Research paper

Generative adversarial network-based inverse design of self-deploying soft kirigami composites for targeted shape transformation

Tomaž Brzin [a], M. Khalid Jawed [b], Miha Brojan [a],*

[a] *University of Ljubljana, Faculty of Mechanical Engineering, Laboratory for Nonlinear Mechanics, Aškerčeva 6, Ljubljana, SI-1000, Slovenia*
[b] *University of California, Los Angeles, Department of Mechanical and Aerospace Engineering, 420 Westwood Plaza, Los Angeles, 90024, CA, USA*

ARTICLE INFO

ABSTRACT

The design and development of morphing structures that transition from compact, transportable forms to stable, deployable configurations is crucial for advances in soft robotics, healthcare applications, and biomimetic systems. These structures often require customized functionalities and must self-deploy into precise target shapes. Therefore, the deformed shapes of such structures are usually prescribed and the parameters for their design are unknown. To obtain the fabrication parameters, the inverse problem needs to be solved, which quickly becomes quite challenging using conventional methods due to the high-dimensional nature of the inverse problem as well as the material and geometric nonlinearities. To overcome these challenges, we combine the best of the two worlds – physics and data – and present a data-driven approach for the inverse design of two-layered soft composites that utilize the principles of kirigami and strain mismatch to self-deploy into different three-dimensional shapes. At the center of our methodology is the generative adversarial network, designed to generate the necessary fabrication parameters. By using a pre-trained simulator network, we condition the generative model to generate feasible and accurate fabrication parameters that are used to make composites that deploy into the target shapes. Our findings demonstrate that the generative model is able to effectively predict kirigami patterns and pre-stretch values required to realize complex three-dimensional shapes from simple and diverse planar designs. By performing simulations and precise desktop experiments, we compare the target with deployed shapes and demonstrate the predictive capacity of the method.

1. Introduction

Morphing structures have the ability to transform from one configuration to another. They can transform from a compact configuration into a predetermined, deployable form in which they are stable and can withstand the prescribed loading conditions. As such, they have been proven to be successful in the development of soft robots and their parts (Mungekar et al., 2023; Nojoomi et al., 2018; Rus and Tolley, 2015), various self-deployable structures (Pezzulla et al., 2015; Siéfert et al., 2019; Wang et al., 2019), in healthcare applications (Brooks et al., 2022) and in the design of biomimetic structures inspired by nature (Gladman et al., 2016; Nojoomi et al., 2018).

Morphing slender structures are usually based on the mechanism principle (Holmes, 2019), consisting of rigid links and flexible joints. Another, more advanced approach to structure transformation is inspired by origami (Schenk and Guest, 2013; Dang et al., 2022), the traditional Japanese art of paper folding, which has been particularly successful in developing deployable aerospace structures (Holmes, 2019) that require little storage space but have a large surface area

when unfolded. These structures are more flexible, but still contain piece-by-piece (semi-)rigid parts and folds that serve as hinges. They may also require support structures and multiple springs to carry the elastic potential energy or other external actuators to unfold (Pezzulla et al., 2015; Siéfert et al., 2019). Unlike origami, the kirigami technique, which focuses on the proper arrangement of incisions and cutouts, does not rely on hinges or folds (Callens and Zadpoor, 2018; Choi et al., 2019). Therefore, structure transformation usually relies on the inhomogeneity of the structure, with stress gradients inducing the structure to morph into a different configuration (van Manen et al., 2018). A well-established mechanism for inducing stress gradients that activate the self-deployment or spatial transformation of an originally flat or straight structure is based on swelling. Such approach can be activated by different principles, e.g., diffusion (Pezzulla et al., 2015), temperature (Nojoomi et al., 2018) or light (Wang et al., 2019). A mechanical analog to swelling – hydrostatic pressure – is also a promising example of actuation (Siéfert et al., 2019; Jin et al., 2020). Although these mechanisms enable the production of morphing structures that

can achieve different shapes and large deformations, they require an additional external stimulus or activation energy applied during the transformation process, such as solvent, temperature, light or pressure.

In contrast to the above mechanisms, strain mismatch can be exploited to program prescribed three-dimensional (3D) shapes into flat, two-dimensional (2D) structures during fabrication that autonomously deploy after fabrication. In this approach, self-deployable structures are made by stacking and bonding individual pre-stretched layers that store elastic potential energy during the fabrication process (Caruso et al., 2017; Fan et al., 2020; Guo et al., 2020). As a result, the so-called strain-mismatched composites deform from originally flat (2D) configurations to deformed, pre-programmed 3D forms entirely on their own. They can even be stored and transported in rolled or crumpled configurations and only need to be released when desired to deploy into their final shape (Zavodnik et al., 2024).

To get the best out of these structures, it is necessary to know the exact fabrication parameters to ensure that they can be morphed into the target 3D shapes that meet all requirements and functionality. For this reason, several methods for form-finding of structures have been proposed that address different types of structures. Bletzinger et al. (2005) dealt with shells and plates loaded in the membrane state; Koohestani (2012) analyzed tensegrity structures, Su et al. (2019) investigated self-supporting reciprocal structures; and the morphing of ribbon-like structures using compressive buckling to form complex 3D shapes was studied in Xu et al. (2015, 2019) and Fan et al. (2020). To avoid possible self-collisions during the morphing phase, a method for encoding the temporal shape evolution was presented by Guseinov et al. (2020). In order to achieve an even greater number of attainable 3D shapes, the parameter space defining the fabrication parameters needs to be drastically enlarged. This could be realized, for example, by modifying the bending stiffness via distributed modulus in functionally graded composites (Kansara et al., 2023) or by precisely encoding controlled in-plane growth into the structures, as shown by Nojoomi et al. (2021). They were able to successfully inversely predict the required input strain mismatch and apply the stretched material using a 3D printer. Their algorithm was experimentally validated by producing small models of cars, fish and human heads. A similar inverse problem was also solved by van Rees et al. (2017), who fabricated 3D structures representing the shape of a flower, a human face and a canyon. Another approach to increase the parameter space is based on the aforementioned kirigami technique, for which there have already been several attempts (Xue et al., 2017; Zhang et al., 2022). In the case of kirigami, computational difficulties quickly arise, because the cutouts or designs need to be parameterized or stored as images, e.g., in 64×64 matrices. Therefore, we are dealing with an inverse problem with more than 4096 parameters, including other fabrication parameters such as the amount of the strain mismatch and material variables. Evaluation of the objective function in such inverse problems, involving geometric and material nonlinearities, becomes too expensive, which is why several assumptions have to be made. This reduces the broadness of the final 3D shapes, therefore, determining suitable fabrication parameters is still a major challenge.

To overcome these challenges, artificial intelligence (AI) and machine learning are becoming increasingly important and successful approaches in structures and materials design (Guo et al., 2021). For example, feed-forward neural networks (FNNs) have been used to inversely construct 2D binary designs for soft membranes, along with the pressure required to achieve the target shapes during inflation (Forte et al., 2022); and to predict 2D precursors for the buckling-controlled assembly of 3D frame structures (Jin et al., 2023). Such approaches save computational effort compared to the traditional optimization algorithms, but do not provide the ability to generate multiple different solutions as is common in inverse problems. A special branch of deep learning – generative modeling – has the potential to overcome these problems, as already demonstrated in engineering design (Regenwetter

et al., 2022). Generative adversarial networks (GANs) showed promising results in the field of topology optimization (Oh et al., 2019; Yu et al., 2019; Nie et al., 2021), design of soft morphing actuator beams (Brzin and Brojan, 2024) and also in the field of photonics (Kim et al., 2022; Liu et al., 2018). For example, Liu et al. (2018) extended the traditional GAN architecture with a pre-trained FNN to inversely design metasurfaces with a targeted manipulation of light behavior. Similar problems were solved by Wang et al. (2020), An et al. (2021) and Yeung et al. (2021), where a conditional GAN was used instead. Such approach eliminated the need for an additional FNN, since the critic network took over its role. In the field of materials design, GANs (Brown et al., 2023; Kim et al., 2020; Lee et al., 2024; Mao et al., 2020) and variational autoencoders (VAEs) (Cang et al., 2018; Yao et al., 2021) are popular choices for generating new material structures with tailored properties. However, generative models are not yet widely used in solving inverse problems of morphing and deployable structures. Ma et al. (2024) used a VAE coupled with the Bayesian optimization to design kirigami patterns that allow composites to self-deploy into the 3D shapes desired by the user. They showed how physics-driven machine learning optimization can be used to generate kirigami designs for making various 3D shapes – from peanuts to flowers – with excellent agreement compared to the target shapes. The only drawback of their approach is the long computational time due to the finite element simulation running within the optimization loop. While generative models have already demonstrated their ability to help develop new materials and create fabrication parameters for the inverse design of functional shapes, a purely data-driven method that does not rely on classical optimization algorithms is yet to be developed.

In this work, we develop a data-driven inverse design framework that enables rapid and on-demand generation of fabrication parameters as it does not rely on the time-consuming finite element simulations. The method is used for designing self-deployable soft kirigami composites that leverage a combination of two different concepts: strain mismatch and kirigami. The first concept provides the elastic potential energy to activate the transformation from an initial 2D into a final 3D configuration. The second concept, the kirigami, creates a specific pattern in the initial configuration with properly placed incisions and cutouts, significantly increasing the number of attainable 3D shapes. We define the problem as an inverse problem, since, starting from the target 3D shape, we are interested in the fabrication parameters for producing a composite structure that deploys into this 3D shape when used. The fabrication parameters include a combination of design variables: the amount of induced elastic potential energy in the form of a homogeneous pre-stretch of the active layer and the (kirigami) cutting pattern of the passive layer (the layer that is not pre-stretched). The nature of the problem allows it to be solved using classical optimization algorithms, which should take into account several types of nonlinearities. Since we consider slender structures that can deform significantly during the transformation process, the algorithms should take into account the theory of large displacements, the nonlinear (hyperelastic) material models and the fact that there can exist several design solutions for the same input conditions. All these nonlinearities make the algorithms complex, difficult to implement and knowledge-demanding because the user needs to be familiar with the problem to fine-tune the parameters to obtain accurate solutions. To overcome these challenges, we employ deep generative models, in particular GANs, which have already proven useful in practice, especially in the field of materials design (Kim et al., 2020; Mao et al., 2020) and photonics (An et al., 2021; Liu et al., 2018). We condition the generator network with a pre-trained simulator network to enable the generation of not just feasible fabrication parameters, but also accurate ones that guarantee transformation into target shapes.

The remainder of the paper is organized as follows. The description of the inverse problem and its solution to solve it is given in Section 2; details on creation of training data and training process are given in
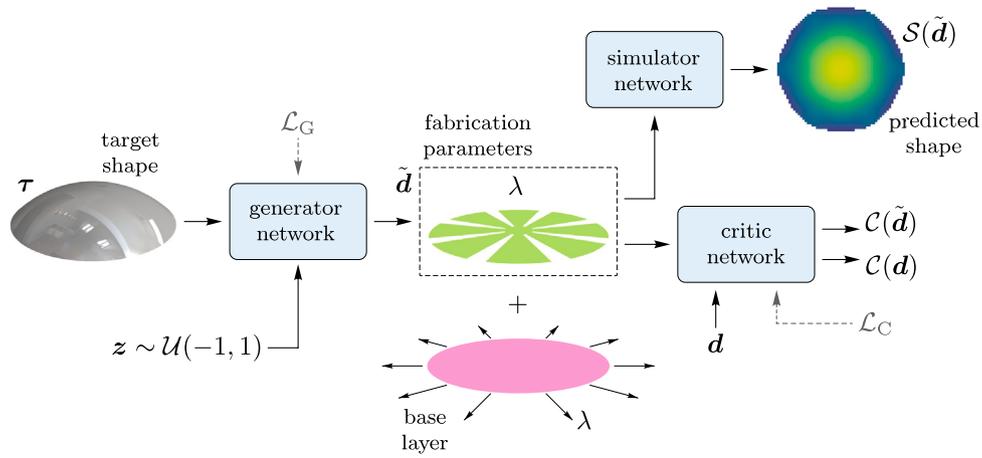
Fig. 1. The proposed inverse design framework for predicting fabrication parameters of self-deploying kirigami composites. The generator network predicts the fabrication parameters $\tilde{d}$ – the kirigami pattern of the passive layer and the pre-stretch λ of the active (base) layer – based on the given target shape. It learns from the design feasibility score $C(\tilde{d})$ provided by the critic network in the adversarial learning regime and the shape loss $\mathcal{L}_S$ calculated based on the predicted shape $S(\tilde{d})$ from the pre-trained simulator network.

Section 3; and the fabrication method to produce the composites is presented in Section 4. In Section 5, the results, including quantitative comparison with experiments, are presented and discussed; in Section 6 alternative approaches are investigated; and in Section 7 conclusions are drawn.

2. Neural networks for inverse design

We start by defining the inverse problem of self-deployable soft kirigami composites, where the (final) target shapes, i.e., the 3D shapes of the transformed composite structures, are known and the fabrication parameters for the realization of these target shapes are unknown. The composites consist of two thin elastic layers that are bonded together during the fabrication phase. The active (base) layer, Fig. 1, has a circular shape with a diameter $2\bar{r}_b$, a thickness t_b and consists of a material with the properties m_b. During fabrication, the active layer is radially pre-stretched by an amount of λ, defined as $\lambda = r_b/\bar{r}_b$, where r_b is the radius of the stretched layer. Then a passive layer with a certain kirigami pattern, part of the fabrication parameters in Fig. 1, is bonded on top of the pre-stretched active layer. The passive (kirigami) layer has a thickness t_k, consists of a material with the properties m_k and is of such size that it always fits inside a circle with the diameter $2r_b$.

Some parameters are fixed due to the practical limitations, e.g., material parameters, while others, e.g., kirigami cuts of the passive layer and pre-stretch λ, can be varied. These "free" parameters are termed hereafter as the fabrication parameters which have the greatest impact on the ability to achieve a targeted 3D shape. They are meant to be determined to ensure that the composite structures morph into the target shapes upon release after bonding is complete. The fabrication parameters are inversely predicted by the generator network, which is trained within a larger neural network framework consisting of three individual components (Fig. 1). The generator network and the critic network are part of the classical GAN, an unsupervised learning technique in which they compete with each other to generate new data with a similar distribution to the training dataset. The first is trained to generate new data – in our case the fabrication parameters $\tilde{d}$ – based on the provided information about the target shape τ, together with a vector of random noise drawn from a uniform distribution $z \sim \mathcal{U}(-1, 1)$. The second network, i.e., the critic network, is trained to recognize the authenticity of the generated fabrication parameters compared to a given sample d from the library of feasible fabrication parameters. In other words, the critic network can be considered as an agent that evaluates feasibility (e.g., takes into account only pre-stretch values that satisfy $\lambda > 1$) by predicting the score of the generated sample $C(\tilde{d})$ and the score of the training sample $C(d)$. The predicted values are used

to calculate the design feasibility (critic) loss $\mathcal{L}_C$, which evaluates the distance between the generated (fake) fabrication parameters $\tilde{d}$ and the actual (real) d from the library of feasible fabrication parameters. In this study, we used the gradient penalty (Gulrajani et al., 2017; Petzka et al., 2018) to extend the critic loss,

$$\mathcal{L}_C = C(\tilde{d}) - C(d) + \gamma \left(\max(0, \|\nabla C(\hat{d})\|_2 - 1) \right)^2,$$

where $\hat{d} = \epsilon d + (1 - \epsilon)\tilde{d}$ is the interpolation between d and $\tilde{d}$, $\epsilon \sim \mathcal{U}(0, 1)$ is a random number and γ is the penalty coefficient ($\gamma = 10$ was used in this study). No tuning was performed for this parameter, because according to Petzka et al. (2018), one-sided penalization leads to a lower sensitivity to the value of this parameter.

A pure GAN framework (Arjovsky et al., 2017; Gulrajani et al., 2017) is not sufficient to generate accurate fabrication parameters which can be used to produce composite structures with target shapes. If relying only on the critic network that is only providing the information about the feasibility, the generated fabrication parameters would be feasible and have a similar distribution to the training dataset, but the fabricated composites would deviate from the target shapes. The solution lies in additional conditioning of the generator network with a pre-trained simulator network that is able to predict the deformed 3D shape of the composite $S(\tilde{d})$ based on the provided fabrication parameters, similar to Liu et al. (2018). The predicted deformed shape is used to calculate the shape loss $\mathcal{L}_S$, which together with the feasibility score forms the generator loss,

$$\mathcal{L}_G = -C(\tilde{d}) + \omega \mathcal{L}_S.$$

The cost function – average of the generator losses in the batch – guides the generator network to generate not only feasible but also appropriate fabrication parameters that ensure morphing of the composites into user-defined target shapes, Fig. 1. Here, the parameter ω is used for the trade-off between feasibility score and shape loss. It was fine-tuned within the hyperparameter study of the neural network framework, see *Supplementary material*.

The aforementioned shape loss $\mathcal{L}_S$ is defined as a linear combination of two values, i.e., the loss between the target shape and the predicted shape, calculated according to the Huber loss function $\mathcal{H}$, and the structural similarity index measure (SSIM),

$$\mathcal{L}_S = \delta \mathcal{H}(\tau, S(\tilde{d})) + (1 - \delta)(1 - \text{SSIM}(\tau, S(\tilde{d}))).$$

The linear combination is defined by parameter δ which is of the same value as it was determined in the hyperparameter study of the simulator network. Details about the hyperparameters, along with the neural network architectures, can be found in the *Supplementary material*.

Table 1

Geometric and material properties for modeling soft kirigami composites. The units are millimeter (mm), megapascal (MPa), and kelvin (K).

Parameter	Value
$2r_b$	64 mm
t_b	1.35 mm
C_1^b	0.04918 MPa
C_2^b	-0.01545 MPa
α_b	$-1\ K^{-1}$
t_k	2.20 mm
C_1^k	0.28070 MPa
C_2^k	-0.07222 MPa
α_k	$0\ K^{-1}$

3. Dataset generation and training

In order to train the generator network within the proposed neural network framework, which we refer to as the inverse design framework, a training dataset had to be created first. This is usually done using the finite element method (FEM) or other similar numerical methods. However, such approaches are often too time-consuming, as it can take up to several days or even weeks to create a sufficiently large and diverse training dataset, as Ma et al. (2024) report. We therefore chose a hybrid approach that relies on both FEM and AI. The former was used to create an initial subset of the training dataset that was used to train the AI part – the simulator network and an additional generator network, which later helped to expand the initial subset into the full training dataset and make it more diverse.

3.1. FEM for initial training dataset

First, we have sampled 9000 different kirigami patterns by taking advantage of the 2-fold reflectional symmetry. This assumption not only allowed us faster computation times in the later FEM phase, but also gave us a good basis for the inverse design, since many man-made or naturally occurring structures exhibit this type of symmetry. Therefore, the kirigami pattern was first constructed in one quadrant by augmenting the randomly placed cutouts, similar to Ma et al. (2024). The full kirigami pattern was then created by mirroring the obtained pattern twice over the vertical and horizontal symmetry axes (details on pattern creation are explained in the *Supplementary material*). The pre-stretch values were also taken randomly from a uniform distribution $\lambda \sim \mathcal{U}(1.18, 1.22)$. The selected range may seem rather narrow, however as explained in the following paragraphs, we included also the lower values of pre-stretch to cover a wider range of target shapes.

The composites were modeled using two-layered, four-node quadrilateral shell elements and a two-parameter, non-incompressible, hyperelastic Mooney–Rivlin material model. The pre-stretch was simulated as an isotropic thermal expansion according to the relation $\lambda = \varepsilon + 1 = \alpha_b \Delta T + 1$, where ε, α_b and ΔT represent the strain, the coefficient of linear thermal expansion and the temperature difference, respectively. The thermal expansion analogy to stretching was applied only to the active layer, while the response of the passive layer (which was not stretched) was not temperature dependent ($\alpha_k = 0$ in Table 1). The composites were fixed at the coordinate $(0, 0)$ and symmetry boundary conditions were applied at the symmetry axes. The geometrical and material properties, $\boldsymbol{m}_b = \{C_1^b, C_2^b, \alpha_b\}$ and $\boldsymbol{m}_k = \{C_1^k, C_2^k, \alpha_k\}$, are shown in Table 1. The Mooney–Rivlin material constants for the base and the kirigami layer, which are made of a two-component silicone-based elastomers, Zhermack Elite Double 8 and Elite Double 32, respectively, were determined from tensile tests (see *Supplementary material* for details). All simulations were performed using ANSYS simulation software controlled by a Python script.

In each simulation we computed three deformed structure shapes for each kirigami pattern by storing the results at two intermediate load steps, in addition to the final load step where the pre-stretch reached the maximum set value λ. This approach allowed us to capture the deformed shapes also for pre-stretch values lower than in the range of between 1.18 and 1.22. Due to the symmetry and the intelligent kirigami composite-to-mesh translation, we were able to reduce the number of finite elements to only 973 without significantly decreasing the computational accuracy. This was done by linking every unique section of the sampled kirigami pattern with a pre-defined structured finite element mesh. All 26,595 samples were therefore computed in about 50 h (theoretically, 27,000 deformed shapes should have been computed from 9000 kirigami patterns, but only the unique ones were kept), since every simulation was completed in an average of 20 s. This is significantly faster compared to the simulation time of roughly 2 min that is needed if every individual pixel of the kirigami pattern is meshed with a 3×3 mesh, resulting in more than 7200 finite elements. However, linking every unique section of the kirigami pattern with a pre-defined mesh is only possible for the sampled kirigami patterns and thus beneficial for training dataset construction. The reason behind this is explained in Section 3.2.

To simplify data storage and increase compatibility with convolutional neural networks, each deformed 3D composite structure was stored as a 64×64 pixel 1-channel image of the projected z-displacements. The corresponding fabrication parameters were similarly stored as a 64×64 pixel 2-channel image, with the first channel representing the kirigami pattern and the second representing the pre-stretch value, repeated 64-times to form a complete image channel (the pre-stretch is homogeneously applied to the entire active layer, however to make it compatible with the convolutional neural networks, it was stored as a matrix representing an image channel). Every single pixel of the first channel with a value 1 represents the uncut material and every single pixel with a value -1 represents a void – cut and removed material from the passive layer.

3.2. AI for expanding the training dataset

To expand the training dataset, we used two neural networks, as schematically shown in Fig. 3a. The first, a generator network trained in a classical GAN framework (Arjovsky et al., 2017; Gulrajani et al., 2017), is used to sample new kirigami patterns given only a uniformly sampled noise $\boldsymbol{z} \sim \mathcal{U}(-1, 1)$. For simplicity and to avoid confusion with the inverse design framework, we name it as the data generation framework. The second, the simulator network, replaces FEM to drastically reduce the computation time by multiple orders of magnitude (several thousands of samples can be prepared in a few seconds). The trained simulator network achieved SSIM and MSE scores of 0.964 and 0.00030 on the test dataset, respectively, indicating that the use of the simulator network is suitable to expand the training dataset. The trained simulator network also served as a pre-trained network to help train the generator network from the inverse design framework. Details on the training of the generator network from the data generation framework and the simulator network are explained in *Supplementary material* together with the architectures of the neural networks and the hyperparameter study.

In this way, we computed 42,000 additional training samples and extended the training dataset to 68,595 training pairs consisting of fabrication parameters and the corresponding deformed structure shapes. The AI-generation and AI-simulation of 42,000 additional training samples took just 7.3 s which is a negligible amount of time. On the other hand, if we had used the classic FEM approach (same as for preparing the initial training dataset) it would theoretically take up to 78 h to expand the training dataset. Moreover, if we wanted to use the AI-generated kirigami patterns (by the generator network from the data generation framework) and calculate the deformed shapes with FEM instead of the simulator network, it would theoretically take more than 1400 h which is unfeasible.

The data distributions of the initial training dataset constructed using FEM and the AI-generated training dataset are shown in Fig. 2.
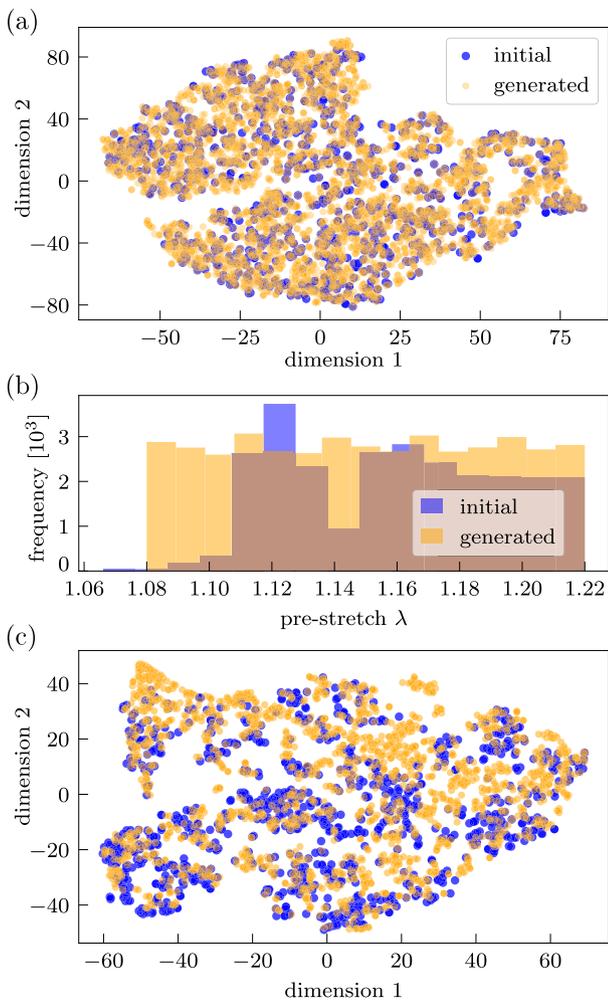
(a)

(b)

(c)

Fig. 2. Data distributions of the initial training dataset and the AI-generated training dataset used to expand the initial one, for (a) kirigami patterns, (b) pre-stretch values and (c) deformed structure shapes.

From the sampled kirigami patterns we randomly chose 1000 samples and a proportionate amount (i.e., 4600 samples) of the generated ones by the generator network from the data generation framework. We employed a pre-trained convolutional neural network VGG-16 (Simonyan and Zisserman, 2015) without the fully-connected dense layers to extract high-dimensional features of both datasets. We reduced their dimensionality into 2D by using t-distributed stochastic neighbor embedding (t-SNE) (van der Maanen and Hinton, 2008) and overlaid the plots in Fig. 2a. We can see that the data points coincide, indicating that the generator network from the data generation framework successfully modeled the data distribution of the sampled kirigami patterns. Next, we compared the distributions of the pre-stretch values that were used in the FEM simulations and the ones that were sampled to expand the training dataset, Fig. 2b. Sampling was done in range $\lambda \sim \mathcal{U}(1.08, 1.22)$ as shown in Fig. 3a. With this method we were able to capture the required range of pre-stretches more uniformly. Finally, we randomly selected 1000 FEM-calculated deformed shapes and a proportionate amount (i.e. 1600 samples, since a ratio between 26,595 FEM-computed samples and 42,000 AI-generated samples is roughly 1.6) of AI-generated ones. Similar as before, we employed VGG-16 and t-SNE to extract the features and reduced their dimensionality. The embeddings are plotted in Fig. 2c where one can see that the data points mainly coincide as expected and moreover, that few new clusters are present indicating an increase in training dataset diversity.

To check the accuracy of the expanded training dataset, we randomly selected different kirigami patterns created by the generator network from the data generation framework and the corresponding predicted deformed shapes. Four post-processed examples are shown in the leftmost column ("*sample*") and their corresponding predicted shapes in the second column from left ("*predicted shape (2D)*") in Fig. 3b. To be able to successfully convert the generated kirigami patterns, that included some unwanted artifacts, into binary images and therefore to construct finite element meshes for simulation, we employed a simple post-processing rule that converted all pixel values below 0.5 to voids (values −1) and vice-versa (pixel values equal or above 0.5 to values 1 that represent material). From now on, if the kirigami patterns are presented in green color, it means that the generator network outputs were post-processed. Similarly, if the kirigami patterns are displayed in grayscale (like in Fig. 4c), the kirigami patterns are not post-processed. For each combination of fabrication parameters, we calculated the deformed shape using FEM. In this case, the kirigami composite-to-mesh translation could not be used, because the generated kirigami patterns do not necessary overlap with the pre-defined finite element mesh sections and therefore every individual pixel had to be meshed separately. The results of the simulations are shown in the third column ("*simulated shape (2D)*") in Fig. 3b. Then, we performed a numerical comparison, i.e., computed the relative difference between the predicted and the simulated shape, which is shown in the fourth column ("*difference*") in Fig. 3b. In the last two rightmost columns in Fig. 3b ("*predicted shape (3D)*" and "*simulated shape (3D)*") we have added the plots showing the predicted and simulated shapes in 3D to help with the visualization of the deformed shapes (the colors represent the z-displacements relative to the plane $z = 0$).

3.3. Training for the inverse design

With the appropriately expanded training dataset, we started training the generator network within the inverse design framework using the pre-trained simulator network. The plot of the negative critic losses is shown in Fig. 4a and shows that the training ended at 150,000 generator iterations (the critic network was trained five times per generator iteration). The weights of the generator network were saved at predetermined generator iterations and used to evaluate the model on the test dataset. Fig. 4b shows the SSIM scores at these predetermined iterations. It can be clearly seen that they converge above 0.900, which is the point at which we stopped the training procedure. The inset in Fig. 4b shows a randomly selected target shape from the test dataset. It was used to visualize the convergence of the training procedure, Fig. 4c, where the generated kirigami patterns and the corresponding predicted 3D shapes are shown at the generator iterations marked in Fig. 4a. The quality of the generated kirigami patterns and the similarity between the target and predicted 3D shapes clearly improve and correlate with the curves shown in Fig. 4a and b. Plot of both components of the generator losses can be found in the *Supplementary material*.

4. Experiments

In order to prove the effectiveness of our proposed method, we also tested the results experimentally. To carry out the experiments, we first developed a device to homogeneously pre-stretch a circular plate (active layer) by stretching it over a cylinder, Fig. 5a. The apparatus consists of 3D-printed parts, an acrylic plate for securing the active layer and some fastening elements.

The fabrication process is illustrated in Fig. 5b–g. First, the base layer is securely attached to the flange with an acrylic plate and some fasteners (Fig. 5b). The flange is pressed over the cylinder to pre-stretch the base layer by the amount defined by λ, which is a fabrication parameter (Fig. 5c). The pre-stretching is always checked by measuring the radial displacements of the auxiliary points marked on the base layer. A 3D-printed mold representing the negative of the kirigami
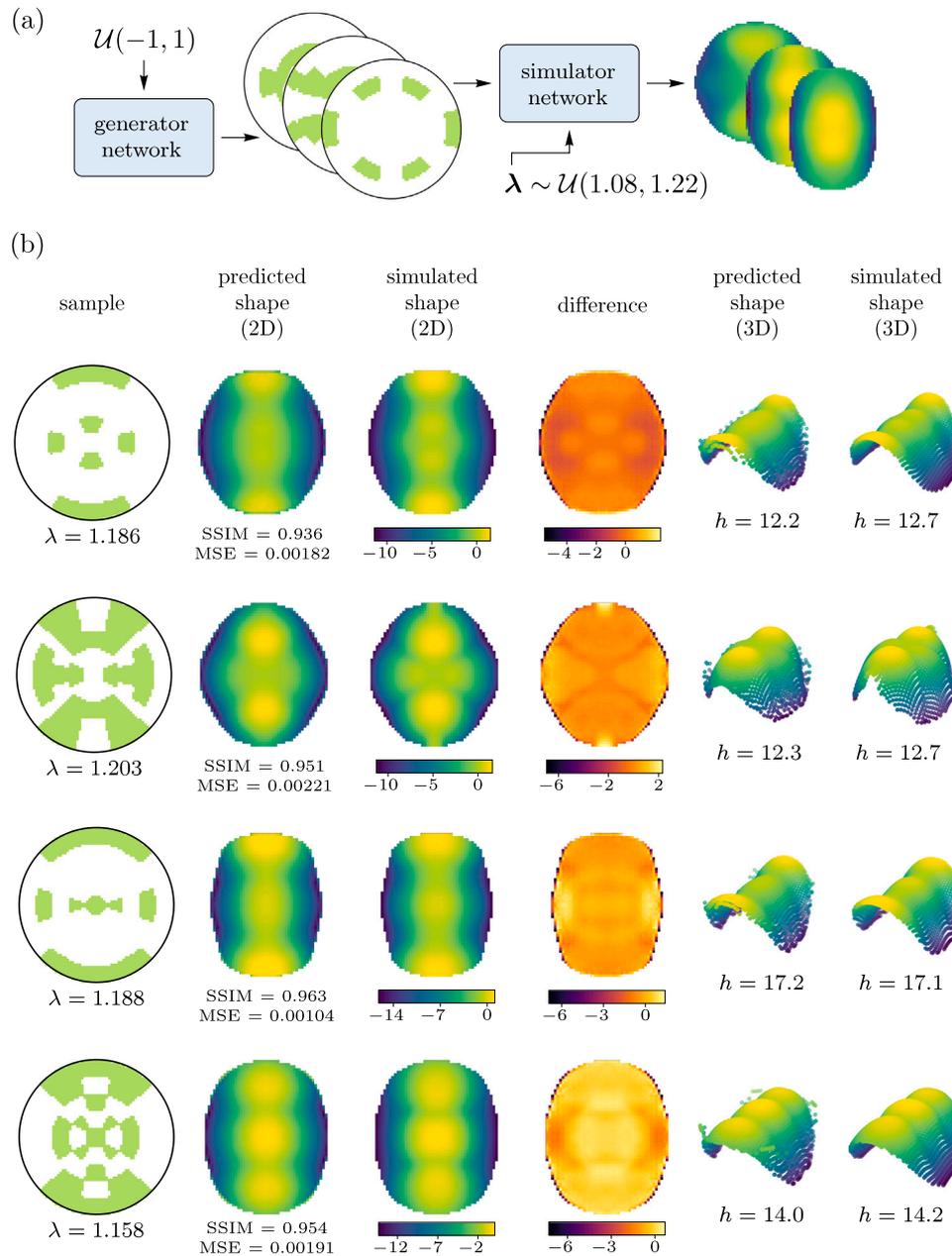
Fig. 3. Expansion of the training dataset for training the inverse design framework. (a) A trained generator network from the data generation framework is used together with the simulator network to expand the training dataset from the original approximately 27,000 to approximately 69,000 training pairs. (b) The adequacy of the new training pairs was checked with FEM (4 random samples were selected for the presentation). The columns from left to right show the generated kirigami patterns, the predicted 2D shapes using the pre-trained simulator network, the results from FEM, and the absolute error between the predicted and simulated shapes. The predicted and simulated 3D shapes are displayed on the right to help visualize the deployed composites. Symbol h denotes to the height of the structure while the colors represent the z-displacements relative to the plane $z = 0$. The units are in mm.

pattern is positioned in the center on the top of the base layer. The silicone-based elastomer still in liquid form is poured into the mold (Fig. 5d). Another acrylic plate and weights are placed on top of the mold to ensure a uniform thickness during the curing phase (Fig. 5e). After the curing is complete, the negative kirigami mold is removed (Fig. 5f). The excess material is cut and the composite self-deploys into a 3D-shaped structure (Fig. 5g).

5. Results and discussion

In this section, we present and discuss the inverse design results of various target shapes for which we inversely predicted the fabrication parameters. We took inspiration from Ma et al. (2024) and designed different target shapes, such as a dome (a), a floppy hat (b), a flower (c),

a peanut shell (d), a Pringle chip (e), a pyramid (f), and an inverted ship hull (g). The 3D target shapes were designed using the 3D modeling software SOLIDWORKS and are shown in the first column ("*target shape (3D)*") in Fig. 6.

First, the target shapes were exported to an STL format which can be read by a Python script, through which the designs were converted into 64×64 pixel images of projected z-displacements, second column ("*target shape (2D)*") in Fig. 6. A rule was implemented here to ensure that the projected center of the structure always lies on the plane at $z = 0$ mm. The symmetric parts of the normalized images in $[0, 1]$, together with the noise vector, served as inputs to the trained generator network from the inverse design framework, which generated the fabrication parameters – the kirigami patterns and the pre-stretch values λ shown in the third column ("*solution*") in Fig. 6. The generated fabrication
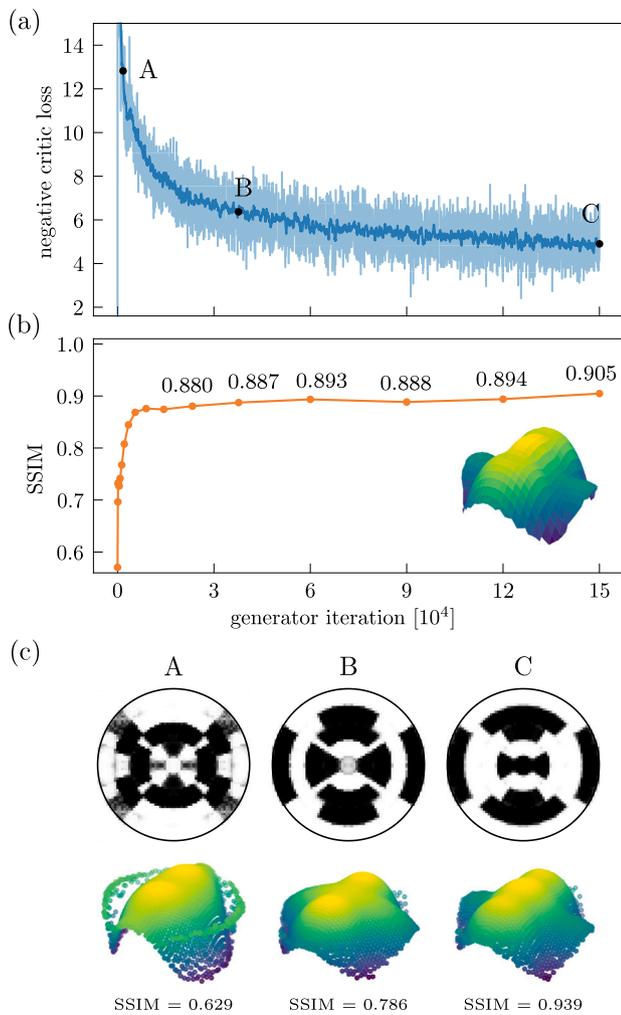
Fig. 4. Training of the inverse design framework and the development of the generated kirigami patterns. (a) Plot of negative critic losses correlating with the quality of the generated kirigami patterns – a random sample is shown to demonstrate the quality improvement. (b) During the training process, the SSIM also converges, indicating the improvement in accuracy – the predicted shape using the pre-trained simulator network converges visibly towards the target shape. (c) The quality of the generated kirigami patterns and also the accuracy of the predicted 3D shape improves during training.

Fig. 5. Fabrication method for self-deployable soft kirigami composites. (a) A circular plate can be homogeneously stretched over the cylinder by pressing down the flange. The contact surface between the plate and the cylinder is lubed to reduce friction effects as much as possible. (b–g) Pictures taken during the fabrication of the flower shape. The active layer is placed over the cylinder, fixed with a flange (b) and stretched by a certain amount (c). A negative kirigami mold is positioned in the middle, a mixture of the silicone-based elastomer is poured into it (d) and covered with an acrylic plate to cure (e). After the top layer of the composite has cured, the mold is removed (f) and the excess material is cut off. Due to the elastic potential energy stored during the fabrication, the newly formed composite structure self-deploys into the target shape by itself (g).

parameters were first verified numerically. We simulated the deformed shape using FEM and plotted the results in the fourth ("*simulated shape (2D)*") and sixth column ("*simulated shape (3D)*") in Fig. 6. The FEM simulations were performed on post-processed kirigami patterns as described in Section 3.2. The 2D representations of the simulated shapes can be directly compared with the 2D representations of the target shapes. For simplicity, we have shown the absolute errors between the simulated and target shapes in the fifth column ("*differences*") in Fig. 6. We have also provided the numerical values of the SSIM and MSE scores between the simulated and target shapes, which show a relatively good agreement. At this point, we have to emphasize that the SSIM scores served as the main metric for evaluating the results and were calculated on normalized images, so the dynamic range, i.e., the difference between the maximum and minimum allowable values, was always set to 1. If the SSIM scores were calculated on images where the differences between the minimum and maximum values were smaller than the set dynamic range, the differences between the simulated and target shapes would be less visible and would have less influence on the calculation, so the SSIM scores would be higher (above 0.900, as we tested).
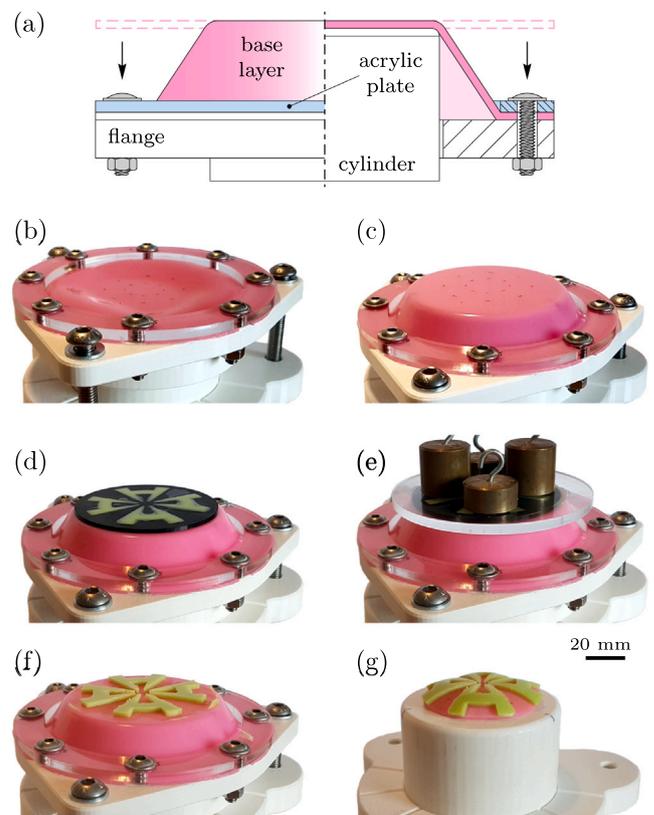
The generated fabrication parameters were additionally verified experimentally — we fabricated the structures that turned out to be very similar to the target structures in their final configuration, see the last column ("*experiment*") in Fig. 6. To facilitate the comparison between the target, the simulated and the fabricated 3D shapes, we added the height of each target, simulated and fabricated structure. By comparing the values, a relatively good agreement can be found, however in some cases the discrepancies are larger, e.g., when comparing the height of the fabricated peanut shape (Fig. 6d) to the target and simulated ones, and in case of the Pringle chip shape (Fig. 6e). The causes for the first one can be related to the fabrication process, since the simulated height is closer to the target one than the height of the fabricated structure. It is possible that unspotted air bubbles became trapped during the fabrication of both the active and passive layers, changed the properties of the material and consequentially had a significant impact on the deformed shape. The reason for the second one can be assigned to insufficiently generated fabrication parameters. Perhaps the generator network was unable to generate a design with increased pre-stretch value that would better relate to the target shape. For this reason we tried to artificially increase the pre-stretch value for this particular kirigami pattern from $\lambda = 1.114$ to $\lambda \in \{1.12, 1.13, 1.14\}$. We obtained a better match between the height of the target ($h = 18.4$ mm) and new simulated shapes ($h \in \{13.6, 14.3, 14.9\}$ mm), however the SSIM scores dropped from 0.889 to $\{0.861, 0.786, 0.720\}$, respectively, indicating a poorer shape match. Although the heights differ in these two cases, the
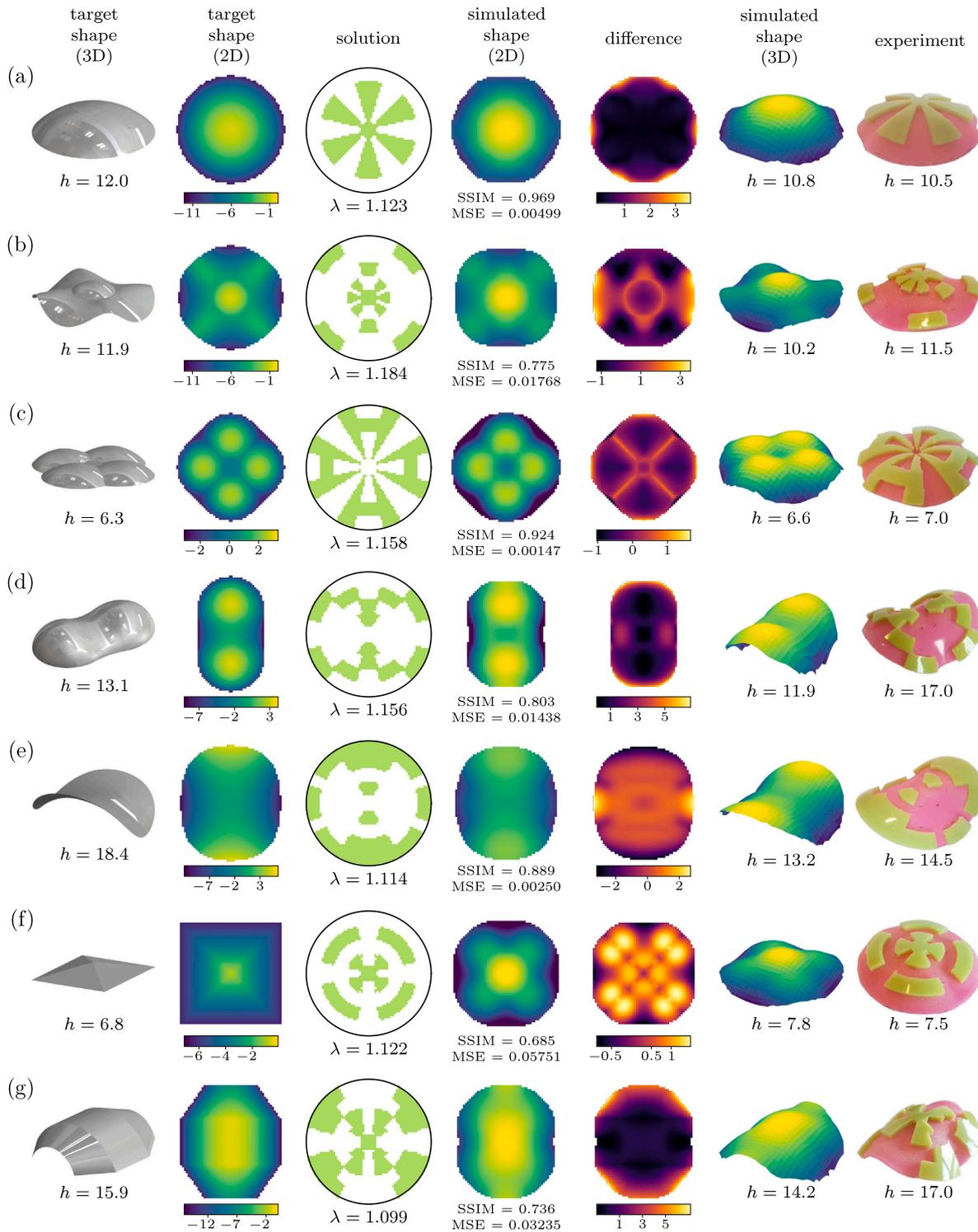
Fig. 6. Results of the inverse design for different shapes representing a dome (a), a floppy hat (b), a flower (c), a peanut shell (d), a Pringle chip (e), a pyramid (f) and an inverted ship hull (g). The columns from left to right show the designed 3D target shapes, the representation of the target shapes in 2D, the generated fabrication parameters, the simulated shapes from the generated fabrication parameters using FEM, and the absolute error between the target and simulated shapes. The 3D shapes were added to help with the visualization of the simulated shapes and to compare them with the experimentally fabricated shapes (last column from right). Symbol h denotes to the height of the structure while the colors represent the z-displacements relative to the plane $z = 0$. The units are in mm.

overall resulting shapes (simulated and fabricated) can still be linked to the target ones with significant shape similarity.

We mentioned that we considered the 2-fold reflectional symmetry when preparing the training dataset. The results show that the generator network (from the inverse design framework) is not only able to generate the fabrication parameters for shapes that have a 2-fold reflectional symmetry, i.e., the shapes of peanut shell, Pringle chip or ship hull. We have also successfully generated the solutions for the shapes with rotational symmetry (i.e., the dome shape in Fig. 6a) or a 4-fold reflectional symmetry (i.e., the shapes of the floppy hat, the

flower and the pyramid in Fig. 6b, c and f, which was also part of the Ma et al. (2024) interest. For example, we can see that the shape of the floppy hat (Fig. 6b) has a 4-fold reflectional symmetry, but the corresponding generated kirigami pattern has a 2-fold reflectional symmetry, although it is very similar to the 4-fold reflectional symmetry. In contrast to the floppy hat example, the generated kirigami pattern for the flower shape has a perfect 4-fold reflectional symmetry and the comparison shows some discrepancies. Nevertheless, the simulated and experimentally fabricated shapes are very similar to the target shape and with a little imagination the structure can be seen as a flower. In the case of the pyramid shape in Fig. 6f, the generator network (from the inverse design framework) was again able to inversely predict a perfect kirigami pattern that has 4-fold reflectional symmetry and is also similar to the design obtained by Ma et al. (2024). If we compare other kirigami patterns of the generated fabrication parameters with the results of Ma et al. (2024), we can see that we obtained different solutions for very similar target shapes. This is due to the slightly different approach for preparing the training data and also due to the fact that we are dealing with a one-to-many mapping problem, where different solutions lead to almost the same final shapes of the composite structures.

We have compared also the computational times required to generate the fabrication parameters with the competing approach from Ma et al. (2024). They report that they were able to reduce the number of forward computations from millions to just around 100, which is a remarkable achievement, however to generate one solution, roughly 5 h of computational time is still required. In contrast, our approach is able to generate tens of different designs in a matter of few seconds. The reason for this is that we use a trained neural network model (the generator network) to generate solutions, while the approach from Ma et al. (2024) uses a trained decoder network inside the optimization loop. Also, to make use of our model, the generator network needs to be trained, which requires time to prepare the training dataset (roughly 50 h as we have stated) and to complete the training (roughly 6 h were required to train the generator network from the inverse design framework in 150,000 generator iterations on NVIDIA GeForce RTX 2070 Super graphics card with 7.5 compute capability).

To explore solution diversity, we chose two target shapes that exhibit the 2-fold reflectional symmetry (the peanut shell and inverted ship hull) and two shapes that exhibit the 4-fold reflectional symmetry (the floppy hat and pyramid shapes). The generator network (from the inverse design framework) was able to generate different fabrication parameters that lead to almost the same target shapes. Three different solutions of the generated fabrication parameters are shown in Fig. 7 for the following shapes: a floppy hat (a), a peanut shell (b), a pyramid (c) and an inverted ship hull (d). In addition to the generated kirigami patterns, we have also added plots of the simulated deformed shapes, which are presented as images of the projected z-displacements and directly correspond to the representations of the target shape in the second column ("*target shape (2D)*") in Fig. 6. As above, we also calculated the SSIM and MSE scores between the simulated and target shapes.

By analyzing again the type of symmetry of the generated kirigami patterns, we came to similar conclusions as above. For the shapes that exhibit the 2-fold reflectional symmetry, i.e., the peanut (Fig. 7b) and the inverted ship hull (Fig. 7d), we can see that the generated kirigami patterns are of the true 2-fold reflectional symmetry type. Moreover, the kirigami pattern of solution 1 for the peanut shell (Fig. 7b) is very similar to the kirigami pattern obtained by Ma et al. (2024). With respect to the shapes that exhibit the 4-fold reflectional symmetry, i.e., the floppy hat (Fig. 7a) and the pyramid (Fig. 7c), we can find that most of the generated kirigami patterns are similar to the 4-fold reflectional symmetry type, with the solution 1 of the pyramid shape (Fig. 7c) being a perfect pattern of the 4-fold reflectional symmetry.

Finally, we analyzed the interpolation ability of the trained generator network from the inverse design framework. We wanted to check whether the generator network generates new kirigami patterns that are not present in the training dataset and whether these lead to better results. For this reason, the nearest neighbors (present in the expanded training dataset) of the generated kirigami patterns, shown in Fig. 6, were found. We employed the nearest neighbors algorithm (Pedregosa et al., 2011) and plotted the kirigami patterns found by the algorithm (column "*nearest neighbor*" in Fig. 8). For the FEM simulations, we assigned the same pre-stretch values as shown in Fig. 6 and plotted the results in column "*simulated shape*" of Fig. 8. The deformed shapes can be directly compared to the target and simulated shapes in Fig. 6, as the colors refer to the same scale. We calculated the differences between the target shapes (from Fig. 6) and the simulated shapes of the neighboring kirigami patterns (column "*difference*" in Fig. 8). Similarly, we also calculated the SSIM and MSE scores between the target shapes and the simulated shapes. By comparing the fabrication parameters from Figs. 6–8, we can see that the generator network was able to find new kirigami patterns that are not present in the training dataset. We can also see that in most cases, the corresponding simulated shapes from the generated fabrication parameters match the target shapes better than those computed from the neighboring kirigami patterns. In contrast, we found that the generated and neighboring kirigami patterns for the pyramid shapes (case (f) in Figs. 6 and 8) are very similar and therefore lead to similar results. Moreover, the neighboring kirigami pattern is a better solution than the solutions for the pyramid shapes presented in Fig. 7c. Similar observations were made for the inverted ship hull, Fig. 7d.

6. Alternative approaches

The analysis presented above raises a question on the use of deep generative modeling for the inverse design. A partial answer has already been given by Ma et al. (2024), who compared their proposed AI-driven approach with that of a genetic algorithm, a standard evolutionary method for optimization. They showed that the traditional evolutionary search fails in finding the fabrication parameters, is limited to the discrete design space, and is much slower compared to their proposed framework. In this section, we further address this question by evaluating our approach against two other alternative (competing) methods.

The first approach is quite similar to the one presented in the sections above, except that it does not rely on the pre-trained simulator network. Its task of conditioning the generator network is completely taken over by the critic network (Mirza and Osindero, 2014), which now evaluates the feasibility of the generated fabrication parameters and also tries to provide information about the accuracy. For simplicity and a more transparent comparison, we name this framework as the alternative inverse design framework. We trained the generator network (and the critic network) from the alternative inverse design framework with the same training dataset, with the same hyperparameters and for the same number of iterations compared to the inverse design framework, except that the architecture of the critic network had to be slightly changed to also use the target shapes as inputs. This change did not increase the number of parameters enough to affect the capacity of the network (the information about the changes made can be found in *Supplementary material*).

The second approach does not come from the family of generative models. The model used is a classical FNN, similar to the inverted simulator network used to condition the generator network from the inverse design framework. We swapped the inputs and outputs so that the inverse simulator network tries to predict the fabrication parameters based on the given target shape. We trained it with the same training dataset and with the same hyperparameters as the simulator network. Only the number of parameters was slightly changed to obtain correctly defined input and output dimensions (the information on the changes made can be found in *Supplementary material*).
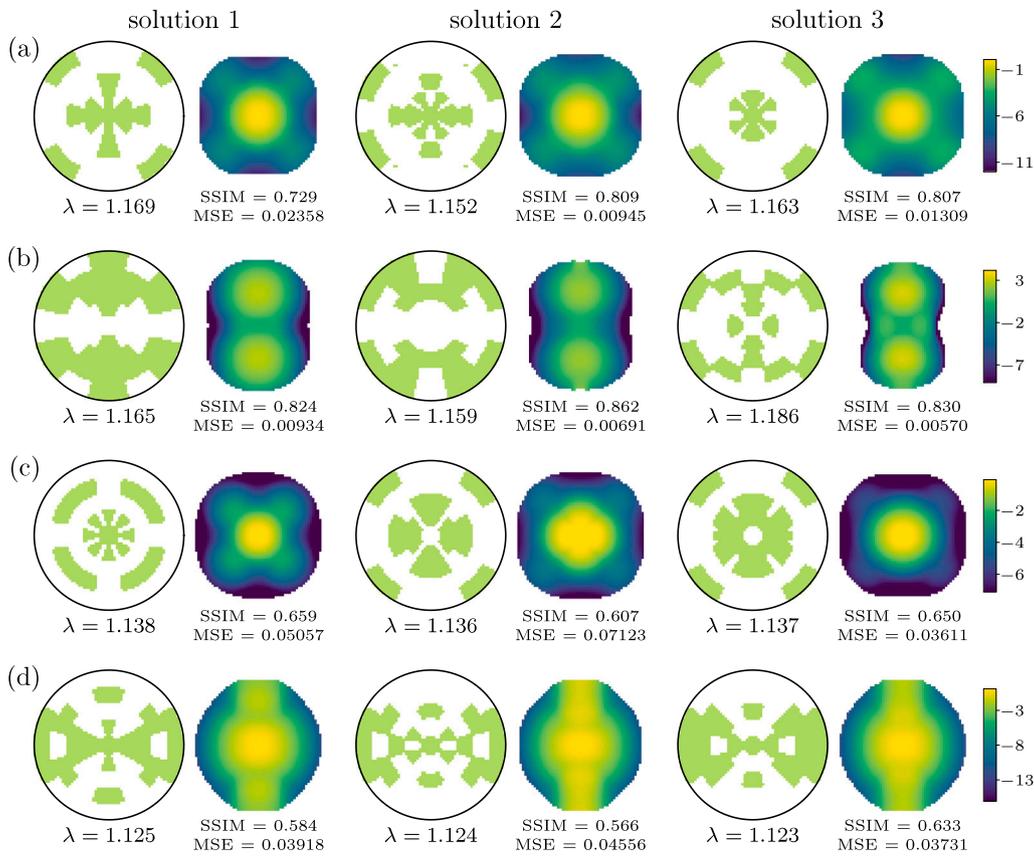
Fig. 7. Additional solutions for target shapes representing the floppy hat (a), the peanut shell (b), the pyramid c) and the inverted ship hull (d). The generator network is able to provide multiple independent fabrication parameters that result in shapes that are very similar to the target shapes. The colors represent the z-displacements relative to the plane $z = 0$. The units are in mm.

To visualize and compare the effectiveness of each inverse design approach, we selected three random training pairs from the test dataset, marked as (a), (b) and (c) in Fig. 9, where ground truths of the fabrication parameters and the corresponding target shapes are plotted in the top row ("*ground truth*"). First, we generated the solutions using the generator network from the inverse design framework. The generated fabrication parameters and the simulated corresponding deformed shapes of the composites are shown in the second row ("*inverse design framework*") in Fig. 9. We included the SSIM and MSE scores between the simulated and target shapes and obtained a strong agreement.

Next, we generated the solutions with the generator network from the alternative inverse design framework. The generated fabrication parameters and the simulated corresponding deformed shapes for all three examples are shown in the third row ("*alternative inverse design framework*") in Fig. 9. We observe that the agreement between the simulated and target shapes is equally good when compared to the inverse design framework approach, however, the generated kirigami patterns show a slightly lower quality. We also found that the generator network from the alternative inverse design framework was not able to generate drastically different fabrication parameters for the same target shape.

To analyze this weakness, we randomly selected 100 different target shapes from the test dataset. For each target shape, we generated 10 solutions using the generator networks from the inverse design framework and the alternative inverse design framework. Thus, we obtained 100 groups with 10 solutions for equal input target shapes for each method. Similar to Section 3.2, we employed VGG-16 and t-SNE to extract the features of the generated kirigami patterns from both solution datasets (each with 1000 samples) and to reduce their dimensionality, see Fig. 10a. It can be observed that the distribution of solutions from the inverse design framework is more spread out

and not clustered as in the alternative inverse design framework. Furthermore, the individual data points from the alternative inverse design framework appear to overlap in many cases, suggesting kirigami pattern solutions with similar or the same characteristics for the same target shapes. We confirmed this by calculating the average pairwise Euclidean distances between the obtained feature vectors within each group. The distributions are shown in Fig. 10b. In the dataset of the alternative inverse design framework, the distances (within interval $[0.2, 10.7]$, average 2.0 and standard deviation 2.3) are smaller than in the dataset of the inverse design framework (within interval $[11.3, 54.3]$, average 24.7 and standard deviation 7.9), implying that the kirigami patterns are less diverse.

Contrary to this finding, the generator network from the inverse design framework was able to provide multiple independent solutions for the same input target shapes, as it is shown in Fig. 7. We drew the same conclusions by visually comparing the generated kirigami patterns within each group and between the two datasets analyzed above. The kirigami patterns in the dataset generated with the generator network from the inverse design framework (four examples from one group are shown in Fig. 10c) are visually more diverse compared to the kirigami patterns generated with the generator network from the alternative inverse design framework. They have a different number of voids and a different orientation of the cutouts. No such differences are found when visually inspecting the kirigami patterns within the groups of the dataset created with the generator network from the alternative inverse design framework. The patterns are very similar, have only minor differences in the kirigami cutout layouts, and often have the same overlapping patterns within each group, such as exhibited by the four patterns from one group in Fig. 10d. These observations were made in all 100 groups and confirm the distributions shown in Fig. 10b.
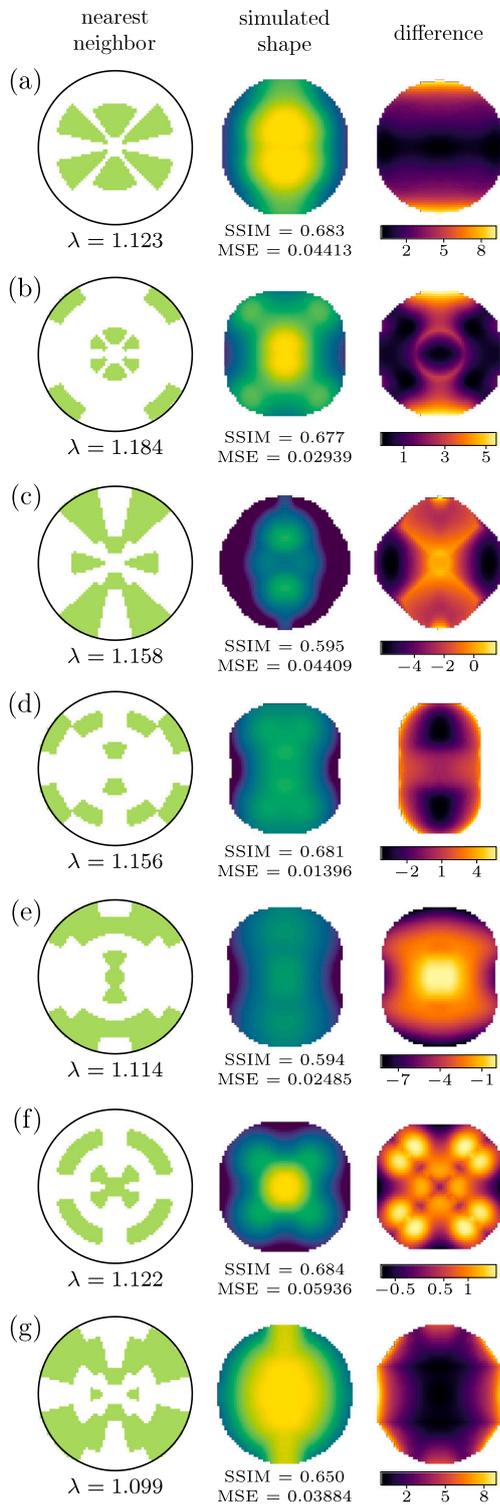
Fig. 8. Nearest neighbors of the kirigami patterns generated for the user-defined shapes shown in Fig. 6, and the corresponding simulated shapes and differences compared to the target shapes. The colors represent the z-displacements relative to the plane $z = 0$. The units are in mm.

We believe that the reason for this lies in the critic network of the alternative inverse design framework, which has also taken over the tasks of the pre-trained simulator network (that is not present in this framework). Since the inverse problem we are dealing with has a high degree of solution diversity, the critic network was not able to additionally condition the generator network to link several

different fabrication parameters with the same or similar target shapes. Therefore, the generator network from the alternative inverse design framework was not able to generate different kirigami patterns for the same input target shapes.

Finally, we also predicted the solutions using the inverse simulator network. The predicted fabrication parameters and the simulated corresponding deformed shapes for all three examples are shown in the bottom row ("*inverse simulator network*") in Fig. 9. The inverse simulator network clearly had problems with the one-to-many mapping problem, since the predicted kirigami patterns are of poor quality compared to the other inverse problem approaches. Consequently, the simulated shapes also do not match well with the target shapes. This confirms our assumptions that classical FNNs are not suitable for solving inverse problems where many solutions exist that fulfill the same or similar conditions.

In Fig. 9 we have intentionally shown the unprocessed kirigami patterns for all three inverse design approaches to emphasize the quality of the generated kirigami patterns, but the simulations were performed with the post-processed patterns. Unprocessed kirigami patterns are also shown in Fig. 10c and d.

7. Conclusions

In this study, a data-driven framework for the inverse design of self-deployable soft kirigami composites was introduced. The inverse problem was defined where the target 3D shapes are known but the fabrication parameters for the realization of such structures are not. The composites are made of two thin elastic layers that are bonded during the fabrication stage. The active layer is homogeneously pre-stretched and therefore carries the elastic potential energy, while the passive layer holds a specific kirigami cut pattern. Upon release, the structure self-deploys into a target shape. Together, the pre-stretch and kirigami form the fabrication parameters that are inversely generated by the generative model trained within the GAN architecture. The training of the generator network (from the inverse design framework) is conditioned simultaneously by the pre-trained simulator network and the critic network. Moreover, the simulator network was also used to create and utilize a hybrid training dataset constructed through a combination of FEM and AI techniques. This allowed us not only to process the data faster, but also to have a greater variety of training samples. Such approach is not mandatory for the creation of the training dataset, as it could be created entirely with FEM. However, it can be considered as a useful concept for other problems where the creation of training samples using conventional methods requires much more time and resources.

Our numerical and experimental results confirm the effectiveness of our method and show that the generator network from the inverse design framework can accurately determine the required kirigami patterns and pre-stretch values to transform simple 2D structures into intricate 3D shapes. We have shown that the generator network from the inverse design framework is able to find kirigami patterns that correspond to the target shapes in terms of the 2-fold or 4-fold reflectional symmetry type, and that it is also able to generate kirigami patterns that do not have this type of connection with the target shapes. Furthermore, the inverse design solutions for a given target shape were not limited to a single combination of fabrication parameters. The generator network from the inverse design framework was shown to be capable of successfully generating multiple independent kirigami patterns and pre-stretches for the same target shape, not present in the training dataset. However, this is not the case when using the generator network from the alternative inverse design framework, which is only conditioned by the critic network, simultaneously providing information about the feasibility and shape. Our results indicate that it is better to split the task of the necessary feedback for the conditioning of the generator network. This was done in the inverse design framework, where the feedback information was divided into two distinct groups
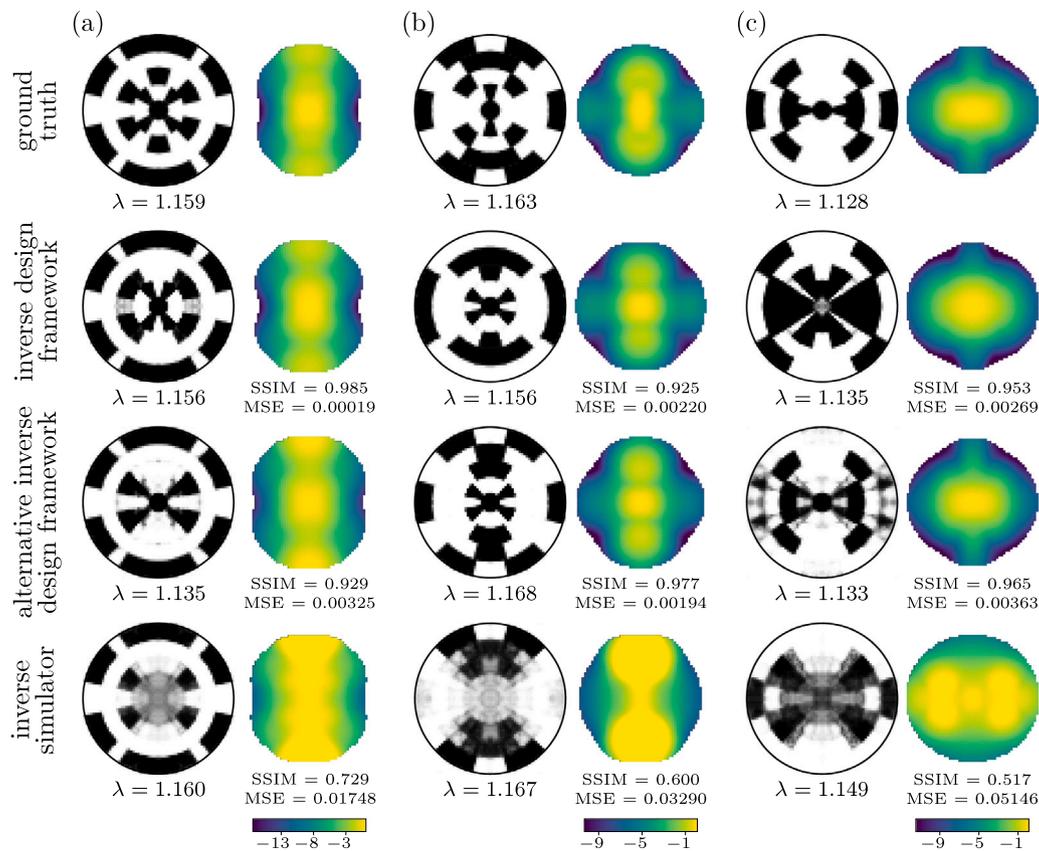
Fig. 9. Comparison of the inverse design of self-deployable kirigami composites with different approaches for three randomly selected samples from the test dataset. The first row shows the ground truths — fabrication parameters from the test dataset and the corresponding shapes, the second row shows the generated fabrication parameters using the generator network from the inverse design framework and FEM results, the third row shows the generated fabrication parameters using the generator network from the alternative inverse design framework and FEM results, the last row shows the predicted fabrication parameters using the inverse simulator network and the corresponding FEM results. The colors represent the z-displacements relative to the plane $z = 0$. The units are in mm.

— the feasibility information (provided by the critic network) and the accuracy information (provided by the pre-trained simulator network). Additionally, it has been shown that the use of a classical FNN is inappropriate in this one-to-many mapping inverse problem.

The method presented addresses the challenges associated with creating complex, self-deployable 3D structures from simple 2D designs. By utilizing the principles of kirigami and strain mismatch combined with the advanced capabilities of GANs, we have demonstrated a significant accuracy, efficiency and diversity of generated fabrication parameters for forming composite structures that deploy into the target shapes. However, limitations exist and they need to be carefully considered when using this approach. The first one is related to the height of the user-defined target shape, or in the other words to the degree of deformation. By selecting different geometric and material properties and by changing the range of the pre-stretch values while preparing the training dataset, one is able to tailor the range of the structure sizes. The second limitation is related to the inability of producing composites with local non-smooth surface curvature in their deformed state, e.g., floppy hat, flower, pyramid and inverted ship hull shapes. This is due to the morphing mechanism where the shapes cannot demonstrate pronounced local changes in the curvature. However, as it can be seen from Fig. 6, we were able to approximate such surfaces quite accurately. Additionally, the integration of a pre-trained simulator network into the GAN framework has proven to be a crucial enhancement, effectively overcoming the limitations of classical FNNs and providing more reliable solutions. These results underline the potential of our approach to revolutionize the design and fabrication of deployable structures in various applications and systems.

CRediT authorship contribution statement

Tomaž Brzin: Writing – original draft, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **M. Khalid Jawed:** Writing – review & editing, Supervision, Methodology, Conceptualization. **Miha Brojan:** Writing – review & editing, Supervision, Resources, Methodology, Investigation, Funding acquisition, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.engappai.2025.110417.
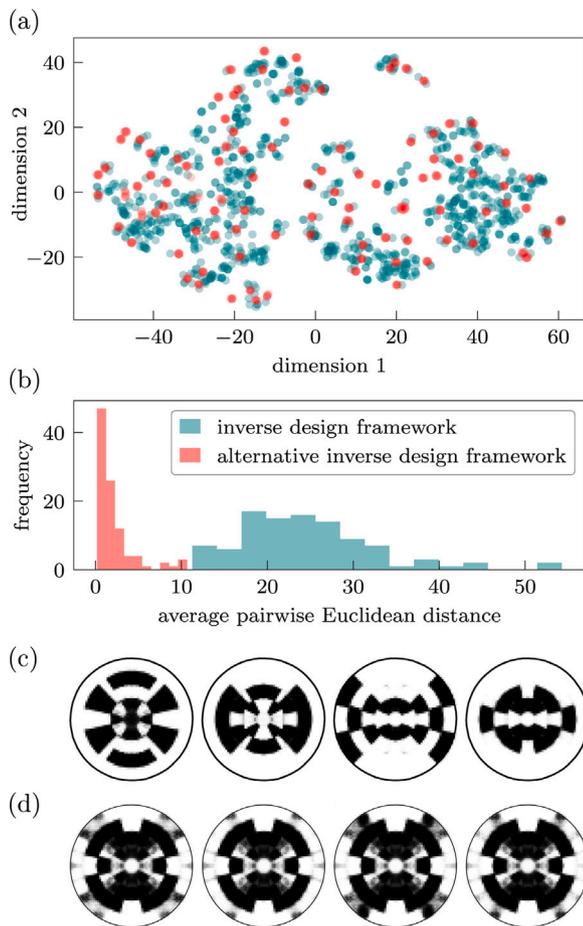
(a)

(b)

(c)

(d)

Fig. 10. Analyzing the inability of the generator network (from the alternative inverse design framework) to generate different kirigami patterns for the same input target shapes. (a) Distribution comparison of the generated kirigami patterns for equal target shapes with the generator networks from the inverse design framework and the alternative inverse design framework. (b) Comparison of the average pairwise Euclidean distances between the kirigami patterns within each group. (c) Kirigami patterns generated with the generator network from the inverse design framework for the same target shape. (d) Kirigami patterns generated by the generator network from the alternative inverse design framework for the same target shape.

Data availability

Data will be made available on request.

References

An, S., Zheng, B., Tang, H., Shalaginov, M.Y., Zhou, L., Li, H., Kang, M., Richardson, K.A., Gu, T., Hu, J., Fowler, C., Zhang, H., 2021. Multifunctional metasurface design with a generative adversarial network. Adv. Opt. Mater. 9 (5), 2001433. http://dx.doi.org/10.1002/adom.202001433.

Arjovsky, M., Chintala, S., Bottou, L., 2017. Wasserstein generative adversarial networks. In: Precup, D., Teh, Y.W. (Eds.), Proceedings of the 34th International Conference on Machine Learning. In: Proceedings of Machine Learning Research, vol. 70, pp. 214–223.

Bletzinger, K.-U., Wüchner, R., Daoud, F., Camprubí, N., 2005. Computational methods for form finding and optimization of shells and membranes. Comput. Methods Appl. Mech. Engrg. 194 (30–33), 3438–3452. http://dx.doi.org/10.1016/j.cma.2004.12.026.

Brooks, A.K., Chakravarty, S., Ali, M., Yadavalli, V.K., 2022. Kirigami-inspired biodesign for applications in healthcare. Adv. Mater. 34, 2109550. http://dx.doi.org/10.1002/adma.202109550.

Brown, N.K., Garland, A.P., Fadel, G.M., Li, G., 2023. Deep reinforcement learning for the rapid on-demand design of mechanical metamaterials with targeted nonlinear deformation responses. Eng. Appl. Artif. Intell. 126, 106998. http://dx.doi.org/10.1016/j.engappai.2023.106998.

Brzin, T., Brojan, M., 2024. Using a generative adversarial network for the inverse design of soft morphing composite beams. Eng. Appl. Artif. Intell. 133, 108527. http://dx.doi.org/10.1016/j.engappai.2024.108527.

Callens, S.J.P., Zadpoor, A.A., 2018. From flat sheets to curved geometries: Origami and kirigami approaches. Mater. Today 21 (3), 241–264. http://dx.doi.org/10.1016/j.mattod.2017.10.004.

Cang, R., Li, H., Yao, H., Jiao, Y., Ren, Y., 2018. Improving direct physical properties prediction of heterogeneous materials from imaging data via convolutional neural network and a morphology-aware generative model. Comput. Mater. Sci. 150, 212–221. http://dx.doi.org/10.1016/j.commatsci.2018.03.074.

Caruso, N., Cvetković, A., Lucantonio, A., Noselli, G., DeSimone, A., 2017. Spontaneous morphing of equibiaxially pre-stretched elastic bilayers: The role of sample geometry. Int. J. Mech. Sci. http://dx.doi.org/10.1016/j.ijmecsci.2017.08.049.

Choi, G.P.T., Dudte, L.H., Mahadevan, L., 2019. Programming shape using kirigami tessellations. Nat. Mater. 18, 999–1004. http://dx.doi.org/10.1038/s41563-019-0452-y.

Dang, X., Feng, F., Plucinsky, P., James, R.D., Duan, H., Wang, J., 2022. Inverse design of deployable origami structures that approximate a general surface. Int. J. Solids Struct. 234, 111224. http://dx.doi.org/10.1016/j.ijsolstr.2021.111224.

Fan, Z., Yang, Y., Zhang, F., Xu, Z., Zhao, H., Wang, T., Song, H., Huang, Y., Rogers, J.A., Zhang, Y., 2020. Inverse design strategies for 3D surfaces formed by mechanical guided assembly. Adv. Mater. 1908424. http://dx.doi.org/10.1002/adma.201908424.

Forte, A.E., Hanakata, P.Z., Jin, L., Zari, E., Zareei, A., Fernandes, M.C., Sumner, L., Alvarez, J., Bertoldi, K., 2022. Inverse design of inflatable soft membranes through machine learning. Adv. Funct. Mater. 2111610. http://dx.doi.org/10.1002/adfm.202111610.

Gladman, A.S., Matsumoto, E.A., Nuzzo, R.G., Mahadevan, L., Lewis, J.A., 2016. Biomimetic 4D printing. Nat. Mater. 15 (4), 413–419. http://dx.doi.org/10.1038/NMAT4544.

Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., Courville, A., 2017. Improved training of Wasserstein GANs. In: Advances in Neural Information Processing Systems. Vol. 30, pp. 5769–5779. http://dx.doi.org/10.48550/arXiv.1704.00028.

Guo, Q., Pan, Y., Lin, J., Wan, G., Xu, B., Hua, N., Zheng, C., Huang, Y., Mei, Y., Chen, W., Chen, Z., 2020. Programmable 3D self-folding structures with strain engineering. Adv. Intell. Syst. 2000101. http://dx.doi.org/10.1002/aisy.202000101.

Guo, K., Yang, Z., Yu, C.-H., Buehler, M.J., 2021. Artificial intelligence and machine learning in design of mechanical materials. Mater. Horizons 8, 1153–1172. http://dx.doi.org/10.1039/d0mh01451f.

Guseinov, R., McMahan, C., Pérez, J., Daraio, C., Bickel, B., 2020. Programming temporal morphing of self-actuated shells. Nat. Commun. 11 (237), http://dx.doi.org/10.1038/s41467-019-14015-2.

Holmes, D.P., 2019. Elasticity and stability of shape-shifting structures. Curr. Opin. Colloid Interface Sci. 40, 118–137. http://dx.doi.org/10.1016/j.cocis.2019.02.008.

Jin, T., Cheng, X., Xu, S., Lai, Y., Zhang, Y., 2023. Deep learning aided inverse design of the buckling-guided assembly for 3D frame structures. J. Mech. Adn Phys. Solids 179, 105398. http://dx.doi.org/10.1016/j.jmps.2023.105398.

Jin, L., Forte, A.E., Deng, B., Rafsanjani, A., Bertoldi, K., 2020. Kirigami-inspired inflatables with programmable shapes. Adv. Mater. 2001863. http://dx.doi.org/10.1002/adma.202001863.

Kansara, H., Liu, M., He, Y., Tan, W., 2023. Inverse design and additive manufacturing of shape-morphing structures based on functionally graded composites. J. Mech. Phys. Solids 180, 105382. http://dx.doi.org/10.1016/j.jmps.2023.105382.

Kim, W., Kim, S., Lee, M., Seok, J., 2022. Inverse design of nanophotonic devices using generative adversarial networks. Eng. Appl. Artif. Intell. 115, 105259. http://dx.doi.org/10.1016/j.engappai.2022.105259.

Kim, B., Lee, S., Kim, J., 2020. Inverse design of porous materials using artificial neural networks. Sci. Adv. 6 (1), eaax9324. http://dx.doi.org/10.1126/sciadv.aax9324.

Koohestani, K., 2012. Form-finding of tensegrity structures via genetic algorithm. Int. J. Solids Struct. 49 (5), 739–747. http://dx.doi.org/10.1016/j.ijsolstr.2011.11.015.

Lee, K.-H., Lim, H.J., Yun, G.J., 2024. A data-driven framework for designing microstructure of multifunctional composites with deep-learned diffusion-based generative models. Eng. Appl. Artif. Intell. 129, 107590. http://dx.doi.org/10.1016/j.engappai.2023.107590.

Liu, Z., Zhu, D., Rodrigues, S.P., Lee, K.-T., Cai, W., 2018. Generative model for the inverse design of metasurfaces. Nano Lett. 18 (10), 6570–6576. http://dx.doi.org/10.1021/acs.nanolett.8b03171.

Ma, L., Mungekar, M., Roychowdhury, V., Jawed, M.K., 2024. Rapid design of fully soft deployable structures via kirigami cuts and active learning. Adv. Mater. Technol. 9 (5), 2301305. http://dx.doi.org/10.1002/admt.202301305.

Mao, Y., He, Q., Zhao, X., 2020. Designing complex architectured materials with generative adversarial networks. Sci. Adv. 17 (6), eaaz4169. http://dx.doi.org/10.1126/sciadv.aaz4169.

Mirza, M., Osindero, S., 2014. Conditional generative adversarial nets. URL: https://arxiv.org/abs/1411.1784.

Mungekar, M., Ma, L., Yan, W., Kackar, V., Shohrazadeh, S., Jawed, M.K., 2023. Design of bistable soft deployable structures via a kirigami-inspired planar fabrication approach. Adv. Mater. Technol. 8 (16), 2300088. http://dx.doi.org/10.1002/admt.202300088.

Nie, Z., Lin, T., Jiang, H., Kara, L.B., 2021. TopologyGAN: Topology optimization using generative adversarial networks based on physical fields over the initial domain. J. Mech. Des. 143 (3), 1–13. http://dx.doi.org/10.1115/1.4049533.

Nojoomi, A., Arslan, H., Lee, K., Yum, K., 2018. Bioinspired 3D structures with programmable morphologies and motions. Nat. Commun. 9, 3705. http://dx.doi.org/10.1038/s41467-018-05569-8.

Nojoomi, A., Jeon, J., Yum, K., 2021. 2D material programming for 3D shaping. Nat. Commun. 12 (1), 603. http://dx.doi.org/10.1038/s41467-021-20934-w.

Oh, S., Jung, Y., Kim, S., Lee, I., Kang, N., 2019. Deep generative design: Integration of topology optimization and generative models. J. Mech. Des. 141 (11), 111405. http://dx.doi.org/10.1115/1.4044229.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, É., 2011. Scikit-learn: Machine learning in python. J. Mach. Learn. Res. 12 (85), 2825–2830.

Petzka, H., Fischer, A., Lukovnikov, D., 2018. On the regularization of Wasserstein GANs. URL: https://arxiv.org/abs/1709.08894.

Pezzulla, M., Shillig, S.A., Nardinocchi, P., Holmes, D.P., 2015. Morphing of geometric composites via residual swelling. Soft Matter 11 (29), 5812–5820. http://dx.doi.org/10.1039/c5sm00863h.

Regenwetter, L., Nobari, A.H., Ahmed, F., 2022. Deep generative models in engineering: A review. J. Mech. Des. 144 (7), 071704. http://dx.doi.org/10.1115/1.4053859.

Rus, D., Tolley, M.T., 2015. Design, fabrication and control of soft robots. Nat. 521 (7553), 467–475. http://dx.doi.org/10.1038/nature14543.

Schenk, M., Guest, S.D., 2013. Geometry of Miura-folded metamaterials. Proc. Natl. Acad. Sci. USA 110 (9), 3276–3281. http://dx.doi.org/10.1073/pnas.1217998110.

Siéfert, E., Reyssat, E., Bico, J., Roman, B., 2019. Bio-inspired pneumatic shape-morphing elastomers. Nat. Mater. 18 (1), 24–29. http://dx.doi.org/10.1038/s41563-018-0219-x.

Simonyan, K., Zisserman, A., 2015. Very deep convolutional networks for large-scale image recognition. URL: https://arxiv.org/abs/1409.1556.

Su, Y., Ohsaki, M., Wu, Y., Zhang, J., 2019. A numerical method for form finding and shape optimization of reciprocal structures. Eng. Struct. 198, 109510. http://dx.doi.org/10.1016/j.engstruct.2019.109510.

van der Maanen, L., Hinton, G., 2008. Visualizing data using t-SNE. J. Mach. Learn. Res. 9, 2579–2605.

van Manen, T., Janbaz, S., Zadpoor, A.A., 2018. Programming the shape-shifting of flat soft matter. Mater. Today 21 (2), 144–163. http://dx.doi.org/10.1016/j.mattod.2017.08.026.

van Rees, W.M., Vouga, E., Mahadevan, L., 2017. Growth patterns for shape-shifting elastic bilayers. Proc. Natl. Acad. Sci. USA 114 (44), 11597–11602. http://dx.doi.org/10.1073/pnas.1709025114.

Wang, H.P., Li, Y.B., Li, H., Dong, S.Y., Liu, C., Jin, S., Cui, T.J., 2020. Deep learning designs of anisotropic metasurfaces in ultrawideband based on generative adversarial networks. Adv. Intell. Syst. 2 (9), 2000068. http://dx.doi.org/10.1002/aisy.202000068.

Wang, J., Zhang, B., Lei, P., Dai, N., 2019. Photo-induced bending and buckling of polymer sheets. In: IOP Conference Series - Materials Science and Engineering. Vol. 493, 012029. http://dx.doi.org/10.1088/1757-899X/493/1/012029.

Xu, Z., Fan, Z., Fu, H., Liu, Y., Zi, Y., Huang, Y., Zhang, Y., 2019. Optimization-based approach for the inverse design of ribbon-shaped three-dimensional structures assembled through compressive buckling. Phys. Rev. Appl. 11 (5), 054053. http://dx.doi.org/10.1103/PhysRevApplied.11.054053.

Xu, S., Yan, Z., Jang, K.-I., Huang, W., Fu, H., Kim, J., Wei, Z., Flavin, M., McCracken, J., Wang, R., Badea, A., Liu, Y., Xiao, D., Zhou, G., Lee, J., Chung, H.U., Cheng, H., Ren, W., Banks, A., Li, X., Paik, U., Nuzzo, R.G., Huang, Y., Zhang, Y., Rogers, J.A., 2015. Assembly of micro/nanomaterials into complex, three-dimensional architectures by compressive buckling. Mater. Sci. 347 (6218), 154–159. http://dx.doi.org/10.1126/science.1260960.

Xue, R., Li, R., Du, Z., Zhang, W., Zhu, Y., Sun, Z., Guo, X., 2017. Kirigami pattern design of mechanically driven formation of complex 3D structures through topology optimization. Extrem. Mech. Lett. 15, 139–144. http://dx.doi.org/10.1016/j.eml.2017.03.004.

Yao, Z., Sànchez-Lengeling, B., Bobbitt, N.S., Bucior, B.J., Kumar, S.G.H., Collins, S.P., Burns, T., Woo, T.K., Farha, O.K., Snurr, R.Q., Aspuru-Guzik, A., 2021. Inverse design of nanoporous crystalline reticular materials with deep generative models. Nat. Mach. Intell. 3 (1), 76–86. http://dx.doi.org/10.1038/s42256-020-00271-1.

Yeung, C., Tsai, R., Pham, B., King, B., Kawagoe, Y., Ho, D., Liang, J., Knight, M.W., Raman, A.P., 2021. Global inverse design across multiple photonic structure classes using generative deep learning. Adv. Opt. Mater. 9 (20), 2100548. http://dx.doi.org/10.1002/adom.202100548.

Yu, Y., Hur, T., Jung, J., Jang, I.G., 2019. Deep learning for determining a near-optimal topological design without any iteration. Struct. Multidiscip. Optim. 59 (3), 787–799. http://dx.doi.org/10.1007/s00158-018-2101-5.

Zavodnik, J., Wang, Y., Yan, W., Brojan, M., Jawed, M.K., 2024. Soft kirigami composites for form-finding of fully flexible deployables. Adv. Mater. Technol. 9 (1), http://dx.doi.org/10.1002/admt.202300909.

Zhang, Y., Yang, J., Liu, M., Vella, D., 2022. Shape-morphing structures based on perforated kirigami. Extrem. Mech. Lett. 56, 101857. http://dx.doi.org/10.1016/j.eml.2022.101857.