cuts. (**a**) Three different configurations (**A:** [5°,4°], **B:** [−5°,−3°] **C:** [65°,−45°]) of adjacent cuts with unit length between centers and length of $\sqrt{3}$. (**b**) The design space for the considered system with two cuts. Purple zones correspond to the angle pairs of intersecting cuts. Magenta and green lines show two possible routes between (**A**) and (**B**). (**c**) Sequence of cut positions corresponding to direct transition from (**A**) and (**B**) (magenta path). **d** Sequence of cut positions for detour path shown by green line. Note passing configuration (**C**) on a route from (**A**) to (**B**).

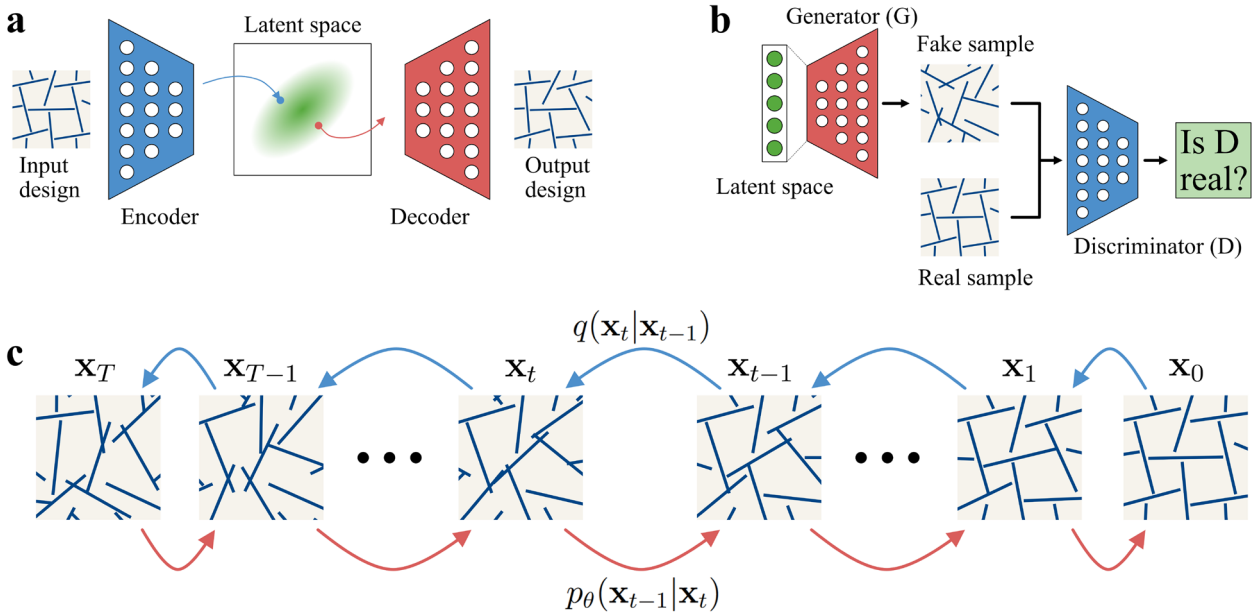



**Figure 3.** Generative approaches. (**a**) Variational Autoencoder (VAE), comprised of Encoder and Decoder stages, learns to map the designs into latent space and retrieve them back. (**b**) Generative Adversarial Network (GAN) utilizes competition between Generator and Discriminator to create samples that look real. (**c**) Denoising Diffusion Probabilistic Model (DDPM) employs sequential addition of noise to map the designs to latent space.

be used for a variety of tasks, including image denoising[52], dimensionality reduction[53], and anomaly detection[54] they lack the ability to generate new data. Unlike traditional autoencoders, VAEs map to mean and variance of a normal distribution in the latent space. An additional regularization term in the loss function compels this distribution to have zero mean and unit variance. This enables the generation of new data by sampling a representation from this normal distribution and then decoding it. While the Kullbeck-Leiber divergence[55] has been consistently used as regularization loss, the choice of reconstruction loss, in general, depends on the input data. In the original formulation, Binary Crossentropy loss was used for the MNIST dataset, where pixels are supposed to be either white or black, while the Mean Squared Error (the square of the ED) was used for continuous problems[22]. Together they yield the following combined loss function:

$$\mathscr{L}_{VAE} = \underbrace{D_{KL}\big(\mathscr{N}(\mu_x, \sigma_x), \mathscr{N}(0,1)\big)}_{\text{regularization loss}} + \underbrace{\kappa D_E(x, \hat{x})^2}_{\text{reconstruction loss}} \tag{3}$$

where $x$ is the original input, $\hat{x}$ the reconstructed one and $\mu_x, \sigma_x$ are the mean and variance of the learned distribution in the latent space, while $\kappa$ is a parameter controlling the trade-off between the two losses. Note the reliance of reconstruction loss on the Euclidean distance. We note that there have been efforts to make VAEs independent of the ED, such as by combining it with a GAN, that have shown promising results[56].

*Generative adversarial networks*

In 2014, Goodfellow et al.[23] introduced a framework for training generative models via an adversarial process, giving rise to Generative Adversarial Networks (GANs). This architecture (Fig. 3b) encompasses both a generative model $\mathscr{G}$, and a discriminative model $\mathscr{D}$, which are trained simultaneously. This training takes the form of a two-player game. While $\mathscr{D}$ is trained to distinguish between data generated by $\mathscr{G}$ and the training data, $\mathscr{G}$ is simply trained to maximize the probability of $\mathscr{D}$ making a mistake. This process reaches an equilibrium when the generative model has learned a mapping between a chosen distribution in the latent space and the data distribution, similar to the discriminator of a VAE. It has further been shown that it is equivalent to minimizing the Jensen-Shannon divergence between the distribution of the data generated by $\mathscr{G}$ and the distribution of the training data:

$$\mathscr{L}_{GAN} = D_{JS}\big(p(x), p(\hat{x})\big) = D_{KL}\big(p(x), p(\hat{x})\big) + D_{KL}\big(p(\hat{x}), p(x)\big) \tag{4}$$

This means that the vanilla GAN relies only on distances between probability distributions and not on the distance between samples, which should allow it to learn intersection avoidance.

*Wasserstein generative adversarial networks*

The Wasserstein Generative Adversarial Network (WGAN)[50] is a GAN variant where the Jensen-Shannon divergence has been replaced by the Wasserstein distance (also called Kantorovich-Rubinstein metric or Earth mover

distance) to measure the discrepancy between the distributions of the generated and the training data. This metric, which was first introduced by Kantorovich[57], is based on the principle of optimal transport. A probability distribution is seen as a distribution of mass in the design space, and the difference between distributions is measured as the minimal cost of transporting mass so that one distribution resembles the other. While it has the benefit of always staying finite and can, therefore, always provide meaningful gradients to update the generator, the Wasserstein distance ($\text{Wass}_1$) relies on an underlying metric to measure how far the mass has been transported. For the WGAN, this is usually the Euclidean metric $D_E$:

$$\text{Wass}_1(p(G(z)), p(x)) = \inf_{\pi \in \Pi(G(z)), p(x))} \mathbb{E}_{(X_1, X_2) \sim \pi} D_E(X_1, X_2) \tag{5}$$

Note that replacing the Jensen-Shannon by the Wasserstein distance makes training generally more stable, but WGAN formulation comes with the drawback of not always converging to the equilibrium point[58].

*Denoising diffusion probabilistic models*

In 2015, Sohl-Dickstein *et al.*[25] laid the foundation for a class of latent variable models, which have since become widely known as Denoising Diffusion models (Fig. 3c). The ingenious concept behind these models is that, rather than attempting to learn an arbitrary direct mapping from the latent space to the design space, they learn to reverse a diffusion process that stepwise transforms an image into its representation in the latent space. This diffusion process is defined as a Markov Chain - a stochastic model where the probability of transitioning to another state depends only on the current state - that gradually introduces noise to the image over a number of steps $T$. So for data of the from $\mathbf{x}_0 \sim q(\mathbf{x}_0)$ the forward process $q(\mathbf{x}_{1:T}|\mathbf{x}_0)$ is given as:

$$q(\mathbf{x}_{1:T}|\mathbf{x}_0) = \prod_{t=1}^{T} q(\mathbf{x}_t|\mathbf{x}_{t-1}) \tag{6}$$

where the variance of the Gaussian noise that is added in each step is altered for each step based on a variance schedule $\beta_1, \ldots, \beta_T$:

$$q(\mathbf{x}_t|\mathbf{x}_{t-1}) := \mathcal{N}(\mathbf{x}_t; \sqrt{1-\beta_t}\mathbf{x}_{t-1}, \beta_t \mathbf{I}) \tag{7}$$

This formulation possesses the advantage that when the variances of the forward process $\beta_t$ are small, the reverse process $p_\theta(\mathbf{x}_{0:T})$ can also be described as a Markov chain with Gaussian transitions. Only that in this instance, both the mean and variance of the transitions are learned[25]:

$$p_\theta(\mathbf{x}_{1:T}) = \prod_{t=1}^{T} p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) \tag{8}$$

where

$$p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) := \mathcal{N}(\mathbf{x}_{t-1}; \boldsymbol{\mu}_\theta(\mathbf{x}_t, t), \boldsymbol{\Sigma}_\theta(\mathbf{x}_t, t)), \quad p(\mathbf{x}_T) = \mathcal{N}(\mathbf{x}_T; \mathbf{0}, \mathbf{I}) \tag{9}$$

Training of the reverse process is usually performed by minimizing the variational upper bound on the negative log-likelihood using stochastic gradient descent. In 2020 Ho *et al.*[26] introduced a variant of these models called Denoising Diffusion Probabilistic Model (DDPM). When conditioned on $\mathbf{x}_0$, this bound can be rewritten using KL divergences between Gaussians:

$$\begin{aligned} L(\theta) = \mathbb{E}_q \Big[ & D_{KL}(q(\mathbf{x}_T|\mathbf{x}_0) \| p(\mathbf{x}_T)) + \sum_{t>1} D_{KL}(q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) \| p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)) \\ & - \log p_\theta(\mathbf{x}_0|\mathbf{x}_1) \Big] \end{aligned} \tag{10}$$

Furthermore, as these KL-divergences are computed between two Gaussian distributions, reparametrization enables the formulation of the variational bound as the MSE between the actual noise $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and its predicted counterpart $\epsilon_\theta(\mathbf{x}_t, t)$:

$$L(\theta) := \mathbb{E}_{t, \mathbf{x}_0, \epsilon} \left[ \| \epsilon - \epsilon_\theta(\mathbf{x}_t, t) \|^2 \right] \tag{11}$$

Here, once again, ED is encountered, though this time it is computed between instances of noise rather than instances from the design space. As this distance is derived from the initial choice of Gaussian noise in the forward process, it means that this choice limits the ability of denoising diffusion models to deal with non-Euclidean data. Therefore, research to extend diffusion models has recently focused on more general choices of forward processes and their reversal[59–61].

## Kirigami dataset

To demonstrate the practical limitations of the generative algorithms discussed in the previous sections, we assessed their ability to generate kirigami metamaterials akin to those introduced by Grima et al.[45]. We started with the alternating $6 \times 6$ pattern (Fig. 1a) and introduced random rotations for each cut, maintaining the periodicity of structure in both directions. These added rotations are denoted as $\beta_{i,j}$ for a given cut $c_{i,j}$, where $i, j = 0, .., 5$ (Fig. 1b). Correspondingly, the angle of each cut relative to the vertical direction is denoted as $\alpha_{i,j}$.

The dimensions were chosen to ensure that intersections begin to occur only if the maximum absolute value for the added rotations $\beta_{max}$ exceeds 30°. The centers of adjacent cuts are one unit apart, while the length of each cut is $l = \sqrt{3}$. Through variation of $\beta_{max}$, it is possible to control the average number of intersections in the generated data and the likelihood of randomly generating unit cells without intersections, as illustrated in Fig. 4a. It becomes virtually impossible to obtain an admissible configuration by randomly selecting 36 rotation values even if a maximum disturbance is limited to $\beta_{max} = 60°$, with only three out of a million samples containing no intersections. It is interesting to observe that the average number of intersections per sample is slightly lower for $\beta_{\max} = 90°$ compared to $\beta_{\max} = 60°$. This is caused by the existence of a number of intersection-free configurations around the alternating structure, mirrored to the one shown in Fig. 1a, where all vertical cuts are replaced by horizontal cuts and vice versa.

Figure 4a illustrates that if $\beta_{max}$ is set to less than 30°, intersections are not possible. In this case, ED serves as a suitable metric, and any linear combination of two samples remains within the admissible design space. However, once $\beta_{max}$ exceeds this threshold, the likelihood of randomly generating intersection-free samples rapidly becomes practically negligible. In general, this means that admissible designs become sparsely scattered in the design space, making it challenging for ED to measure similarity, as there is no guarantee that a linear combination of two samples is admissible anymore. Therefore the extent to which ED can effectively describe the similarity between unit cells for generative models can be indirectly controlled through $\beta_{max}$. Here, we created datasets for three distinct values of $\beta_{max}$: 20°, 60° and 90°, to see the corresponding effect on different machine learning algorithms. Each dataset comprised 20, 000 unit cells, represented as $6 \times 6$ matrices containing the values of $\alpha_{i,j}$. Given the near impossibility of randomly generating intersection-free unit cells for $\beta_{max} = 60°$ and $\beta_{max} = 90°$, we employed a randomization process. Starting from the base alternating structure, each cut was sequentially replaced with another cut, chosen randomly from those that would not create intersections. This sequence of random replacements was repeated until each cut had been replaced 200 times. Examples of admissible designs for $\beta_{max}$ of 20°, 60° and 90° are shown in Fig. 4b. After normalization, independent for each dataset, the values of $\alpha_{i,j}$ were in the range of $[-1, 1]$. In general, samples can be represented as greyscale $6 \times 6$ images where a value of $-1$ corresponds to a black pixel and a value of 1 to a white pixel, with intermediate values linearly interpolating to corresponding shades of grey.

In order to demonstrate the varying degree of reliance of four different machine learning algorithms (VAE, GAN, WGAN and DDPM) on ED, generators for each respective approach were trained on three datasets ($\beta_{max} = 20°, 60°$ and 90°) separately. Theory suggests that VAE and WGAN will learn to avoid intersections only when ED is applicable for the dataset, i.e., when rotations are limited. Proving that an approach is not learning can be conceptually more challenging than showing that one is. The inability to learn might stem from insufficient model complexity or poor parameter selection. To rule out these factors, VAE, GAN and WGAN models had almost identical architectures with consistent hyperparameters across all experiments. This approach ensures that observed differences in performance can be attributed solely to the training process. Thus, if one of the methods successfully learns to avoid intersections, the model complexity and parameter choices are not to blame.

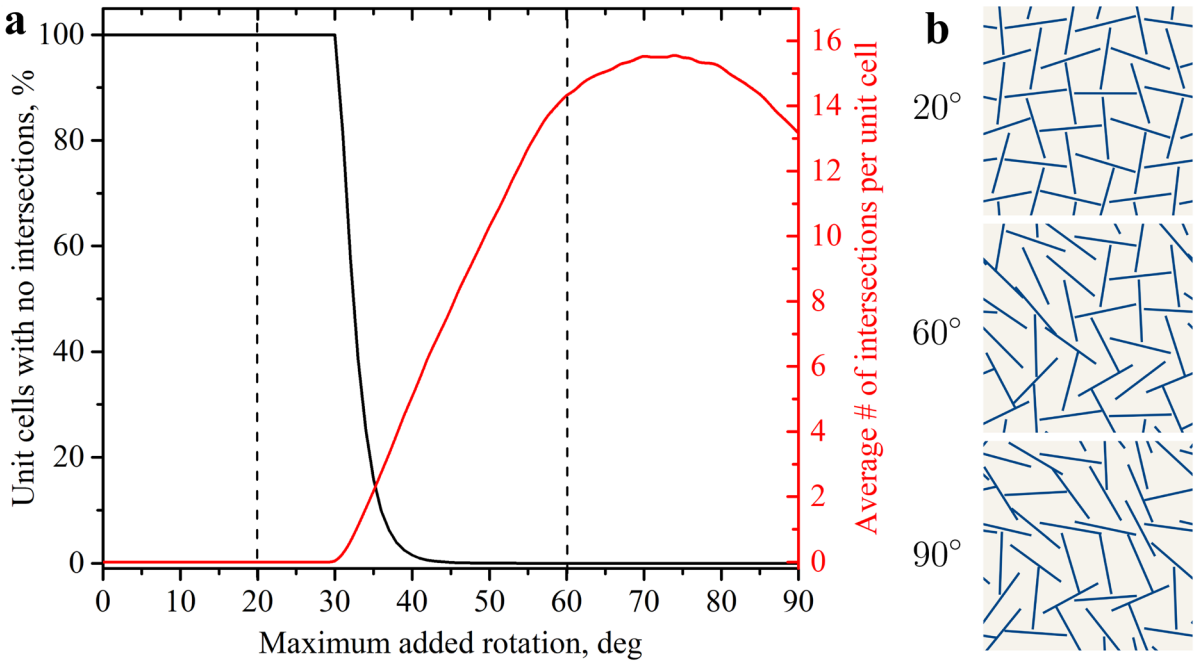



**Figure 4.** Limiting the perturbations enables control over the success rate of generation. (**a**) The average number of intersections in the samples and the likelihood of generating unit cells without intersections vs maximum deviation $\beta_{max}$ from the base structure. (**b**) Exemplary unit cells for a maximum added rotations $\beta_{max}$ of 20°, 60° and 90°.

*Implementation*

For implementing the four machine learning approaches - VAE, GAN, WGAN, and DDPM - the PyTorch deep learning framework[62] was chosen. PyTorch provides the capability to capture the periodicity of the samples by using Convolutional Neural Networks (CNN) with circular padding. CNN architecture is particularly well-suited for the unit cells of the kirigami metamaterial under investigation, as intersections can be effectively represented through hierarchical feature mapping. Intersections primarily depend on the direct neighbors of a cut, which corresponds to a $3 \times 3$ convolution, with these neighboring arrangements influencing each other. Both GAN architectures, as well as the decoder of the VAE, were based on the DCGAN framework[63]. Batch Normalization[64] was employed in VAE's encoder and sparingly in the generators/decoder to minimize batch internal dependencies[65]. The discriminator architecture was nearly identical for both GAN and WGAN, with the addition of a sigmoid activation function for the GAN due to different value range requirements imposed by the objective functions. Given the potential instability of GAN training, different learning rates were utilized for the generator and discriminator $(10^{-5}, 5 \cdot 10^{-4})$ to improve convergence towards a local Nash equilibrium[66]. Due to the restrictions on the dimensions of the latent space, a different architecture was required for DDPM[26]. A convolutional U-Net architecture[67] was chosen there, with filter numbers similar to the other generative network. For all models, Adam[68] was used as an optimizer, and a batch size was set to 32. Note that no implicit penalties for intersections were given, enabling models to learn the restrictions just by examining the training dataset. More details about the network architectures are presented in the supplementary materials.

*Euclidean case*

As previously discussed, when the maximum absolute value for the added rotations $\beta_{max}$ is set to 20°, the angles of these rotations can be chosen independently, allowing ED to effectively measure similarity between samples. This scenario is akin to the classic mechanical metamaterials with benign parameterization. In this case, to generate intersection-free unit cells, the model simply needs to understand that the angle of each cut must be confined within a specific range. Figure 5a demonstrates that after a few epochs, three out of four approaches learn to generate unit cells with none or very few intersections on average. The WGAN is capable of reducing the number of intersections to less than 0.1 on average, although it exhibits poorer stability during training. In general, if $\beta_{i,j}$ values are drawn from a uniform random distribution with $\beta_{max} = 20°$, only admissible configurations are created. The histograms in Fig. 5b reveal that after 3000 epochs, the DDPM successfully learns to emulate this uniform distribution of rotation angles $\beta_{i,j}$, maintaining the generated angles within the range of $-25°$ to 25°. Minor deviations between the training dataset and the DDPM-generated datasets at the boundary angles of $-20°$ and 20° can be attributed to the challenges of learning sharp transitions within continuous models.

Another method to evaluate the effectiveness of a model in learning the intricate constraints of the design space involves examining the overall distribution of rotation angles between neighboring cuts. If the samples generated by the trained model exhibit distributions that closely match those of the training dataset, the model can be deemed suitable for generation. Figure 6 shows 2D histograms for the training dataset and datasets generated by trained models. The color intensity represents the number of instances where a random cut and its bottom-side neighbor possess a specific combination of angles $(\beta_{i,j}, \beta_{i+1,j})$. For illustrative purposes, the first elements of these angle pairs are always from cuts at positions corresponding to vertical cuts in the initial undisturbed sample (Fig. 1a). Since no intersections are possible by construction due to $\beta_{max} = 20°$, all combinations of neighboring angles are equally likely to be observed in the training dataset, as indicated by the homogeneous square in Fig. 6. A comparison of the datasets generated by trained models with the initial training dataset reveals that

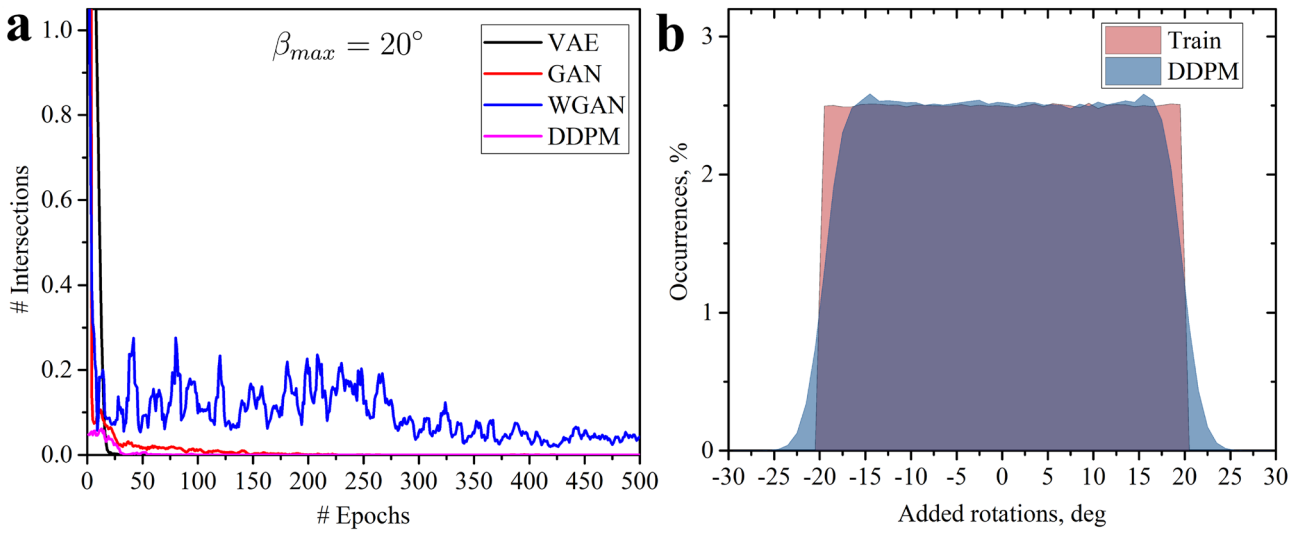



**Figure 5.** Training of models for $\beta_{max} = 20°$. (**a**) The evolution of the average number of intersections during training for unit cells generated by different machine learning approaches. An averaging over five epochs was used for curve smoothing. (**b**) Distribution of the cuts with the specific added angles $\beta_{i,j}$ in the training dataset for $\beta_{max} = 20°$ (red) and in the set generated by trained DDPM (blue).

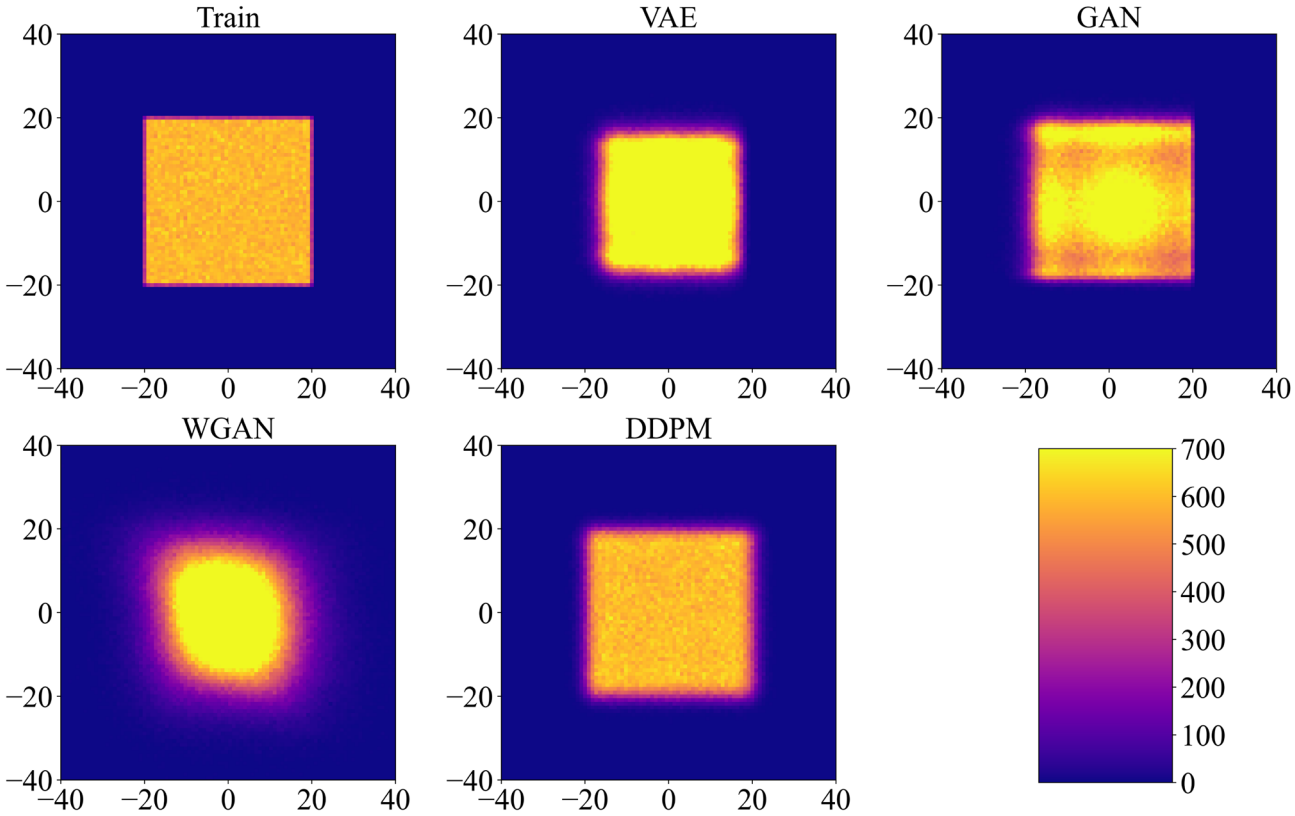



**Figure 6.** Relation between adjacent cuts for $\beta_{max} = 20°$. 2D histograms show the likelihood of angle combinations for a cut and its bottom neighbor for a maximum absolute value for $\beta = 20°$. Only angles at positions that correspond to vertical cuts in the base structure were chosen as the first elements in pairs.

all evaluated models (VAE, GAN, WGAN, DDPM) effectively capture the limits of the perturbations from the initial alternating pattern. However, it is noticeable that the VAE model slightly narrows the range of generated angles, avoiding borderline cases. This behavior is associated with the VAE's loss function (eq. 3), consisting of reconstruction and regularization terms. The inclusion of the regularization term with the hyperparameter $\kappa$ is driven by the need to facilitate a normal distribution in the latent space. Due to the regularization term, even imperfect reconstruction of the sample, usually characterized by rotation angles closer to the average values, becomes acceptable. The observed effect of narrowing the angle range persists even after careful hyperparameter tuning. In contrast with other models, the histogram for WGAN clearly shows a more Gaussian distribution rather than a uniform one, indicating the challenge in capturing the angle relationships from the training dataset. The other models (GAN and DDPM) maintain a nearly uniform 2D distribution akin to the training dataset.

*Fully random case*

When the maximal absolute value for added rotations, $\beta_{max}$, is set to 90°, the design space encompasses all possible samples with non-intersecting cuts. The final rotations of the cuts $\alpha_{i,j}$ are no longer affected by their position in the unit cell and depend only on the rotations of the neighboring cuts instead. This dependency influences the frequency at which certain rotations occur in the dataset, as some rotations are less likely to result in intersections with random neighbors. As a result, randomly generated unit cells, created by sampling cuts based on the angle distribution of the training dataset, typically have fewer intersections (represented by the black dashed line in Fig. 7a) compared to those generated from a uniform distribution (indicated by the orange dotted line). Therefore, a reduction in the number of intersections during the training of a machine learning algorithm may occur for two different reasons: the algorithm might learn to fit the angle distribution, or it might also learn the dependency of cuts on each other. In this context, the approach of randomly drawing the rotation angles of the cuts from the angle distribution of the training data serves as a valuable baseline. A decrease in the number of intersections on average during training unequivocally indicates that the model is learning the dependencies between neighboring cuts. Figure 7a demonstrates that VAE does not succeed in reducing the average number of intersections below the established baseline. Meanwhile, WGAN manages to slightly lower the number of intersections without going beyond the baseline but fails to converge, which is a common drawback of WGAN[58]. In contrast, both GAN and DDPM achieve significantly lower intersection counts, although they still fall short of generating completely intersection-free samples.

A more illustrative measure of whether different generative approaches successfully learn the dependencies between neighboring cuts, as previously mentioned, can be captured using 2D histograms (Fig. 8). Unlike the dataset with $\beta_{max} = 20°$, the distribution of angles in neighboring cut pairs for the training dataset with

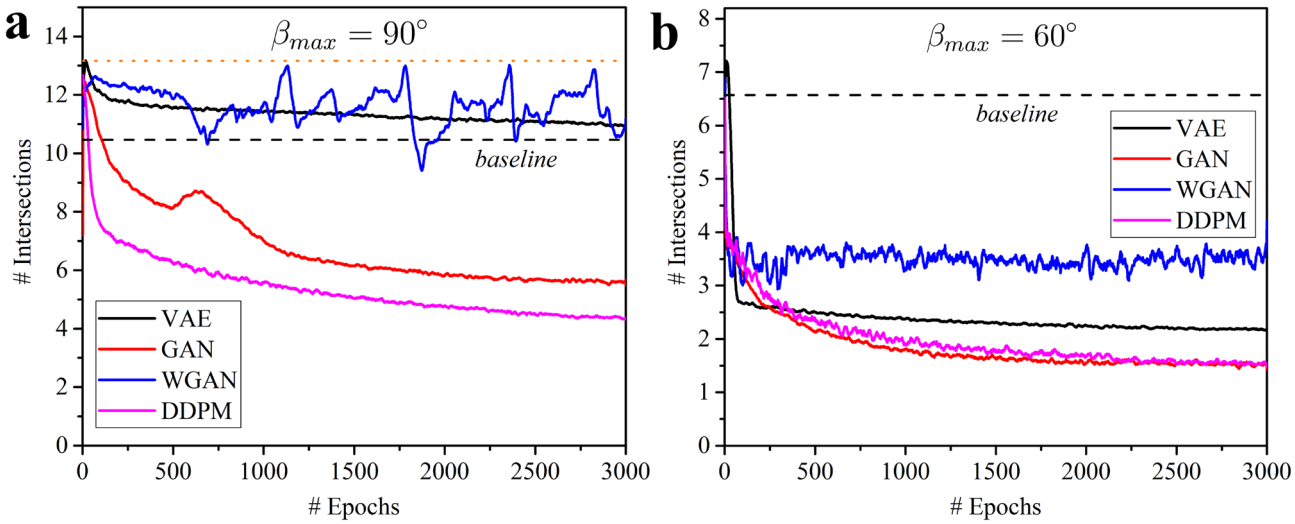



**Figure 7.** Training of models for $\beta_{max} = 90°$ (a) and $\beta_{max} = 60°$ (b). The evolution of the average number of intersections during training for unit cells generated by different machine learning approaches. An averaging over 20 epochs was used for curve smoothing. The dashed line corresponds to the average number of intersections when sampling all angles of a unit cell randomly and independently from the angle distribution of the training dataset.

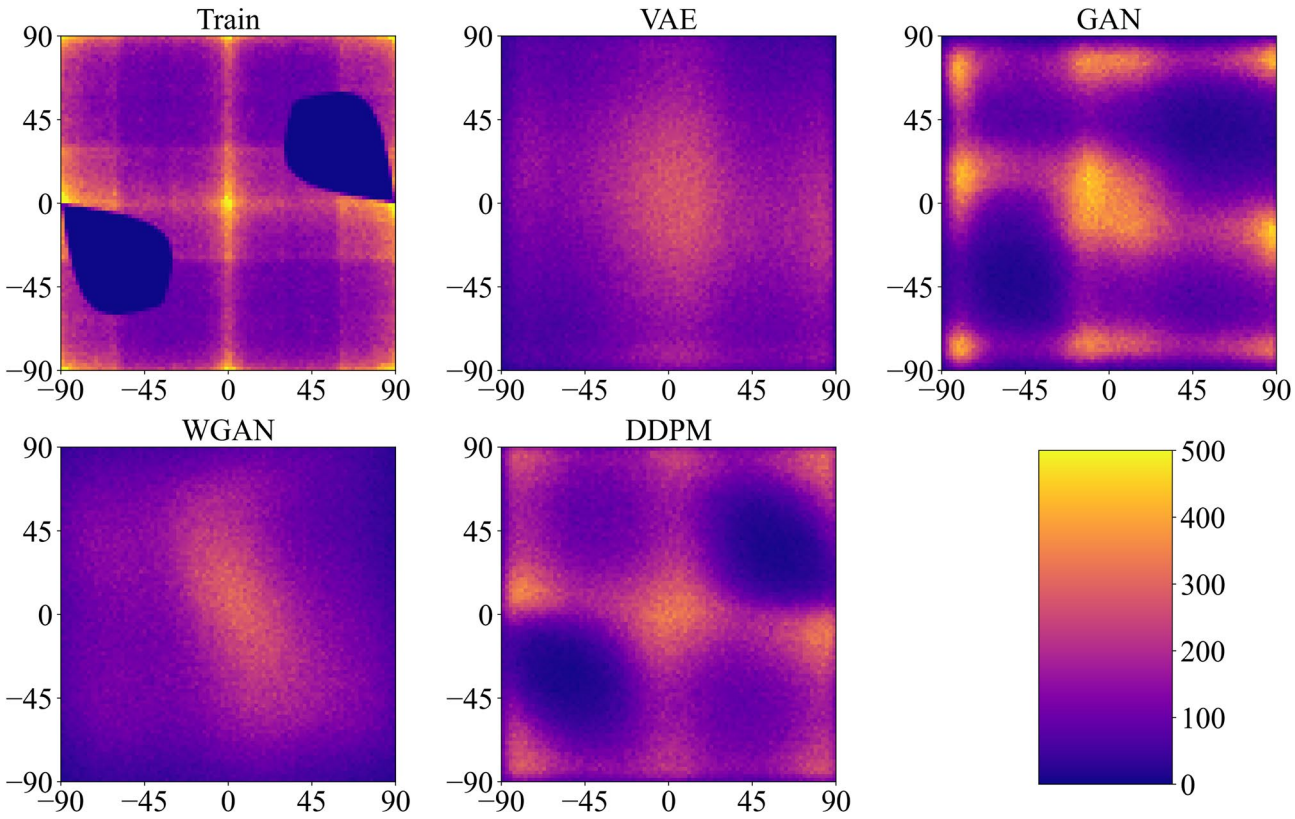



**Figure 8.** Relation between adjacent cuts for $\beta_{max} = 90°$. 2D histograms show the likelihood of angle combinations for a cut and its bottom neighbor for fully random rotations of the cuts. Only angles at positions that correspond to vertical cuts in the base structure were chosen as the first elements in pairs.

$\beta_{max} = 90°$ is no longer uniform. Recall that the dark regions in the histogram for the training dataset correspond to angle pairs where cuts intersect (compare with Fig. 2b). Therefore, avoiding these zones is an important indicator of a successful generative model. By comparing the training dataset with datasets generated by different trained models, we can categorize these models into two groups. Models belonging to the first category (VAE and WGAN) can limit the range of rotation angles but fail to capture the dependency of a cut on its bottom

neighbor. Notably, the datasets generated by trained VAE and WGAN models contain angle pairs even from non-admissible zones, likely due to relying on linear interpolation between samples. On the other hand, models from the second category (GAN and DDPM) demonstrate a much better understanding of the design space by learning additional constraints between neighboring cuts, with DDPM slightly outperforming GAN.

*Intermediate case*

When the maximum absolute value for the added rotations, $\beta_{max}$, is set to 60°, it constitutes an intermediate situation between the Euclidean and the fully random cases, with intersections still occurring in the design space. Figure 4a illustrates that it remains almost impossible to generate designs without intersections by chance, even under stricter restrictions compared to the fully random case. Figure 7b displays the training progress for different generative models on the dataset with $\beta_{max} = 60°$. Similar to the fully random scenario, all models exhibit gradual improvement during training, as evidenced by a decrease in the average number of intersections in the generated samples. However, in contrast to the $\beta_{max} = 90°$ case, all models are shown to be capable of going significantly below the baseline defined by random sampling from the training dataset distribution. Since the learning of the angle restriction ($|\beta_{i,j}| < 60°$) plays an important role, both VAE and WGAN show their capacity for performing that task, similar to the Euclidean case. Nevertheless, as in the fully random case, DDPM and GAN surpass the other two models, more accurately recreating the admissible design space after training. As seen in Fig. 7b, DDPM does not achieve a 100% success rate in generating intersection-free configurations, with an average of 1.7 intersections per sample observed. While the average number of intersections per sample provides an adequate measure of model performance (Fig. 7), it is more convenient to use another metric demonstrating the probability of generating an intersection-free sample by the corresponding model. For random sampling from a uniform distribution, approximately three structure out of a million have no intersections (success rate less than 0.001%). If random sampling is done from the training set distribution, then the success rate is around 0.5%. After training, both GAN and DDPM show approximately a 25% success rate, meaning that one out of every four generated samples has no intersections. This represents a 100,000-fold improvement over uniform distribution sampling and a more than 50-fold improvement compared to sampling from the training set. VAE and WGAN lag behind with a success rate of around 10%. These results underscore the potential of DDPM and GAN for generating metamaterials with complex geometrical restrictions.

While DDPM demonstrates the best performance in this scenario, it is noteworthy that previously weaker models, such as VAE and WGAN, also show significant improvements. Given that their network architectures are identical to those used in the fully random case, it logically follows that changes in the design space are the primary contributors to their improved performance. To avoid the need to work in a 36D space, the corresponding changes in the characteristics of the design space can be illustrated using the example of two neighboring cuts with perturbations added to the initial angles of 0° and 90° (Fig. 1b). In the case of fully random rotations ($\beta = 90°$), the fraction of angle pairs corresponding to intersecting configurations, calculated as the normalized area of non-admissible zones (shown in Fig. 2b), is approximately 16.5%. This implies that one in six randomly chosen cut pairs is non-admissible. Surprisingly, the probability of intersection under a $\beta_{max} = 60°$ constraint increases to 18.7%. This trend also holds true for $6 \times 6$ samples, where the average number of intersections is slightly higher for $\beta_{max} = 60°$ as compared to $\beta_{max} = 90°$ (Fig. 4a). Therefore, the size of the non-admissible zone alone does not account for the improved performance of generative models in the intermediate case.

At the same time, an alternative metric, closely linked to ED, can be considered. As shown prior, the suitability of ED as a similarity measure is compromised when there is no direct path between samples, as demonstrated in Fig. 2. Therefore, the shapes and positions of non-admissible zones, in addition to their overall area, could significantly influence the appropriateness of ED as a similarity measure in the examined cases. In the previous example involving two adjacent cuts, there is a 23.7% probability that a straight path connecting two random points within the admissible design space of the fully random case ($\beta_{max} = 90°$) passes through a non-admissible zone. However, when $\beta_{max}$ is set to 60°, this probability drops significantly to only 3.5%, making ED a much more suitable similarity measure. While generalizing these findings from this 2D example to a 36D case is not straightforward, the observed relationships between neighboring cuts suggest that the improved performance of certain models, particularly VAE and WGAN, in the intermediate case is likely due to a design space that aligns better with ED metric. Consequently, the ability to influence the "goodness" of the design space through the selection of $\beta_{max}$ could hold significant promise for future generative models tailored to deal with the complex design spaces of mechanical metamaterials.

## Discussion

Machine learning is deeply embedded into the research on mechanical metamaterials, as evidenced by the growing number of successful generative approaches applied to inverse design problems. However, this perceived success may be somewhat misleading, akin to survivorship bias, where only designs with "nice" parameterizations are considered. In image generation, machine learning often derives design constraints from training data, whereas, for many mechanical metamaterials, parameterizations are preselected to include these constraints. Our study extends beyond these benevolent parameterizations to examine more complex kirigami structures. One identified issue with the application of generative models to such metamaterials lies in the inapplicability of the classical Euclidean distance (ED) as a metric for assessing the similarity between unit cells.

Thus, in this study, we explore the extent to which four of the most common generative design algorithms – VAE, GAN, WGAN, and DDPM – rely on ED. Using established theoretical findings and the example of kirigami structures, we demonstrate that out of these four algorithms, both VAE and WGAN depend on this similarity measure for effective generation. This dependence limits their ability to learn complex design space constraints, although they are more suitable for generating metamaterials with simpler all-admissible

parameterizations due to their stability and lower computational costs during training as compared to GAN and DDPM. In contrast, GAN and DDPM demonstrate potential in learning design space limitations but still fall short of fully capturing these constraints. This suggests that reliance on ED is just one factor contributing to the lack of generative models for kirigami metamaterials, highlighting a need for further investigation. Simultaneously, even imperfect generation does not undermine the potential of the considered generative models. Since checking if a sample contains intersections is faster than generating one using any available algorithm, ML-based generative models capable of proposing samples with a good enough success rate, in combination with subsequent screening, might provide superior performance. At the same time, further improvement in ML-based generation of the structures is beneficial, especially considering the relative ease of conditioning the models for specific mechanical properties.

Several factors could contribute to such an improvement of ML-based generation. One idea is the incorporation of negative data during training[69,70]. When learning the admissible design space using only valid data, it does not matter whether samples are generated inside or outside the admissible space; what matters is how well the probability distributions match. Adding invalid examples during training penalizes generated samples that lie outside the admissible space, allowing the model to more accurately estimate the density of the valid data. A similar approach is to increase the model's ability to infer sharp decision boundaries from the training data. Even with negative examples included, edge cases where two cuts barely avoid intersection have a very low likelihood. As a result, these cases may not be represented in the training data, making it impossible to determine their admissibility from the data alone. Active Learning, which is concerned with choosing samples in a dataset so that the information gain is maximized[71], might offer the solution for inclusion of important border cases. Various active learning approaches have been successfully applied to constraint learning in engineering applications, such as Gaussian Processes[72] and t-METASET[73]. Recent progress in this field has been rapid, especially with the development of Generative flow networks[74]. However, these methods still need to be adapted for complex and high-dimensional spaces.

Another possible factor in the struggle of generative models (GAN and DDPM) with kirigami metamaterials is connected to the assumptions they make on the structure underlying the data. While VAE and WGAN assume the Euclidean distance can be used to measure data similarity, GAN and DDPM assume the data lies on a low-dimensional Euclidean manifold. If such a mapping does not exist and the underlying structure of the training data is more complex, GAN and DDPM will perform suboptimally. Research into extending diffusion models to more complex data geometries, such as Riemannian manifolds, is still young but promising[59–61].

Our research highlights the inherent limitations of classical generative approaches when employed in the domain of mechanical metamaterials, as opposed to their typical application in image generation. We underscore the reliance of these methods on the Euclidean distance metric, which is unsuitable for many metamaterials with intricate design spaces. The kirigami metamaterials presented here serve as an ideal benchmark for the development of new generative models, given that the complexity of their design space can be modulated by adjusting the perturbations of the initial system.

## Data availability

Data supporting the findings of this study are available from the corresponding author upon reasonable request.

Received: 6 May 2024; Accepted: 16 August 2024
Published online: 20 August 2024

## References

1. Lakes, R. Foam structures with a negative Poisson's ratio. *Science* **235**, 1038–1040. https://doi.org/10.1126/science.235.4792.1038 (1987).
2. Milton, G. W. & Cherkaev, A. V. Which elasticity tensors are realizable?. *J. Eng. Mater. Technol.* **117**, 483–493. https://doi.org/10.1115/1.2804743 (1995).
3. Qi, J. *et al.* Recent progress in active mechanical metamaterials and construction principles. *Adv. Sci.* **9**, 2102662. https://doi.org/10.1002/advs.202102662 (2022).
4. Slesarenko, V. & Pastewka, L. The bumpy road to friction control. *Science* **383**, 150–151. https://doi.org/10.1126/science.adn1075 (2024).
5. Askari, M. *et al.* Additive manufacturing of metamaterials: A review. *Addit. Manuf.* **36**, 101562. https://doi.org/10.1016/j.addma.2020.101562 (2020).
6. Surjadi, J. U. *et al.* Mechanical metamaterials and their engineering applications. *Adv. Eng. Mater.* **21**, 1800864. https://doi.org/10.1002/adem.201800864 (2019).
7. Zheng, X. *et al.* Ultralight, ultrastiff mechanical metamaterials. *Science* **344**, 1373–1377. https://doi.org/10.1126/science.1252291 (2014).
8. Kumar, S., Tan, S., Zheng, L. & Kochmann, D. M. Inverse-designed spinodoid metamaterials. *NPJ Comput. Mater.* **6**, 73. https://doi.org/10.1038/s41524-020-0341-6 (2020).
9. Doškář, M. *et al.* Wang tiles enable combinatorial design and robot-assisted manufacturing of modular mechanical metamaterials. *Extreme Mech. Lett.* **64**, 102087. https://doi.org/10.1016/j.eml.2023.102087 (2023).
10. Shan, S. *et al.* Multistable architected materials for trapping elastic strain energy. *Adv. Mater.* **27**, 4296–4301. https://doi.org/10.1002/adma.201501708 (2015).
11. Waheed, U., Myant, C. W. & Dobson, S. N. Boolean AND/OR mechanical logic using multi-plane mechanical metamaterials. *Extreme Mech. Lett.* **40**, 100865. https://doi.org/10.1016/j.eml.2020.100865 (2020).
12. Jiao, P., Mueller, J., Raney, J. R., Zheng, X. R. & Alavi, A. H. Mechanical metamaterials and beyond. *Nat. Commun.* **14**, 6004. https://doi.org/10.1038/s41467-023-41679-8 (2023).
13. El Helou, C., Grossmann, B., Tabor, C. E., Buskohl, P. R. & Harne, R. L. Mechanical integrated circuit materials. *Nature* **608**, 699–703. https://doi.org/10.1038/s41586-022-05004-5 (2022).
14. Rodríguez, S. E., Calius, E. P., Khatibi, A., Orifici, A. & Das, R. Mechanical metamaterial systems as transformation mechanisms. *Extreme Mech. Lett.* **61**, 101985. https://doi.org/10.1016/j.eml.2023.101985 (2023).

15. Guo, K., Yang, Z., Yu, C.-H. & Buehler, M. J. Artificial intelligence and machine learning in design of mechanical materials. *Mater. Horiz.* **8**, 1153–1172. https://doi.org/10.1039/D0MH01451F (2021).
16. Muhammad, Kennedy, J. & Lim, C. W. Machine learning and deep learning in phononic crystals and metamaterials–a review. *Mater. Today Communi.* **33**, 104606. https://doi.org/10.1016/j.mtcomm.2022.104606 (2022).
17. Zheng, X., Chen, T.-T., Guo, X., Samitsu, S. & Watanabe, I. Controllable inverse design of auxetic metamaterials using deep learning. *Mater. Des.* **211**, 110178. https://doi.org/10.1016/j.matdes.2021.110178 (2021).
18. Felsch, G., Ghavidelnia, N., Schwarz, D. & Slesarenko, V. Controlling auxeticity in curved-beam metamaterials via a deep generative model. *Comput. Methods Appl. Mech. Eng.* **410**, 116032. https://doi.org/10.1016/j.cma.2023.116032 (2023).
19. Chen, C.-T. & Gu, G. X. Generative deep neural networks for inverse materials design using backpropagation and active learning. *Adv. Sci.* **7**, 1902607. https://doi.org/10.1002/advs.201902607 (2020).
20. Wilt, J. K., Yang, C. & Gu, G. X. Accelerating auxetic metamaterial design with deep learning. *Adv. Eng. Mater.* **22**, 1901266. https://doi.org/10.1002/adem.201901266 (2020).
21. Deng, B. *et al.* Inverse design of mechanical metamaterials with target nonlinear response via a neural accelerated evolution strategy. *Adv. Mater.* **34**, 2206238. https://doi.org/10.1002/adma.202206238 (2022).
22. Kingma, D. P. & Welling, M. Auto-encoding variational bayes. In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14–16, 2014, Conference Track Proceedings*. https://doi.org/10.48550/arXiv.1312.6114. arXiv:1312.6114v10 (2014).
23. Goodfellow, I. *et al.* Generative adversarial nets. In *Advances in Neural Information Processing Systems* Vol. 27 (eds Ghahramani, Z. *et al.*) (Curran Associates Inc, New York, NY, 2014).
24. Bastek, J.-H. & Kochmann, D. M. Inverse design of nonlinear mechanical metamaterials via video denoising diffusion models. *Nat. Mach. Intell.* **5**, 1466–1475. https://doi.org/10.1038/s42256-023-00762-x (2023).
25. Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N. & Ganguli, S. Deep unsupervised learning using nonequilibrium thermodynamics. In *Proceedings of the 32nd International Conference on Machine Learning, Proceedings of Machine Learning Research* Vol. 37 (eds Bach, F. & Blei, D.) 2256–2265 (PMLR, Lille, France, 2015).
26. Ho, J., Jain, A. & Abbeel, P. Denoising diffusion probabilistic models. *Adv. Neural Inf. Process. Syst.* **33**, 6840–6851 (2020).
27. Zheng, L., Karapiperis, K., Kumar, S. & Kochmann, D. M. Unifying the design space and optimizing linear and nonlinear truss metamaterials by generative modeling. *Nat. Commun.* **14**, 7563. https://doi.org/10.1038/s41467-023-42068-x (2023).
28. Wang, L. *et al.* Deep generative modeling for mechanistic-based learning and design of metamaterial systems. *Comput. Methods Appl. Mech. Eng.* **372**, 113377. https://doi.org/10.1016/j.cma.2020.113377 (2020).
29. Liu, Z., Zhu, D., Rodrigues, S. P., Lee, K.-T. & Cai, W. Generative model for the inverse design of metasurfaces. *Nano Lett.* **18**, 6570–6576. https://doi.org/10.1021/acs.nanolett.8b03171 (2018).
30. Wang, J., Chen, W. W., Da, D., Fuge, M. & Rai, R. Ih-gan: A conditional generative model for implicit surface-based inverse design of cellular structures. *Comput. Methods Appl. Mech. Eng.* **396**, 115060. https://doi.org/10.1016/j.cma.2022.115060 (2022).
31. Lew, A. J. & Buehler, M. J. Single-shot forward and inverse hierarchical architected materials design for nonlinear mechanical properties using an Attention–Diffusion model. *Mater. Today* **64**, 10–20. https://doi.org/10.1016/j.mattod.2023.03.007 (2023).
32. Zheng, X., Zhang, X., Chen, T.-T. & Watanabe, I. Deep learning in mechanical metamaterials: From prediction and generation to inverse design. *Adv. Mater.* **35**, 2302530. https://doi.org/10.1002/adma.202302530 (2023).
33. Lee, D., Chen, W. W., Wang, L., Chan, Y.-C. & Chen, W. Data-driven design for metamaterials and multiscale systems: A review. *Adv. Mater.*https://doi.org/10.1002/adma.202305254 *(2023).*
34. Liu, Z., Luo, P., Wang, X. & Tang, X. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, 3730–3738 (2015).
35. Bastek, J.-H., Kumar, S., Telgen, B., Glaesener, R. N. & Kochmann, D. M. Inverting the structure-property map of truss metamaterials by deep learning. *Proc. Natl. Acad. Sci.* **119**, e2111505119. https://doi.org/10.1073/pnas.2111505119 (2022) https://www.pnas.org/doi/pdf/10.1073/pnas.2111505119.
36. Giannone, G., Regenwetter, L., Srivastava, A., Gutfreund, D. & Ahmed, F. Learning from invalid data: On constraint satisfaction in generative models. ArXiv arXiv:2306.15166 (2023).
37. Asokan, S. & Seelamantula, C. Teaching a gan what not to learn. *Adv. Neural Inf. Process. Syst.* **33**, 3964–3975 (2020).
38. Di Liello, L. *et al.* Efficient generation of structured objects with constrained adversarial networks. *Adv. Neural Inf. Process. Syst.* **33**, 14663–14674 (2020).
39. Hu, Z. *et al.* Deep generative models with learnable knowledge constraints. In *Advances in Neural Information Processing Systems* Vol. 31 (2018).
40. Jin, L. & Yang, S. Engineering Kirigami frameworks toward real-world applications. *Adv. Mater.*https://doi.org/10.1002/adma.202308560 *(2023).*
41. Zhai, Z., Wu, L. & Jiang, H. Mechanical metamaterials based on origami and Kirigami. *Appl. Phys. Rev.* **8**, 041319. https://doi.org/10.1063/5.0051088 (2021).
42. Yang, N., Zhang, M. & Zhu, R. 3D Kirigami metamaterials with coded thermal expansion properties. *Extreme Mech. Lett.* **40**, 100912. https://doi.org/10.1016/j.eml.2020.100912 (2020).
43. Misseroni, D., Pratapa, P. P., Liu, K. & Paulino, G. H. Experimental realization of tunable Poisson's ratio in deployable origami metamaterials. *Extreme Mech. Lett.* **53**, 101685. https://doi.org/10.1016/j.eml.2022.101685 (2022).
44. Grima, J. N. & Evans, K. E. Auxetic behavior from rotating squares. *J. Mater. Sci. Lett.* **19**, 1563–1565. https://doi.org/10.1023/A:1006781224002 (2000).
45. Grima, J. N., Mizzi, L., Azzopardi, K. M. & Gatt, R. Auxetic perforated mechanical metamaterials with randomly oriented cuts. *Adv. Mater.* **28**, 385–389. https://doi.org/10.1002/adma.201503653 (2016).
46. Liu, T., Sun, S., Liu, H., An, N. & Zhou, J. A predictive deep-learning approach for homogenization of auxetic Kirigami metamaterials with randomly oriented cuts. *Mod. Phys. Lett. B* **35**, 2150033. https://doi.org/10.1142/S0217984921500330 (2020).
47. Hanakata, P. Z., Cubuk, E. D., Campbell, D. K. & Park, H. S. Forward and inverse design of Kirigami via supervised autoencoder. *Phys. Rev. Res.* **2**, 042006. https://doi.org/10.1103/PhysRevResearch.2.042006 (2020).
48. Alderete, N. A., Pathak, N. & Espinosa, H. D. Machine learning assisted design of shape-programmable 3D Kirigami metamaterials. *NPJ Comput. Mater.* **8**, 1–12. https://doi.org/10.1038/s41524-022-00873-w (2022).
49. Adler, J. & Lunz, S. Banach wasserstein gan. In *Advances in Neural Information Processing Systems* Vol. 31 (eds Bengio, S. *et al.*) (Curran Associates Inc, New York, NY, 2018).
50. Arjovsky, M., Chintala, S. & Bottou, L. Wasserstein generative adversarial networks. In *Proceedings of the 34th International Conference on Machine Learning, Proceedings of Machine Learning Research* Vol. 70 (eds Precup, D. & Teh, Y. W.), 214–223 (PMLR, Lille, France, 2017).
51. Ballard, D. H. Modular learning in neural networks. In *Proceedings of the Sixth National Conference on Artificial Intelligence–Volume 1*, AAAI'87, 279-284 (AAAI Press, Washington, DC, USA, 1987).
52. Vincent, P., Larochelle, H., Bengio, Y. & Manzagol, P. -A. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, 1096–1103. https://doi.org/10.1145/1390156.1390294 (Association for Computing Machinery, New York, NY, USA, 2008).
53. Hinton, G. E. & Salakhutdinov, R. R. Reducing the dimensionality of data with neural networks. *Science* **313**, 504–507. https://doi.org/10.1126/science.1127647 (2006) https://www.science.org/doi/pdf/10.1126/science.1127647.

54. Sakurada, M. & Yairi, T. Anomaly detection using autoencoders with nonlinear dimensionality reduction. In *Proceedings of the MLSDA 2014 2nd Workshop on Machine Learning for Sensory Data Analysis*, MLSDA'14, 4–11. https://doi.org/10.1145/2689746.2689747 (Association for Computing Machinery, New York, NY, USA, 2014).
55. Kullback, S. & Leibler, R. A. On Information and Sufficiency. *Ann. Math. Stat.* **22**, 79–86. https://doi.org/10.1214/aoms/1177729694 (1951).
56. Larsen, A. B. L., Sønderby, S. K., Larochelle, H. & Winther, O. Autoencoding beyond pixels using a learned similarity metric. In *Proceedings of The 33rd International Conference on Machine Learning, Proceedings of Machine Learning Research* Vol. 48 (eds Balcan, M. F. & Weinberger, K. Q.), 1558–1566 (PMLR, New York, NY, USA, 2016).
57. Kantorovich, L. V. Mathematical methods of organizing and planning production. *Manag. Sci.* **6**, 366–422. https://doi.org/10.1287/mnsc.6.4.366 (1960).
58. Mescheder, L., Geiger, A. & Nowozin, S. Which training methods for GANs do actually converge? In *Proceedings of the 35th International Conference on Machine Learning, Proceedings of Machine Learning Research* Vol. 80 (eds. Dy, J. & Krause, A.), 3481–3490 (PMLR, Stockholm, Sweden, 2018).
59. De Bortoli, V. *et al.* Riemannian score-based generative modelling. In *Advances in Neural Information Processing Systems* Vol. 35 (eds Koyejo, S. *et al.*) 2406–2422 (Curran Associates Inc, New York, 2022).
60. Huang, C.-W., Aghajohari, M., Bose, J., Panangaden, P. & Courville, A. C. Riemannian diffusion models. In *Advances in Neural Information Processing Systems* Vol. 35 (eds Koyejo, S. *et al.*) 2750–2761 (Curran Associates Inc, New York, NY, 2022).
61. Zhuang, P. *et al.* Diffusion probabilistic fields. In *The 11th International Conference on Learning Representations* (2023).
62. Paszke, A. *et al.* Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems* Vol. 32 8024–8035 (Curran Associates Inc, New York, NY, 2019).
63. Radford, A., Metz, L. & Chintala, S. Unsupervised representation learning with deep convolutional generative adversarial networks. In *4th International Conference on Learning Representations, ICLR 2016* (eds Bengio, Y. & LeCun, Y.) (San Juan, Puerto Rico, 2016).
64. Ioffe, S. & Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, JMLR Workshop and Conference Proceedings* Vol. 37 (eds Bach, F. R. & Blei, D. M.), 448–456 (JMLR.org, Lille, France, 2015).
65. Salimans, T. *et al.* Improved techniques for training gans. In *Advances in Neural Information Processing Systems* Vol. 29 (eds Lee, D. *et al.*) (Curran Associates Inc, New York, NY, 2016).
66. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B. & Hochreiter, S. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in Neural Information Processing Systems* Vol. 30 (eds Guyon, I. *et al.*) (Curran Associates Inc, New York, NY, 2017).
67. Ronneberger, O., Fischer, P. & Brox, T. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5–9, 2015, Proceedings, Part III 18*, 234–241 (Springer, 2015).
68. Kingma, D. P. & Ba, J. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7–9, 2015, Conference Track Proceedings* (eds Bengio, Y. & LeCun, Y.) (2015).
69. Asokan, S. & Seelamantula, C. Teaching a gan what not to learn. *Adv. Neural Inf. Process. Syst.* **33**, 3964–3975 (2020).
70. Giannone, G., Regenwetter, L., Srivastava, A., Gutfreund, D. & Ahmed, F. Learning from invalid data: On constraint satisfaction in generative models. arXiv preprint arXiv:2306.15166 (2023).
71. Ren, P. *et al.* A survey of deep active learning. *ACM Comput. Surv. (CSUR)* **54**, 1–40 (2021).
72. Khatamsaz, D. *et al.* Bayesian optimization with active learning of design constraints using an entropy-based approach. *NPJ Comput. Mater.* **9**, 49 (2023).
73. Lee, D. *et al.* t-metaset: Tailoring property bias of large-scale metamaterial datasets through active learning. arXiv preprint arXiv:2202.10565 (2022).
74. Bengio, E., Jain, M., Korablyov, M., Precup, D. & Bengio, Y. Flow network based generative models for non-iterative diverse candidate generation. In *Advances in Neural Information Processing Systems* Vol. 34 (eds Ranzato, M. *et al.*) 27381–27394 (Curran Associates Inc, New York, NY, 2021).


## Acknowledgements

Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy - EXC-2193/1 - 390951807. The authors acknowledge support by the state of Baden-Württemberg through bwHPC and the German Research Foundation (DFG) through grant no INST 39/963-1 FUGG (bwForCluster NEMO).

## Author contributions

V.S. and G.F. conceived the idea, G.F. perform numerical experiments, G.F. and V.S. analysed the results, G.F. wrote a first draft, G.F. and V.S. revised the manuscript, V.S. supervised this work.

## Funding

Open Access funding enabled and organized by Projekt DEAL. Deutsche Forschungsgemeinschaft (EXC-2193/1 - 390951807).

## Competing interests

The authors declare no competing interests.

## Additional information

**Supplementary Information** The online version contains supplementary material available at https://doi.org/10.1038/s41598-024-70364-z.

**Correspondence** and requests for materials should be addressed to V.S.

**Reprints and permissions information** is available at www.nature.com/reprints.

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

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

© The Author(s) 2024