Check for updates

Advanced Materials

ADVANCED MATERIALS
www.advmat.de

RESEARCH ARTICLE OPEN ACCESS

Learning to Program Dimple-Encoded Strips for Morphing Architectures

Qun Zhang[1] | Omid Doustdar[1] | Hyunyoung Kim[2] | Claire Dancer[3] | Zhaohe Dai[4] | Weicheng Huang[5] | Karl Dearn[1] | Dominic Vella[6] | Mingchao Liu[1]

[1]Department of Mechanical Engineering, University of Birmingham, Birmingham, UK | [2]School of Computer Science, University of Birmingham, Birmingham, UK | [3]School of Metallurgy and Materials, University of Birmingham, Birmingham, UK | [4]School of Mechanics and Engineering Science, State Key Laboratory for Turbulence and Complex Systems, Peking University, Beijing, China | [5]School of Engineering, Newcastle University, Newcastle upon Tyne, UK | [6]Mathematical Institute, University of Oxford, Oxford, United Kingdom

Correspondence: Weicheng Huang (weicheng.huang@newcastle.ac.uk) | Karl Dearn (k.d.dearn@bham.ac.uk) | Dominic Vella (dominic.vella@maths.ox.ac.uk) | Mingchao Liu (m.liu.2@bham.ac.uk)

Received: 25 May 2026 | **Revised:** 23 October 2026 | **Accepted:** 4 November 2026

Keywords: bistability | dimple-encoded strips | learning-enabled inverse design | mechanical metamaterials | programmable morphing

ABSTRACT

Programmable morphing structures that translate local mechanical states into global shape transformations enable deployable, adaptive, and reconfigurable systems. Dimple-encoded elastic strips provide a minimal platform in which arrays of bistable dimples can collectively program large bending deformation. Yet the nonlinear mapping from the encoded geometry to the final configuration makes inverse design challenging. Here, we report a learning-enabled strategy for programming dimple-encoded strips with prescribed morphing profiles and using them as building blocks for higher-level morphing architectures. Finite-element simulations are converted into centerline-based tangent-field representations to train a one-dimensional convolutional neural-network surrogate that maps local dimple encoding and strip geometry to global shape. The trained surrogate enables gradient-based inverse design of target morphologies, with optimized designs validated by independent simulations and physical prototypes. Variable-width strip modules expand the accessible shape space and enable width-conditioned curvature programming. Rotational tessellation of programmed strip modules creates dome-like morphing shells, while hierarchical integration of curvature-programmed strips with folding modules yields an artifical flower exhibiting sequential opening through dimple switching. These results establish dimple-encoded strips as learnable building blocks that connect local bistable encoding, inverse design, and hierarchical shape formation, providing a route toward task-specific morphing components for soft robotics, adaptive interfaces, and deployable structures.

1 | Introduction

Morphing thin-sheet structures capable of transforming from simple planar states into complex three-dimensional (3D) architectures offer a powerful route toward adaptive robotics, deployable devices, and reconfigurable structural systems [1–5]. In these systems, geometry can act as an embedded mechanical program: local structural features prescribe deformation pathways and thereby steer large-scale shape transformation without conventional hinges or discrete mechanical assemblies. Strategies based on patterned thickness, cuts, residual stress, and architected surface features have enabled a broad range of programmable shape-morphing behaviors [6–10] and mechanically encoded functionalities [11–13]. Geometric patterning can also be combined with active materials such as liquid-crystal elastomers and liquid-crystal gels: programmed molecular orientations or

This is an open access article under the terms of the Creative Commons Attribution License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.
© 2026 The Author(s). *Advanced Materials* published by Wiley-VCH GmbH

director fields can be coupled with geometric features to realize complex, reversible shape transformations and locomotion [14–16]. Together, these approaches illustrate the diverse ways in which local geometric and material programming can be translated into global deformation and function.

Among these approaches, geometrically embedded bistable features are particularly attractive because they introduce distributed multistability and localized deformation pathways while remaining compatible with lightweight, monolithic fabrication [17–21]. By embedding bistable units within thin elastic sheets, complex global morphologies can emerge from collective local deformations, enabling mechanically programmed shape transformation and deployable functionalities [22, 23]. The mechanically persistent local states provided by bistability further create opportunities for reconfigurable and deployable structures, in which global configurations can be programmed through changes in local mechanical states without altering the fabricated architecture.

Dimple-embedded elastic sheets represent an experimentally accessible realization of geometrically embedded bistability and provide a versatile platform for programmable morphing [25–27]. In these systems, discrete bistable dimples collectively regulate global bending, curvature localization, and shape transformation within a continuous elastic body. Through the spatial arrangement of local dimples, diverse bending-dominated morphologies and multistable configurations can emerge from the same initially flat sheet. However, the resulting morphing behavior is governed by strongly nonlinear interactions between neighboring dimples, geometric constraints, and large-deformation elasticity, making predictive control of global configuration highly challenging [28, 29]. Here, we focus on dimple-encoded elastic strips—each comprising an array of interacting bistable dimples embedded in an elastic strip, with the dimple locations, spacings, and heights treated as programmable design variables that encode the desired morphing response. These strips provide a minimal yet expressive platform that retains the essential nonlinear interactions among neighboring dimples while allowing the dominant global deformation to be described through its centerline, making systematic geometric programming and inverse design tractable. This modularity allows the strip-level design strategy to be extended naturally toward higher-dimensional morphing architectures [6].

Even in this reduced-dimensional setting, predicting the morphing response of dimple-encoded strips remains challenging because the global morphology emerges from strongly nonlinear interactions among neighboring dimples. The inclusion of strip geometry as an additional design variable further enlarges the accessible design space, making exhaustive parameter exploration increasingly inefficient. Learning-enabled surrogate models offer a promising route to replace repeated finite-element method (FEM) exploration with a reusable forward mapping between geometric encoding and deformed configuration [30–33]. Once trained, such surrogates can enable rapid inverse design by identifying local dimple encodings and strip geometries that match prescribed target shapes. However, stable inverse morphing requires representations that remain robust under large deformations, geometric variability, and smooth target descriptions. The underlying strip geometry provides an additional source of variability, with changes in strip width further modulating the local-to-global deformation pathway [34, 35]. Consequently, the inverse mapping need not be unique: distinct combinations of dimple encoding and strip geometry can generate comparable global configurations. These challenges call for geometry-aware learning strategies that can capture nonlinear morphing behavior while remaining compatible with scalable inverse design [36, 37].

Here, we introduce a geometry-aware, learning-enabled forward–inverse framework for programming dimple-encoded strips with prescribed morphing. Discrete dimple codings and strip geometries are embedded into continuous one-dimensional (1D) representations, enabling compact learning of the nonlinear relation between local bistable encoding and global shape formation. To improve robustness and support stable inverse optimization, deformed strip configurations are represented by smoothed tangent fields extracted from FEM simulations. The resulting 1D convolutional neural-network (CNN) surrogate enables rapid inverse design from target curves and image-derived geometries while remaining compatible with varying strip geometries and assembly constraints [38, 39]. The central advance lies in establishing a geometry-conditioned inverse-design strategy that connects physically realizable local dimple encodings to prescribed global morphing profiles.

Building on this framework, we demonstrate target-guided programmable morphing across constant- and variable-width strips, shell morphogenesis through tessellated strip assemblies, and multilayer hierarchical folding–morphing architectures inspired by flower opening. These capabilities suggest potential applications in reconfigurable and deployable devices, adaptive structures, and soft robotics in which global shape is programmed through local mechanical encoding. Together, these results establish dimple-encoded strips as learnable building blocks for programmable morphing architectures, bridging local bistable encoding, inverse design, and hierarchical three-dimensional shape formation.

2 | Results and Discussion

2.1 | Learning-Enabled Forward–Inverse Framework

We first establish a learning-enabled forward–inverse framework that connects local dimple encoding to global strip morphology (Figure 1). In the forward stage, FEM simulations generate deformed morphing configurations of dimple-encoded strips, which are converted into centerline-based tangent-field representations for surrogate-model training. Unless otherwise stated, all bistable dimples in a prescribed design are assumed to be actuated to the snapped-through state. In the experiments, each dimple is switched manually by pressing its apex until snap-through; owing to its bistability, the inverted state is retained after the applied force is removed. The present framework therefore optimizes the geometry of a fully activated dimple pattern, while selective activation of different subsets of dimples could be incorporated as an additional discrete state variable in future extensions. In the inverse stage, a target curve is represented using the same tangent-field description, allowing the design variables to be optimized by back-propagating the mismatch through the

15214095, 0, Downloaded from https://advanced.onlinelibrary.wiley.com/doi/10.1002/adma.75796, Wiley Online Library on [24/11/2026]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

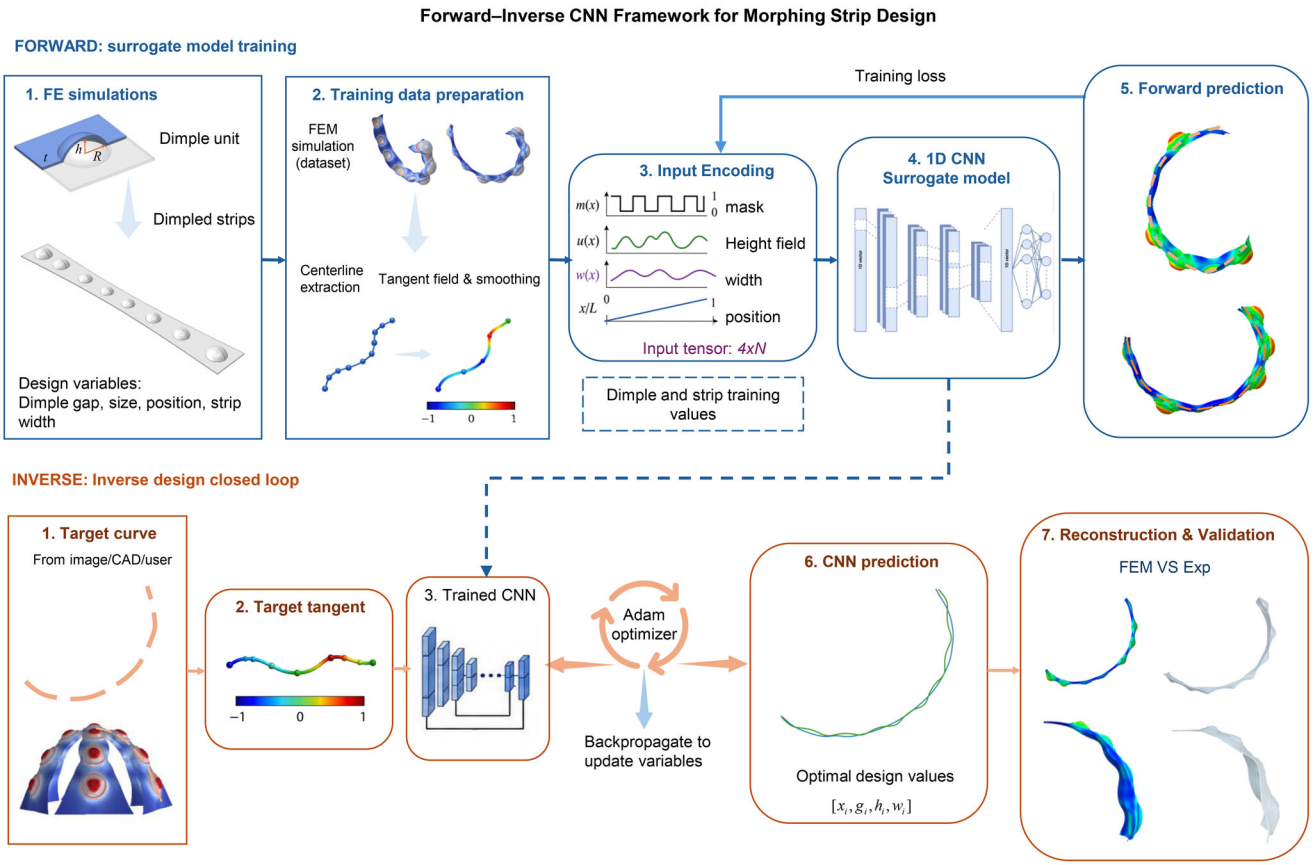


FIGURE 1 | Learning-enabled forward–inverse framework for dimple-encoded strip design. In the forward stage, FEM-simulated deformed strips are converted into centerline-based tangent fields and encoded with the strip and dimple design parameters to train a 1D CNN surrogate. In the inverse stage, a target curve is mapped to the same tangent representation, and the design variables are optimized by back-propagating the loss through the trained surrogate. The optimized designs are then validated by FEM simulations and experiments.

trained surrogate. Further implementation details are provided in Note S1. This closed-loop formulation provides a differentiable route from prescribed global shape to the geometric programming of local dimples.

The training dataset is constructed from both constant-width and variable-width dimple-encoded strips (Figure 2). Constant-width strips are used as a controlled baseline in which the planform is fixed, allowing the influence of dimple spacing and height on the resulting morphology to be examined independently of variations in strip width. We choose $N = 8$ dimples as a practical compromise between the number of programmable geometric degrees of freedom and the computational and experimental cost of dataset generation, fabrication, and validation. In the constant-width case, each strip has fixed length L and width $w = C$, with N bistable dimples distributed along the centerline (Figure 2a). A design is specified by the inter-dimple gaps $\{g_i\}$ and dimple heights $\{h_i\}$, while the dimple radius R and edge clearance are fixed (Note S2).

To introduce geometry-dependent morphing responses, we further consider variable-width strip families, including barrel, trapezoid, and hourglass profiles (Figure 2b). The local width is prescribed along the normalized arc length $s \in [0, 1]$ as

$$w(s) = w_{\min} + \Delta w\, \phi(s), \qquad \Delta w = w_{\min}(\Gamma_w - 1) \tag{1}$$

where $w_{\min}$ is the minimum strip width, $\Gamma_w = w_{\max}/w_{\min}$ is the width ratio, and $\phi(s) \in [0, 1]$ defines the width-profile family. For a dimple centered at s_i, local geometric compatibility requires

$$R_i + e \le \frac{w(s_i)}{2} \tag{2}$$

where R_i is the dimple radius and e is the edge-clearance margin. Together with non-overlap and end-clearance constraints, this condition ensures that all sampled designs are compatible with the local strip geometry (Note S1).

The variable-width families extend this controlled baseline in two ways. First, they allow the surrogate to learn how variations in local planform width, together with the associated dimple geometry, modify the local-to-global morphing response. Second, they establish the width-conditioned design framework required later for the tapered strip modules used in the rotational tessellation of the dome-like shell in Section 2.3. Figure 2 therefore defines a geometry-conditioned training domain that supports both strip-level inverse design and the subsequent extension to strip-based 3D architectures.

For each feasible design, FEM simulations are performed to obtain the relaxed configuration after dimple snap-through (Figure 2c,d). The deformed centerline is extracted and

15214095, 0, Downloaded from https://advanced.onlinelibrary.wiley.com/doi/10.1002/adma.75796, Wiley Online Library on [24/11/2026]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

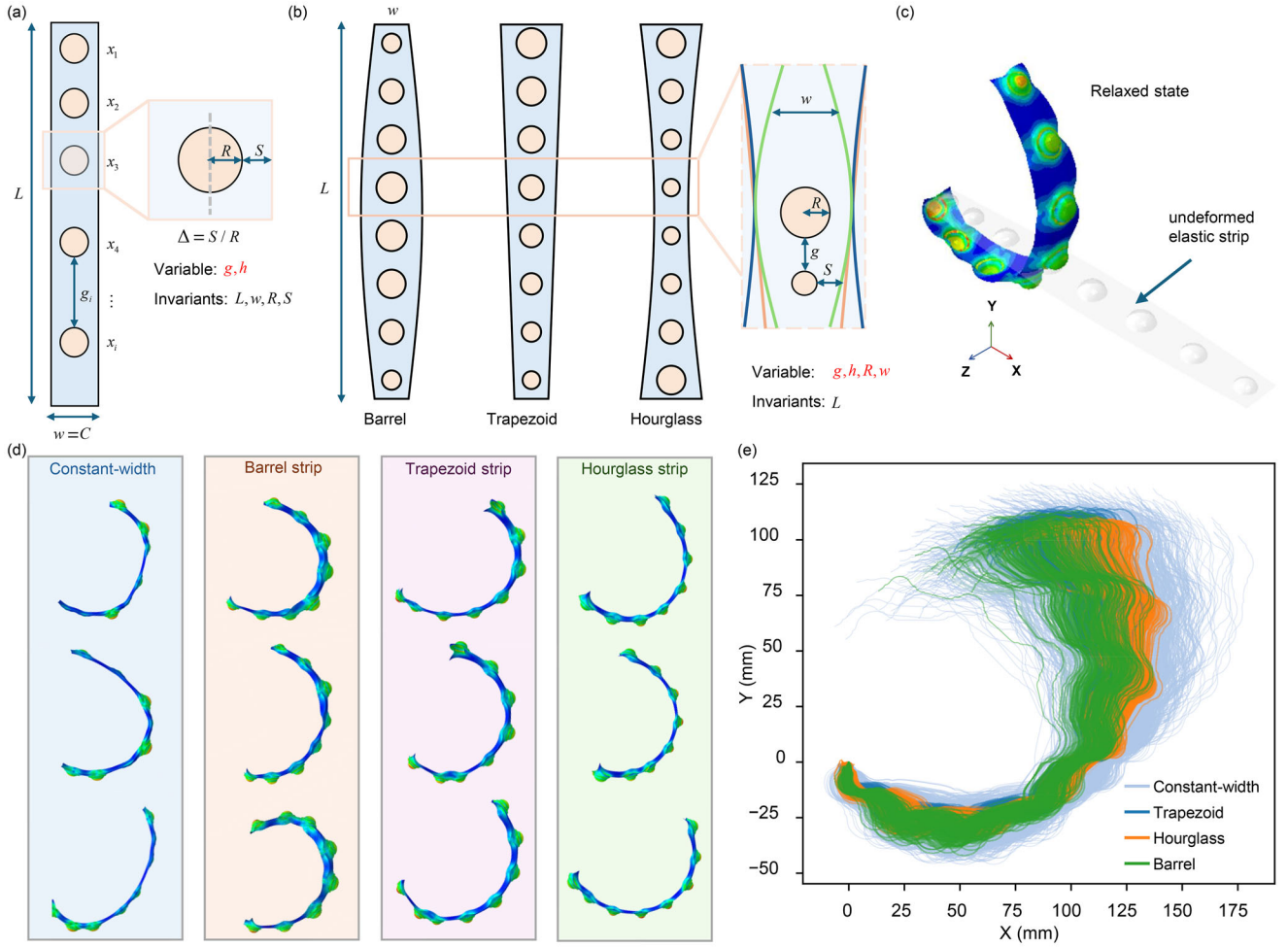


FIGURE 2 | Dimple-encoded strip parameterization and FE-generated morphing dataset. (a) Constant-width strip with N bistable dimples; the inter-dimple gaps $\{g_i\}$ and dimple heights $\{h_i\}$ are varied while L, w, R, and S remain fixed. (b) Variable-width strip families, including barrel, trapezoid, and hourglass profiles, where the local width $w(s)$ defines the available dimple size and placement along the strip. (c) Representative FEM simulation of a deformed strip after dimple snap-through. (d) Example deformed configurations from constant- and variable-width strips. (e) Extracted centerlines from the full FEM dataset used for surrogate-model training. All strips have length $L = 250$ mm and dimple count $N = 8$. For constant-width strips, the width is uniform ($w = 24$ mm) and the dimple radius is fixed at $R = 8$ mm. For variable-width strips, $w_{\min} = 22$ mm and the dimple radius is coupled to the local width through $R/(w/2) = 0.5$, giving $R \approx 5$–10 mm depending on the local width. The inter-dimple gaps vary within $g_i \in [6, 20]$ mm, with end dimples placed subject to fixed edge-clearance constraints. The variable-width families establish the geometry-conditioned design space required for the tapered gore modules used in the subsequent shell tessellation.

resampled at fixed resolution along normalized arc length as $\mathbf{r}(s) = (X(s), Y(s))$. Instead of learning absolute coordinates, which are sensitive to rigid-body motion, we represent each configuration using the unit tangent field (Note S3)

$$\mathbf{t}(s) = \frac{\partial \mathbf{r}/\partial s}{\left\|\partial \mathbf{r}/\partial s\right\|} = \left(t_x(s), t_y(s)\right) \tag{3}$$

The centerline is resampled at $M = 256$ equally spaced points in normalized arc length to provide a common output resolution across all configurations. The tangent representation removes sensitivity to rigid translation while retaining the orientation changes associated with global bending. The raw tangent field is then smoothed by a 1D Gaussian convolution and re-normalized pointwise,

$$\tilde{\mathbf{t}}(s_k) = \sum_{q=0}^{M-1} K_\sigma(s_k - s_q)\mathbf{t}(s_q), \qquad \bar{\mathbf{t}}(s_k) = \frac{\tilde{\mathbf{t}}(s_k)}{\|\tilde{\mathbf{t}}(s_k)\|} \tag{4}$$

where K_σ is a normalized Gaussian kernel. The smoothing suppresses dimple-scale local imprints while preserving the large-scale bending profile targeted by the inverse-design process. The same resampling and smoothing operations are applied to both FEM-generated configurations and target curves, ensuring a consistent shape representation for training and inverse design. The extracted centerlines from the FEM dataset summarize the morphology space used for learning (Figure 2e).

The corresponding design inputs are embedded into continuous one-dimensional fields on a uniform axial grid $x \in [0, L]$.

15214095, 0, Downloaded from https://advanced.onlinelibrary.wiley.com/doi/10.1002/adma.75796, Wiley Online Library on [24/11/2026]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

Gaussian basis functions centered at each dimple location are used to construct a smooth occupancy mask $m(x)$ and a height-weighted dimple channel $u(x)$,

$$G_i(x) = \exp\left[-\frac{(x - x_i)^2}{2\sigma^2}\right], \qquad m(x) = 1 - \exp\left(-\sum_{i=1}^{N} G_i(x)\right),$$
$$u(x) = \sum_{i=1}^{N} h_i\, G_i(x) \tag{5}$$

where σ controls the width of the Gaussian basis functions, $m(x)$ encodes the spatial distribution of dimples, while $u(x)$ encodes their local height distribution. To provide the convolutional network with information about position along the finite strip, we additionally introduce a normalized coordinate channel $c(x)$. For variable-width strips, a normalized width channel $\tilde{w}(x)$ is used to encode the local planform geometry,

$$c(x) = \frac{x}{L}, \qquad \tilde{w}(x) = \frac{w(x)}{\langle w \rangle} \tag{6}$$

where $\langle w \rangle$ denotes the mean width of the strip. Giving the network inputs

$$\mathbf{X}_{\mathrm{cw}}(x) = \left[m(x), u(x), c(x)\right], \qquad \mathbf{X}_{\mathrm{vw}}(x) = \left[m(x), u(x), c(x), \tilde{w}(x)\right] \tag{7}$$

The constant- and variable-width datasets therefore use the same field-based encoding strategy and the same surrogate architecture, with the local width supplied explicitly as an additional conditioning channel for the variable-width cases (Table S1, Notes S1–S2).

A compact 1D convolutional neural-network (CNN) surrogate is trained to map the encoded design fields to the smoothed tangent representation. We use a dilated residual 1D CNN, whose enlarged receptive field enables the model to capture nonlocal interactions between spatially separated dimples while retaining the one-dimensional structure of the encoded geometry. The network maps the encoded input field to a two-component tangent field sampled at $M = 256$ points,

$$\hat{\mathbf{t}}(s) = f_\theta(\mathbf{X}), \qquad \hat{\mathbf{t}} \in \mathbb{R}^{2\times M} \tag{8}$$

The surrogate is trained directly against the smoothed FEM-derived tangent field. To preserve both pointwise orientation and the large-scale bending profile, the forward loss combines tangent error, tangent-gradient error, low-pass shape error, and a unit-norm regularization term,

$$\mathcal{L}_{\mathrm{fwd}} = \left\|\hat{\mathbf{t}} - \bar{\mathbf{t}}\right\|_2^2 + \lambda_\nabla \left\|\nabla_s \hat{\mathbf{t}} - \nabla_s \bar{\mathbf{t}}\right\|_2^2 + \lambda_{\mathrm{lp}} \left\|\mathcal{P}(\hat{\mathbf{t}}) - \mathcal{P}(\bar{\mathbf{t}})\right\|_2^2 + \lambda_{\mathrm{unit}} \left\| \left\|\hat{\mathbf{t}}\right\|_2 - 1 \right\|_2^2 \tag{9}$$

where ∇_s denotes the discrete derivative along normalized arc length and $\mathcal{P}$ is a local low-pass averaging operator. The four terms respectively promote pointwise tangent agreement, consistency of local tangent variation, preservation of the large-scale bending profile, and near-unit tangent magnitude. The surrogate is trained on 3000 accepted FEM-generated configurations. Network architecture and training details are provided in Note S4, with the training configuration summarized in Table S2, convergence in Figure S1, and representative held-out forward predictions in Figure S2.

Once trained, this surrogate serves not only as a fast forward predictor but also as the differentiable core of the inverse-design loop (Figure 1). Given a target curve, the target tangent field $\bar{\mathbf{t}}^*(s)$ is computed using the same arc-length resampling, tangent extraction, Gaussian smoothing, and re-normalization pipeline used for the FEM data.

During inverse design, the trained network parameters θ are frozen and only the physical design variables are updated. For a constant-width strip, the continuous inverse variables are $\mathbf{p} = \{g_i, h_i\}$; for a variable-width strip, the prescribed width profile is supplied as an additional geometric condition. The current design is encoded as $\mathbf{X}(\mathbf{p})$ and propagated through the fixed surrogate,

$$\hat{\mathbf{t}}(s; \mathbf{p}) = f_\theta[\mathbf{X}(\mathbf{p})] \tag{10}$$

The inverse loss is evaluated in the same tangent space used for forward training,

$$\mathcal{L}_{\mathrm{inv}} = W_{\mathrm{point}} \left\|\hat{\mathbf{t}} - \bar{\mathbf{t}}^*\right\|_2^2 + W_{\mathrm{low}} \left\|\mathcal{P}(\hat{\mathbf{t}}) - \mathcal{P}(\bar{\mathbf{t}}^*)\right\|_2^2 + W_{\mathrm{grad}} \left\|\nabla_s \hat{\mathbf{t}} - \nabla_s \bar{\mathbf{t}}^*\right\|_2^2 + \mathcal{L}_{\mathrm{con}} + \mathcal{L}_{\mathrm{reg}}$$

where $\mathcal{L}_{\mathrm{con}}$ enforces the prescribed strip length, admissible dimple-height and gap ranges, non-overlap, edge clearance, and, for variable-width strips, local geometric compatibility. The regularization term $\mathcal{L}_{\mathrm{reg}}$ suppresses excessively oscillatory dimple configurations. The optimized design is therefore

$$\mathbf{p}^* = \arg\min_{\mathbf{p}\in\mathcal{P}} \mathcal{L}_{\mathrm{inv}}(\mathbf{p}) \tag{12}$$

with $\mathcal{P}$ denoting the feasible design domain. Bounded variables are represented by differentiable mappings during optimization, while the fixed-length gap constraint is enforced by differentiable rescaling. Because the inverse objective is non-convex, 40 feasible multi-start initializations are used to explore distinct minima. The lowest-loss design is rounded to the 0.5– mm fabrication grid, projected onto the feasible set, re-encoded, and evaluated again by the surrogate. Representative inverse-design diagnostics and quantization effects are shown in Figure S4. The selected design is then verified independently by FEM and, for experimental demonstrations, by physical prototypes (Note S5).

The explicit separation between offline surrogate training and inverse optimization allows a single trained model to be reused for multiple target shapes. Retraining is not required when the target shape or design parameters vary within the geometric and material ranges represented in the training data. In particular, the width profile is supplied explicitly as a conditioning input, allowing the surrogate to interpolate across the constant-width and sampled variable-width families. Additional training or fine-tuning is required only when extending substantially beyond the trained domain, for example to different structural topologies, dimple numbers, material models, loading conditions, or geometric ranges. This unified representation allows the same forward–inverse framework to program constant-width

15214095, 0, Downloaded from https://advanced.onlinelibrary.wiley.com/doi/10.1002/adma.75796, Wiley Online Library on [24/11/2026]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

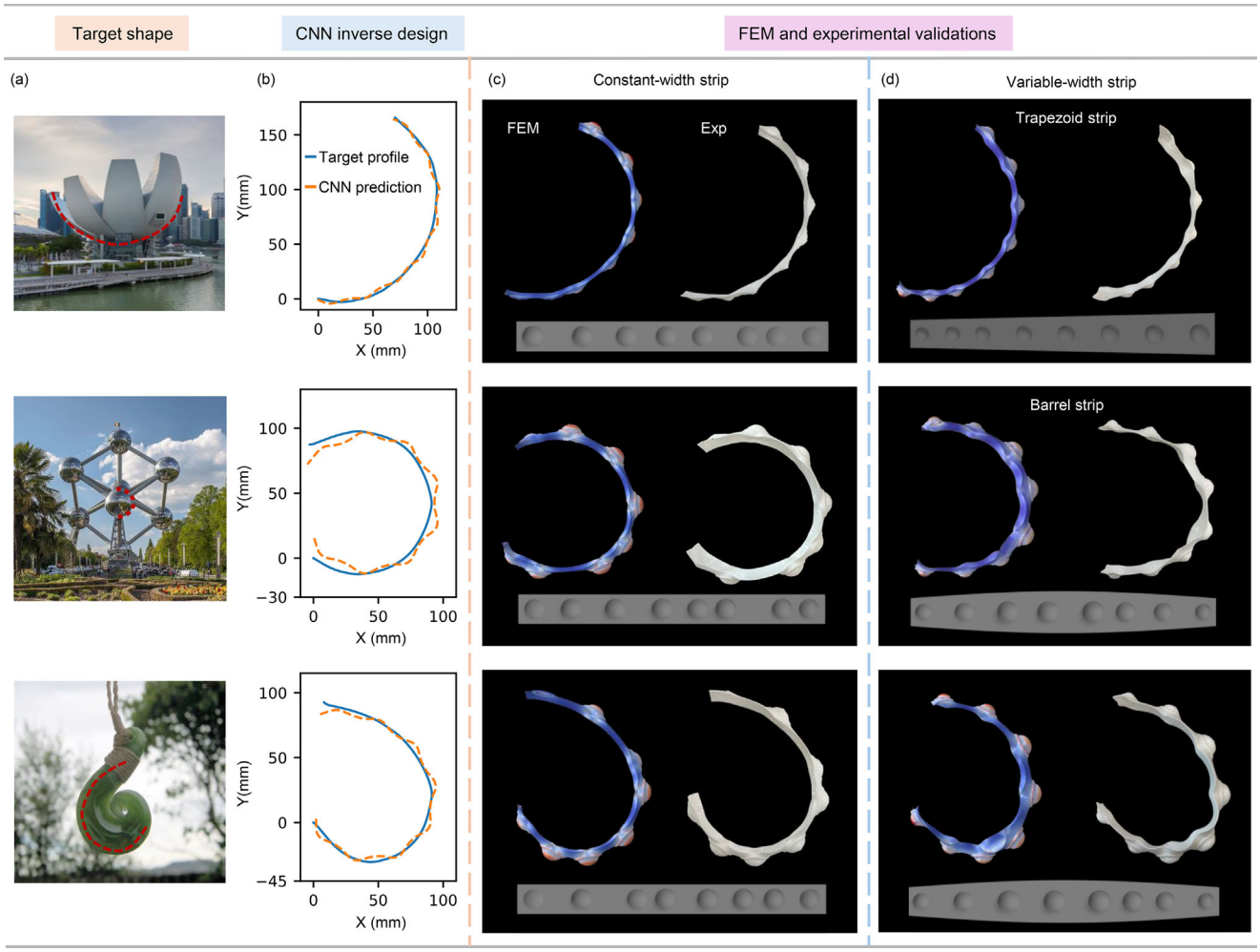


FIGURE 3 | Target-guided programmable morphing of dimple-encoded strips. (a) Representative target shapes with different curvature distributions used to define the desired morphing profiles. (b) Target profiles and CNN-predicted curves obtained through inverse design. (c) FEM and experimental validation of the optimized constant-width strip designs. (d) FEM and experimental validation of optimized variable-width strip designs, including trapezoid and barrel geometries. The examples demonstrate target-guided shape programming within the morphology range accessible to the present dimple and strip design space. Photographs in (a), from top to bottom: ArtScience Museum by Basile Morin (CC BY-SA 4.0), Atomium by Marek Śliwecki (CC BY-SA 4.0), and Pounamu pendant by Andrew Sullivan (CC BY-SA 3.0). All images were adapted by adding red dashed guide curves. Full attribution and license details are provided in the Image Credits and Permissions section.

strips, variable-width strips, and strip modules whose geometries are later constrained by tessellation or hierarchical assembly. The following section validates this target-guided programming capability through FEM simulations and physical experiments.

2.2 | Target-Guided Programmable Morphing of Dimple-Encoded Strips

We next use the trained forward–inverse framework to program dimple-encoded strips toward prescribed target configurations (Figure 3). Representative target shapes are selected to span different levels and distributions of curvature, including weakly curved, strongly curved, and more demanding high-turning profiles (Figure 3a). These image-derived profiles are processed using the same arc-length resampling, tangent extraction, and smoothing pipeline as the FEM-generated dataset, ensuring consistency between the target representation and the surrogate output space (Figure S3).

For each target, the inverse-design loop optimizes the dimple encoding to minimize the mismatch between the CNN-predicted configuration and the prescribed target profile (Figure 3b). The optimization returns physically realizable combinations of dimple gaps and heights under the geometric constraints defined above. The resulting CNN predictions capture the dominant global bending features over a broad range of target morphologies, showing that the learned forward model can be used as a differentiable surrogate for target-guided geometric programming (Note S4).

We first validate the optimized designs using constant-width strips, for which the global configuration is controlled primarily through the spatial distribution and height of the dimples (Note S5). Independent nonlinear FEM simulations and physical prototypes reproduce the dominant programmed bending trends predicted by the surrogate within the morphology range accessible to the present strip design space (Figure 3c). The more demanding closed-loop and highly coiled targets,

15214095, 0, Downloaded from https://advanced.onlinelibrary.wiley.com/doi/10.1002/adma.75796, Wiley Online Library on [24/11/2026]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

however, are only partially reproduced and should therefore be interpreted as tests of the current design-space limits rather than demonstrations of exact closure or multiple-turn winding. The present surrogate is trained on open strips with fixed ranges of strip length, dimple number, and dimple geometry, which limits the accumulated turning that can be generated by a single strip. Consistent with this interpretation, an open half-circle target can be closely reproduced within the current design domain (Figure S5, Note S5.6), whereas nonlinear FEM of an extended multi-dimple strip demonstrates that substantially larger accumulated curvature and genuine coiling become mechanically accessible when the structural design space is enlarged (Figure S6, Note S5.7). These results indicate that larger dimple numbers or strip lengths, together with training data and objective functions tailored to closure, winding, or self-contact, could extend the framework toward more extreme target geometries.

We then apply the same inverse-design strategy to variable-width strips. In this case, the prescribed width profile is included as an explicit geometric condition during optimization, allowing the dimple coding to be programmed while accounting for geometry-dependent bending responses. Representative trapezoid and barrel strips are inverse-designed and validated by both FEM simulations and experiments (Figure 3d). The comparison between constant- and variable-width examples also reveals that the inverse mapping is not generally unique: distinct combinations of strip geometry and dimple coding can generate comparable global centerline profiles within finite geometric and tangent-space tolerances (Note S7). This many-to-one character is a natural consequence of the distributed mechanical interactions among dimples and the strip planform. Importantly, it does not imply that strip geometry is redundant. In higher-level architectures, the planform may instead be prescribed by application-level geometric constraints; the inverse problem then becomes one of determining a compatible dimple coding for that imposed strip geometry. This distinction becomes important in the shell construction introduced in Section 2.3.

The accessible morphology space is also broader than the predominantly single-sign curvature targets used above. FEM simulations reveal configurations containing changes in curvature sign, including S-shaped and wave-like profiles with one or more inflection points (Figure S8, Note S9). These examples show that the dimple-encoded strips are not intrinsically restricted to one-sided concave bending and that interactions between neighboring dimples can generate spatially varying bending directions along the same strip. Local dimple bistability provides an additional discrete degree of freedom: for a fixed fabricated geometry, switching different subsets of dimples can lead to distinct stable configurations and mechanical responses (Figure S9, Table S3, Note S10). The present inverse-design demonstrations focus primarily on geometric encoding with a prescribed set of activated dimples, while incorporating the activation pattern itself as a discrete design variable would extend the framework toward post-fabrication reconfiguration of a single manufactured structure.

Together, these results establish target-guided programming across both constant- and variable-width strips, while also identifying the non-uniqueness and finite turning range of the present inverse-design space. The variable-width formulation further provides the geometric bridge from isolated strip programming to higher-dimensional assembly, where the strip planform is constrained by the architecture rather than selected freely. We next exploit this capability to construct tessellated three-dimensional morphing shells.

2.3 | Shell Morphogenesis From Tessellated Strips

We next extend the strip-level programming strategy to create three-dimensional shells by rotationally tessellating inverse-designed tapered strip modules (Figure 4). In this way, shells are formed from strips that organize their collective shape—a process we refer to as shell morphogenesis. This step introduces a geometric constraint absent from isolated strips: neighboring modules must meet after assembly to form a closed surface. As a result, the strip width can no longer be selected solely to expand the accessible morphing space, but must also satisfy the closure condition imposed by the target shell geometry. We therefore separate the design problem into two coupled parts: geometry determines the planar width profile of the strip required for tessellation, while learning determines the dimple coding required to generate the desired meridional curvature (Note S6).

A dome-like architectural roof is selected as the target geometry and used to define the desired meridional profile (Figure 4a). For an axisymmetric shell assembled from N identical strip modules, closure requires the width of each module to match the circumferential spacing of one sector [6, 35]. This gives the required width profile

$$w^*(\xi) = 2\rho^*(\xi)\tan\left(\frac{\pi}{N}\right) \tag{13}$$

where $\rho^*(\xi)$ is the radial distance of the target meridian from the symmetry axis. This relation converts the target shell geometry into a tapered strip module with a prescribed width profile (Figure 4b), which can then be rotationally tessellated into an $N = 12$ planar layout (Figure 4c). In this way, shell closure is built into the undeformed planform before actuation, rather than being corrected after deformation.

The target meridian is rescaled to the prescribed module length and converted into the same smoothed tangent representation used for strip-level inverse design, while Equation (13) maps its radial coordinate to the planar strip width. The resulting width profile therefore encodes the circumferential closure requirement, whereas the tangent profile specifies the meridional deformation to be programmed by the dimples. This geometric construction provides a nominal tessellation prescription; possible seam gaps, overlaps, and transverse deformations are subsequently assessed using the full-shell FEM model and experiments (Note S6).

Once the width profile is fixed by tessellation, the remaining programmable degree of freedom is the dimple encoding along each strip module. This differs from the variable-width demonstrations in Section 2.2, where the width profile can be selected freely. Here,

15214095, 0, Downloaded from https://advanced.onlinelibrary.wiley.com/doi/10.1002/adma.75796, Wiley Online Library on [24/11/2026]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

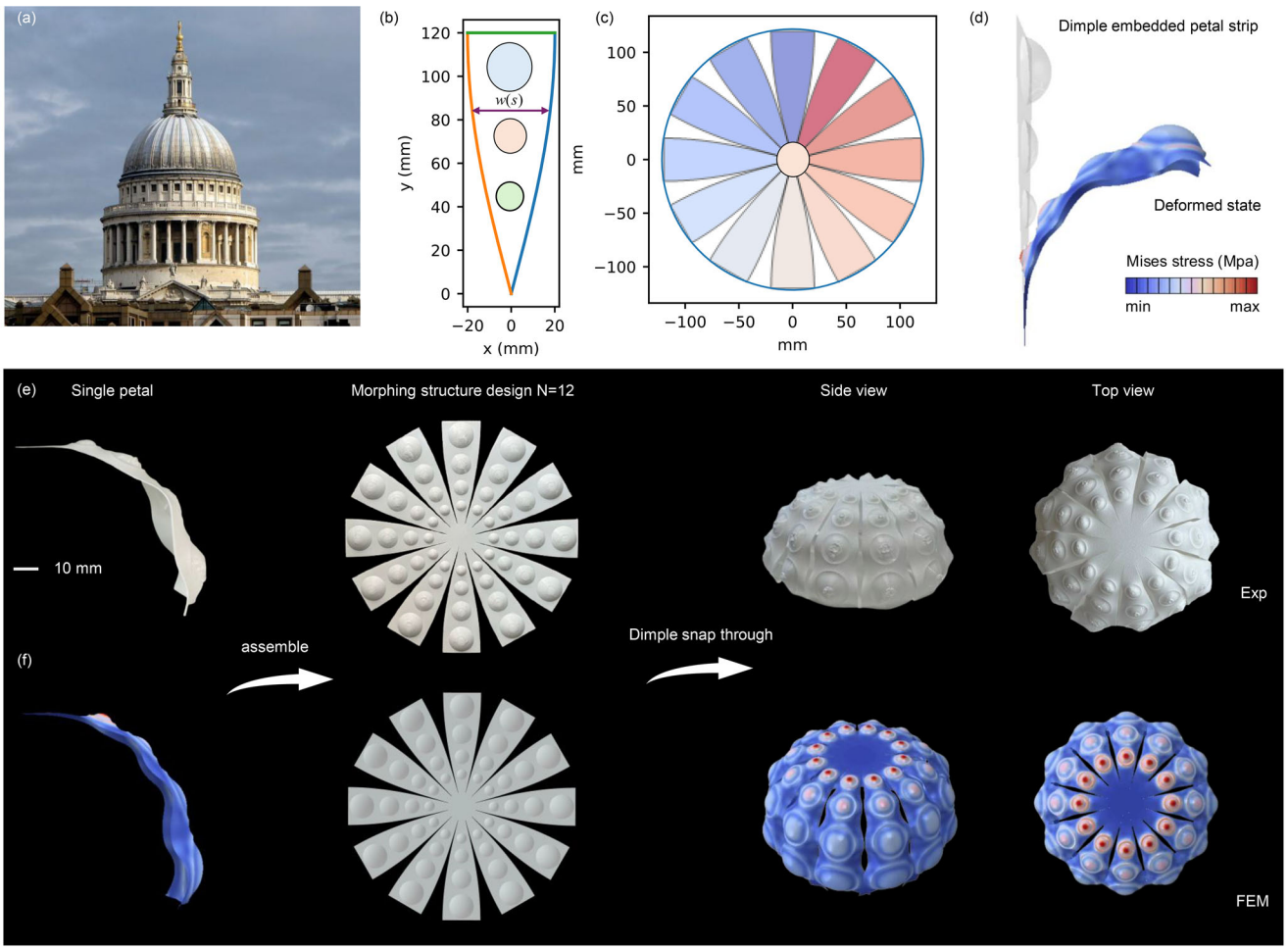


FIGURE 4 | Shell morphogenesis from tessellated dimple-encoded strips. (a) Dome-inspired target geometry used to define the desired meridional profile. (b,c) Variable-width tapered strip module with a prescribed width profile $w(s)$ and its rotational tessellation into an $N = 12$ planar layout. (d) FEM simulation of a single module, showing the undeformed strip and the deformed state after dimple snap-through. (e) Experimental and FEM demonstration of shell morphogenesis: individually programmed strip modules are assembled into a planar tessellation and transformed into a dome-like shell, shown in side and top views. Photograph in (a): St Paul's Cathedral by Bernard Gagnon (CC BY-SA 3.0). Full attribution and license details are provided in the Image Credits and Permissions section.

the width is imposed by the target shell geometry, and the inverse-design task is to identify a feasible dimple pattern that bends the module toward the prescribed meridional configuration under this constraint. We use the trained width-conditioned surrogate to optimize the dimple locations and heights while enforcing local feasibility constraints. The optimized module is then simulated independently by FEM to verify that the initially flat strip transforms into the desired curved meridional profile after dimple snap-through (Figure 4d).

The inverse-designed module is finally replicated and assembled by azimuthal rotation to form the complete shell. Both FEM simulations and physical experiments show that the planar tessellation transforms into a dome-like shell after dimple snap-through, with consistent side and top views between simulation and experiment (Figure 4e). The assembled shell retains the main features of the prescribed meridional curvature while satisfying the global geometric requirement of rotational closure. At the shell level, the meridional shape is therefore programmed through the dimple encoding, while the circumferential organization is imposed geometrically through tessellation.

A continuous two-dimensional sheet patterned directly with bistable dimples could, in principle, also be used to generate dome-like morphologies. Such a formulation would, however, require the inverse-design framework to account simultaneously for deformation in two in-plane directions, surface compatibility, and the coupled interactions of dimples distributed over the full sheet [25]. The tessellated-strip construction used here provides a modular alternative by reducing the learned morphing problem to one-dimensional meridional programming while enforcing circumferential compatibility through the prescribed planform [6]. This simplification comes at the cost of discrete seams and a finite number of modules, and the present demonstration is restricted to an axisymmetric shell assembled from identical strips. Extension to general non-axisymmetric surfaces would require non-identical modules and a correspondingly broader geometry-conditioned design space [40].

15214095, 0, Downloaded from https://advanced.onlinelibrary.wiley.com/doi/10.1002/adma.75796, Wiley Online Library on [24/11/2026]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

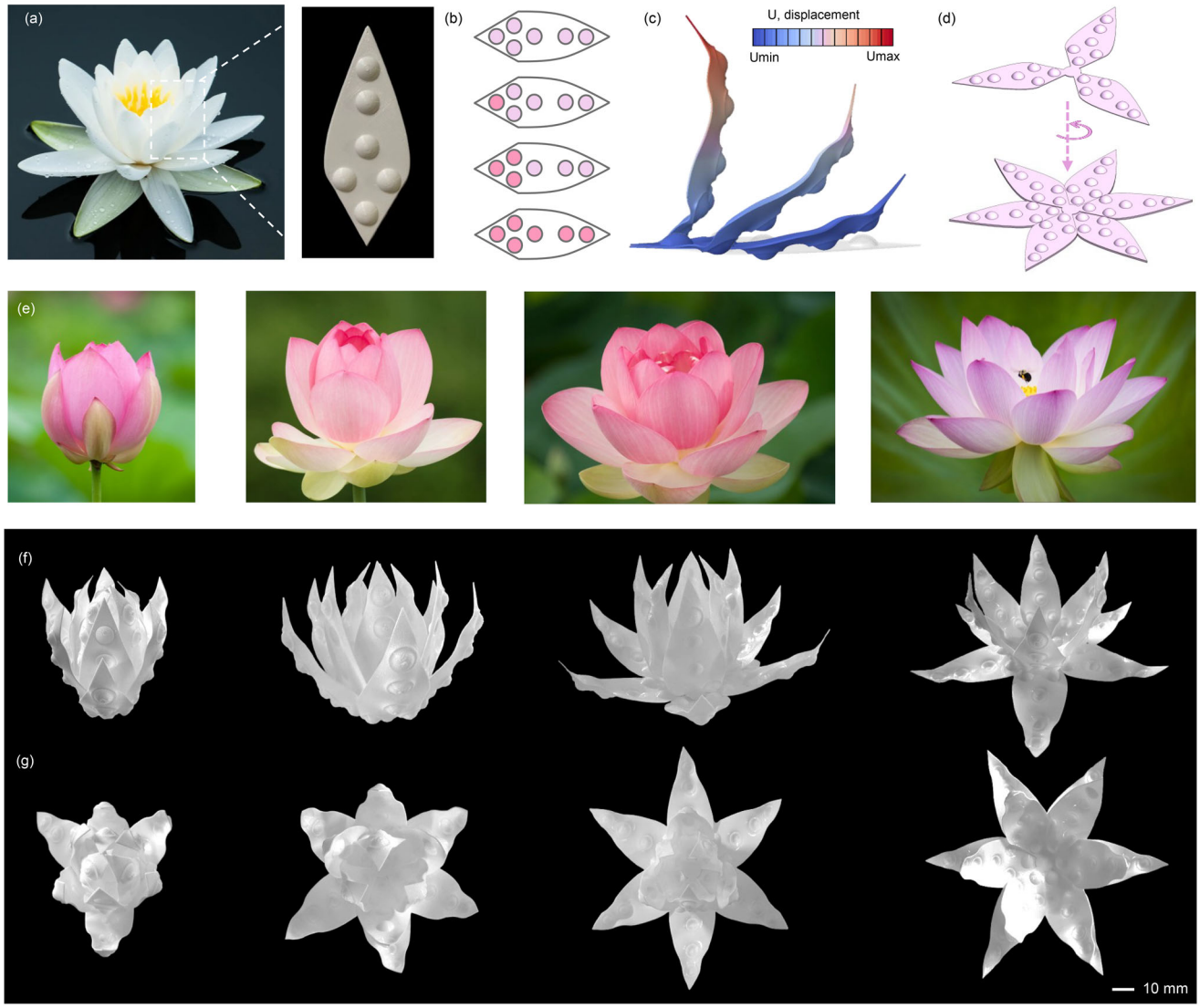


FIGURE 5 | Digital flower based on hierarchical folding–morphing modules (a) Lotus-inspired design concept and a dimple-encoded petal module (b,c) Digital dimple coding for programming different folding–morphing responses, illustrated by representative dimple patterns and corresponding FEM deformation modes. (d) Rotational assembly of petal modules into a multilayer flower architecture. (e) Reference lotus blooming sequence. (f,g) Experimental demonstrations of the digital flower during sequential actuation, shown in side and top views. Lotus photographs in (a) and (e) reproduced with permission. Copyright Todd Henson Photography. These photographs are excluded from the article's Creative Commons licence. Full permission details are provided in the Image Credits and Permissions section.

This shell-level demonstration shows how local bistable programming can be combined with geometric tessellation to create deployable three-dimensional architectures from initially planar strip modules. It also establishes an intermediate design level between individual morphing strips and more complex hierarchical architectures, which we explore next through flower-inspired folding–morphing assemblies.

2.4 | Digital Flower: Hierarchical Folding–Morphing Architectures

Having established programmable morphing in individual strips and tessellated shells, we finally use dimple-encoded petal modules to construct a digital flower through hierarchical folding–morphing assembly (Figure 5). The design is inspired by the layered opening of a lotus, in which multiple petals bend, fold, and deploy cooperatively from a compact bud-like state to an open flower morphology (Figure 5a,e). Each petal module is based on a tapered dimple-encoded strip (Figure 5a), enabling digital dimple programming to capture the main geometric features of flower opening through modular petal design and multilayer assembly. The resulting hierarchy connects local dimple states to petal-level folding and bending, and ultimately to the collective opening of the multilayer architecture. Essential construction details are summarized here, with dimensions and additional intermediate configurations provided in Note S8 (Figure S7).

To enable large changes in opening angle, we introduce a hinge-like dimple arrangement near the petal root [27]. This localized root region acts as a programmed folding zone, allowing the petal to rotate and deploy over a relatively large angular range.

15214095, 0, Downloaded from https://advanced.onlinelibrary.wiley.com/doi/10.1002/adma.75796, Wiley Online Library on [24/11/2026]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

In contrast, dimples distributed over the petal body tune the post-deployment bending profile. This separation between root folding and body curvature provides a simple design principle for combining discrete folding with continuous morphing in a single petal module. Representative digital dimple encodings and the corresponding FEM-predicted deformation modes illustrate how different folding–morphing responses can be generated by changing the encoded dimple pattern, especially near the root region (Figure 5b,c). For assembly, three petals are arranged at 120° intervals and combined with a second set rotated by 60° to form a six-petal layer. Multiple such layers are then stacked around a common central hub with staggered petal orientations, reducing direct overlap and producing a fuller radial flower-like planform.

Different dimple encodings are assigned to the petal modules to regulate the root folding angle and the curvature developed along the petal body. Sequential manual switching of these prescribed dimples produces a qualitative progression from compact, through partially opened, to fully opened configurations (Figure 5e–g). The corresponding side views capture the gradual increase in opening angle, while the top views show the development of the radially organized flower morphology. The complete actuation sequence is provided in Supplementary Movie S1. Because the dimples are bistable, the deployed configurations remain stable after the external pressing force is removed. The structures are mechanically resettable rather than autonomously reversible: returning to the initial configuration requires the dimples to be manually switched back to their original stable states.

Unlike the target-guided strip examples in Section 2.2, the complete flower is not obtained through an end-to-end inverse design of the entire blooming sequence. Instead, it demonstrates how the local programming principles established at the strip level can be composed hierarchically: root-localized dimples generate folding, distributed dimples tune petal bending, and rotational multilayer assembly organizes these local responses into a collective three-dimensional transformation. This modular construction illustrates how spatially differentiated mechanical functions can be encoded in individual structural regions and subsequently combined at the architectural scale.

3 | Conclusions and Outlook

We developed a learning-enabled forward–inverse framework for programming large deformation morphing of dimple-encoded elastic strips and their assembled architectures. By embedding discrete dimple encodings and strip geometries into continuous one-dimensional fields and representing deformed configurations through smoothed tangent fields, the framework links local bistable encoding to global shape formation in a differentiable form. This enables target-guided programming of both constant-width and variable-width strips, with optimized designs validated by independent FEM simulations and physical prototypes. The results further reveal the geometry-dependent and many-to-one nature of the inverse mapping: distinct feasible combinations of strip geometry and dimple coding can generate comparable global morphologies, while higher-level assembly constraints can instead prescribe the strip geometry and leave the dimple coding as the principal programmable variable.

Beyond individual strips, we demonstrated how programmed strip modules can be assembled into higher-level morphing architectures. Closure-imposed tapered strip profiles, combined with learned dimple programming, enable initially planar tessellations to transform into dome-like shells. Further integration of root-localized hinge-like folding, distributed petal bending, and multilayer rotational assembly gives rise to a digital flower that qualitatively captures lotus-like staged opening. Together with the inflection-containing and two-sided strip configurations, these demonstrations show how local dimple programming can be composed across multiple structural levels, from distributed curvature generation within a strip to geometry-constrained tessellation and multi-layer hierarchical three-dimensional assembly.

The present framework also has clear boundaries. The inverse design is restricted to the geometric, material, and actuation ranges represented in the training data, and extreme targets requiring large accumulated turning, closure, or self-contact may require an enlarged structural and training space. The shell demonstration is currently limited to an axisymmetric tessellation of identical modules, while the digital flower is a hierarchical proof of concept rather than an end-to-end inverse design of a complete deployment pathway. Actuation is presently manual, and the structures are mechanically resettable rather than autonomously reversible. At the same time, selective activation of different dimple subsets introduces an additional discrete state variable, allowing a fixed fabricated geometry to access distinct stable configurations and motivating future inverse-design formulations that jointly optimize geometry and activation state.

Looking forward, this framework could be extended from static target configurations to morphing pathways, enabling the design of structures that follow prescribed deployment sequences rather than only final shapes. Potential applications include shape-matching robotic elements, adaptive interfaces and skins, and deployable shell-like structures, where bistability can provide passive retention of a programmed configuration without continuous energy input. Integrating distributed pneumatic, magnetic, or other addressable actuation with sensing and feedback would further enable remote and repeatable switching between programmed states. More broadly, extending the framework to richer material behaviors, non-axisymmetric architectures, and state-conditioned design could provide a route toward adaptive structures whose global transformations are encoded jointly through geometry and local mechanical states.

4 | Experimental Section

4.1 | Fabrication, Actuation, and Assembly

All specimens were fabricated by fused deposition modeling (FDM) using Overture TPU 95A filament on a Raise3D E2 printer. The printed TPU had an elastic modulus of $E = 12$ MPa and a Poisson's ratio of $\nu = 0.45$. Before printing, the filament was dried at 70°C for 7 h to minimize moisture-induced defects. Printing was performed with a 0.4 mm nozzle and a layer height

15214095, 0, Downloaded from https://advanced.onlinelibrary.wiley.com/doi/10.1002/adma.75796, Wiley Online Library on [24/11/2026]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

of 0.1 mm. The printing speed was set to 20 mm s^{-1}, and the build-plate and nozzle temperatures were maintained at 45°C and 223°C, respectively. The isolated dimple-encoded strips were printed as monolithic TPU components, with the bistable dimples integrated directly into the strip geometry.

The specimens were actuated manually by pressing the apex of each prescribed dimple until snap-through occurred. After the applied force was removed, the inverted dimples remained in their second stable states and the corresponding global morphology was retained. Mechanical resetting was achieved by manually snapping the dimples back to their original stable states; the structures are therefore mechanically resettable rather than autonomously reversible.

The higher-level demonstrators were assembled from separately printed modules. For the tessellated shell, the tapered strip modules were arranged according to the prescribed rotational tessellation and the dimples were actuated to generate the dome-like configuration. For the digital flower, separately printed petal modules were assembled into staggered multilayer sets and selected dimple patterns were switched sequentially to produce the staged opening sequence. The corresponding experimental transformations are shown in Supplementary Movie S1. Detailed dimensions, joining procedures, and assembly layouts are provided in Notes S6 and S8.

4.2 | FEM Simulation and Loading Protocol

Three-dimensional geometries were designed in SolidWorks 2024 and analyzed in Abaqus/Standard on the University of Birmingham BlueBEAR high-performance computing (HPC) cluster. The structures were discretized using four-node reduced-integration shell elements (S4R). An implicit dynamic procedure operated in the quasi-static regime was used to capture snap-through instability and post-buckling deformation. A three-step loading–relaxation protocol was implemented. In the FEM simulations, pressure loading was used solely as a numerical means of triggering dimple inversion; the experimental specimens were actuated manually as described above. First, the boundary edges of the sheet were fully constrained, and a pressure load was applied to the dimpled regions using a linear ramp to trigger dimple inversion. Second, the pressure load was removed while the boundary constraints were maintained, allowing the structure to relax toward an equilibrium configuration. Third, all boundary constraints were released to obtain the final stress-relaxed free-standing configuration. The TPU was modeled as a compressible Neo-Hookean hyperelastic material with parameters $C_{10} = 2.069$ MPa and $D_1 = 0.05$ MPa^{-1}. Energy-based convergence criteria were monitored to ensure numerical stability. Mesh-independence studies were performed before the parametric analyses to verify that the predicted deformation modes and equilibrium configurations were insensitive to further mesh refinement.

For the auxiliary state-dependent mechanical-response study in Note S10, a common eight-dimple strip geometry and identical numerical settings were used for all selected activation codes, so that only the dimple activation state was varied. After each coded configuration was allowed to relax, the strip was subjected to the same transverse indentation protocol and its force–displacement response was evaluated through the secant stiffness at 10 mm indentation, $k_{eq,10} = F(10\,mm)/(10\,mm)$, and the corresponding indentation work, $W_{10} = \int_0^{10\,mm} F(\delta)\,d\delta$. A common linear-elastic material representation was used for this comparative scan to isolate the relative influence of activation state. These quantities are therefore interpreted as configuration-dependent comparative measures rather than intrinsic material properties or direct experimental stiffness measurements. Full loading, constraint, and post-processing procedures are provided in Note S10.

4.3 | Learning Implementation and Computational Cost

The forward surrogate and gradient-based inverse-design procedure were implemented in PyTorch. The surrogate was trained on 3000 accepted FEM-generated configurations using an NVIDIA A100 GPU; detailed network architecture, training hyperparameters, loss functions, and inverse-optimization settings are provided in Notes S4 and S5.

Generation of the FEM database constituted the main one-time computational cost, with each nonlinear simulation requiring approximately 24 min on average and the complete dataset requiring approximately 101 h of elapsed wall-clock time under parallel execution on the BlueBEAR cluster. CNN training required approximately 2.6 h, while a multi-start inverse-design search required approximately 23 min per target. The trained surrogate can be reused for new targets and design combinations within the represented geometric and material domain; additional training is required when extending substantially beyond this domain.

Author Contributions

Qun Zhang: methodology, software, data curation, formal analysis, validation, investigation, visualization, writing – original draft. **Omid Doustdar**: investigation, validation, formal analysis, writing – review & editing. **Hyunyoung Kim**: investigation, validation, formal analysis, writing – review & editing. **Claire Dancer**: investigation, validation, formal analysis, writing – review & editing. **Zhaohe Dai**: investigation, validation, formal analysis, writing – review & editing. **Weicheng Huang**: methodology, investigation, validation, formal analysis, writing – review & editing. **Karl Dearn**: investigation, validation, formal analysis, supervision, resources, funding acquisition, writing – review & editing. **Dominic Vella**: conceptualization, investigation, validation, formal analysis, writing – review & editing. **Mingchao Liu**: conceptualization, methodology, investigation, validation, formal analysis, supervision, funding acquisition, visualization, project administration, writing – review & editing.

Acknowledgments

Q.Z. acknowledges the scholarship support from the School of Engineering at the University of Birmingham. M.L. acknowledges the start-up funding from the University of Birmingham, UK. OpenAI ChatGPT (GPT-5.6 Sol) was used in August 2026 for language editing and refinement of selected manuscript and Supporting Information text. All AI-assisted content was reviewed and verified by the authors against the underlying methods and results. No sensitive, personal, or confidential data were provided to the tool.

15214095, 0, Downloaded from https://advanced.onlinelibrary.wiley.com/doi/10.1002/adma.75796, Wiley Online Library on [24/11/2026]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

Conflicts of Interest

The authors declare no conflicts of interest.

Data Availability Statement

The data that support the findings of this study are available from the corresponding author upon reasonable request.

Image Credits and Permissions

The three reference photographs in Figure 3a were obtained from Wikimedia Commons/Wikipedia: (i) Top: *"ArtScience Museum, Marina Bay Sands, Singapore"* by Basile Morin (CC BY-SA 4.0), https://en.wikipedia.org/wiki/File:ArtScience_Museum,_Marina_Bay_Sands,_Singapore.jpg ; (ii) Middle: *"Brussels – Atomium 2022.jpg"* by Marek Śliwecki (CC BY-SA 4.0), https://commons.wikimedia.org/wiki/File:Brussels_-_Atomium_2022.jpg ; (iii) Bottom: *"Pounamu pendant.jpg"* by Andrew Sullivan (CC BY-SA 3.0), https://commons.wikimedia.org/wiki/File:Pounamu_pendant.jpg. All three photographs were adapted by adding red dashed guide curves to indicate the target morphing profiles. The images are used under the Creative Commons Attribution-ShareAlike licenses: CC BY-SA 4.0 (https://creativecommons.org/licenses/by-sa/4.0/) and CC BY-SA 3.0 (https://creativecommons.org/licenses/by-sa/3.0/). The adaptations remain subject to the respective license terms.

The photograph in Figure 4a, *"Cathédrale St-Paul – coupole.jpg"* by Bernard Gagnon (CC BY-SA 3.0), was obtained from Wikimedia Commons: https://commons.wikimedia.org/wiki/File:Cath%C3%A9drale_St-Paul_-_coupole.jpg. The photograph is used under the Creative Commons Attribution-ShareAlike 3.0 Unported license (https://creativecommons.org/licenses/by-sa/3.0/).

The lotus photographs used in Figure 5a,e are reproduced with permission from Todd Henson Photography (https://toddhensonphotography.com). The images include photographs from "Stages of a Lotus Flower Blooming at Kenilworth Aquatic Gardens" and the Flower Photography Gallery. Copyright Todd Henson Photography. These photographs remain the copyright of Todd Henson Photography and are not covered by the article's Creative Commons licence; any reuse requires permission from Todd Henson Photography as the copyright holder.

References

1. K. K. Dudek, M. Kadic, C. Coulais, and K. Bertoldi, "Shape-Morphing Metamaterials," *Nature Reviews Materials* 10, no. 10 (2025): 783–798, https://doi.org/10.1038/s41578-025-00828-9.

2. K. Oliver, A. Seddon, and R. S. Trask, "Morphing in Nature and Beyond: A Review of Natural and Synthetic Shape-Changing Materials and Mechanisms," *Journal of Materials Science* 51, no. 24 (2016): 10663–10689, https://doi.org/10.1007/s10853-016-0295-8.

3. I. P. S. Qamar, R. Groh, D. Holman, and A. Roudaut, "HCI Meets Material Science: A Literature Review of Morphing Materials for the Design of Shape-Changing Interfaces," in *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (ACM, 2018), 1–23.

4. X. Yang, Y. Zhou, H. Zhao, et al., "Morphing Matter: From Mechanical Principles to Robotic Applications," *Soft Science* 3, no. 4 (2023): 38, https://doi.org/10.20517/ss.2023.42.

5. S. Xu, R. Yang, Y. Yang, and Y. Zhang, "Shape-morphing bioelectronic devices," *Materials Horizons* 12, no. 16 (2025): 5994–6017, https://doi.org/10.1039/d5mh00453e.

6. M. Liu, L. Domino, and D. Vella, "Tapered Elasticæ as a Route for Axisymmetric Morphing Structures," *Soft Matter* 16, no. 33 (2020): 7739–7750, https://doi.org/10.1039/d0sm00714e.

7. P. Celli, C. McMahan, B. Ramirez, et al., "Shape-Morphing Architected Sheets With Non-Periodic Cut Patterns," *Soft Matter* 14, no. 48 (2018): 9744–9749, https://doi.org/10.1039/c8sm02082e.

8. Y. Yang, Y. Wang, M. Lin, M. Liu, and C. Huang, "Bio-Inspired Facile Strategy for Programmable Osmosis-Driven Shape-Morphing Elastomer Composite Structures," *Materials Horizons* 11, no. 9 (2024): 2180–2190, https://doi.org/10.1039/d3mh01731a.

9. D. Duffy, J. S. Biggins, I. Griniasty, et al., "Nematic Design for Shape Morphing," *Annual Review of Control, Robotics, and Autonomous Systems* 9, no. 1 (2026): 75–98, https://doi.org/10.1146/annurev-control-030624-011240.

10. Y. Cai, C. Wang, M. Yang, Y. Li, F. Zhang, and S. Wang, "Bioinspired Programmable Biaxial Rolling Gel Sheets for Complex 3D Morphing," *Advanced Materials* 38, no. 13 (2026): e19226, https://doi.org/10.1002/adma.202519226.

11. Y. Yang, K. Vella, and D. P. Holmes, "Grasping With Kirigami Shells," *Science Robotics* 6, no. 54 (2021): eabd6426, https://doi.org/10.1126/scirobotics.abd6426.

12. J. Wang and A. Chortos, "Performance Metrics for Shape-Morphing Devices," *Nature Reviews Materials* 9, no. 10 (2024): 738–751, https://doi.org/10.1038/s41578-024-00714-w.

13. M. Liu, "Mimicking Prions for Self-Replicating Machines," *Newton* 1, no. 5 (2025): 100107, https://doi.org/10.1016/j.newton.2025.100107.

14. J. Chen, J. Jiang, J. Weber, V. Gimenez-Pinto, and C. Peng, "Shape Morphing by Topological Patterns and Profiles in Laser-Cut Liquid Crystal Elastomer Kirigami," *ACS Applied Materials & Interfaces* 15, no. 3 (2023): 4538–4548, https://doi.org/10.1021/acsami.2c20295.

15. J. Chen, A. S. Johnson, J. Weber, O. I. Akomolafe, J. Jiang, and C. Peng, "Programmable Light-Driven Liquid Crystal Elastomer Kirigami With Controlled Molecular Orientations," *Advanced Intelligent Systems* 4, no. 7 (2022): 2100233, https://doi.org/10.1002/aisy.202100233.

16. Z. Lei, Z. Chai, K. Wan, et al., "Aquatic Topological Liquid Crystal Gel Kirigami," *Advanced Functional Materials* 36, no. 12 (2026): e12059, https://doi.org/10.1002/adfm.202512059.

17. N. Oppenheimer and T. A. Witten, "Shapeable Sheet Without Plastic Deformation," *Physical Review E* 92, no. 5 (2015): 052401, https://doi.org/10.1103/PhysRevE.92.052401.

18. Y. Liu, F. Pan, B. Ding, Y. Zhu, K. Yang, and Y. Chen, "Multistable Shape-Reconfigurable Metawire in 3D Space," *Extreme Mechanics Letters* 50 (2022): 101535, https://doi.org/10.1016/j.eml.2021.101535.

19. G. Risso, M. Sakovsky, and P. Ermanni, "A Highly Multi-Stable Meta-Structure via Anisotropy for Large and Reversible Shape Transformation," *Advanced Science* 9, no. 26 (2022): 2202740, https://doi.org/10.1002/advs.202202740.

20. A. Rahman, S. Ferracin, S. Tank, C. Zhang, and P. Celli, "Shape-Retaining Beam-Like Morphing Structures via Localized Snap Through," *International Journal of Solids and Structures* 300 (2024): 112917, https://doi.org/10.1016/j.ijsolstr.2024.112917.

21. E. Ben-Abu, A. Zigelman, Y. Veksler, et al., "Reprogrammable 3D Shapes From 1D Metamaterial," *Advanced Materials Technologies* 10, no. 4 (2025): 2401113, https://doi.org/10.1002/admt.202401113.

22. A. S. Meeussen and M. Van Hecke, "Multistable Sheets With Rewritable Patterns for Switchable Shape-Morphing," *Nature* 621, no. 7979 (2023): 516–520, https://doi.org/10.1038/s41586-023-06353-5.

23. G. Risso, M. Kudisch, P. Ermanni, and C. Daraio, "Tuning the Properties of Multi-Stable Structures Post-Fabrication Via the Two-Way Shape Memory Polymer Effect," *Advanced Science* 11, no. 21 (2024): 2308903, https://doi.org/10.1002/advs.202308903.

24. K. A. Seffen, "Mechanical Memory Metal: A Novel Material for Developing Morphing Engineering Structures," *Scripta Materialia* 55, no. 4 (2006): 411–414, https://doi.org/10.1016/j.scriptamat.2006.03.028.

25. M. Liu, L. Domino, I. Dupont de Dinechin, M. Taffetani, and D. Vella, "Snap-Induced Morphing: From a Single Bistable Shell to the Origin of Shape Bifurcation in Interacting Shells," *Journal of the Mechanics and Physics of Solids* 170 (2023): 105116, https://doi.org/10.1016/j.jmps.2022.105116.

15214095, 0, Downloaded from https://advanced.onlinelibrary.wiley.com/doi/10.1002/adma.75796, Wiley Online Library on [24/11/2026]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

26. F. Albertini, M. G. Tarantino, and L. Daniel, "Mechanical Behavior of Embedded Bistable Dome Shell With Tunable Energy Barrier Asymmetry," *International Journal of Mechanical Sciences* 263 (2024): 108762, https://doi.org/10.1016/j.ijmecsci.2023.108762.

27. Q. Zhang, W. Huang, A. Hajiyavand, et al., "Dimple-Encoded Reprogrammable Origami," *Advanced Science* 14, no. 70 (2026): e76901, https://doi.org/10.1002/advs.76901.

28. J. A. Faber, J. P. Udani, K. S. Riley, A. R. Studart, and A. F. Arrieta, "Dome-Patterned Metamaterial Sheets," *Advanced Science* 7, no. 22 (2020): 2001955, https://doi.org/10.1002/advs.202001955.

29. J. P. Udani and A. F. Arrieta, "Taming Geometric Frustration by Leveraging Structural Elasticity," *Materials & Design* 221 (2022): 110809, https://doi.org/10.1016/j.matdes.2022.110809.

30. T. Brzin and M. Brojan, "Using a Generative Adversarial Network for the Inverse Design of Soft Morphing Composite Beams," *Engineering Applications of Artificial Intelligence* 133 (2024): 108527, https://doi.org/10.1016/j.engappai.2024.108527.

31. D. Samadian, I. B. Muhit, and N. Dawood, "Application of Data-Driven Surrogate Models in Structural engineering: A Literature Review," *Archives of Computational Methods in Engineering* 32, no. 2 (2025): 735–784, https://doi.org/10.1007/s11831-024-10152-0.

32. S. Lee, S. Kang, J. Nam, J. Yu, M. Kim, and S. Ryu, "Inverse Design of Thermally Active Composite via Policy-Transferred Reinforcement Learning," *Materials Horizons* 13, no. 13 (2026): 6500–6516, https://doi.org/10.1039/D6MH00239K.

33. Z. Xu, S. Zhang, X. Sun, S. Mao, and R. Xiao, "Machine Learning-Assisted Inverse Design of 3D Shape Morphing in Liquid Crystal Elastomer Composite Strips," *Advanced Functional Materials* 36, no. 29 (2026): e26924, https://doi.org/10.1002/adfm.202526924.

34. Z. Fan, Y. Yang, F. Zhang, et al., "Inverse Design Strategies for 3D Surfaces Formed by Mechanically Guided Assembly," *Advanced Materials* 32, no. 14 (2020): 1908424, https://doi.org/10.1002/adma.201908424.

35. Y. Zhang, J. Yang, M. Liu, and D. Vella, "Shape-Morphing Structures Based on Perforated Kirigami," *Extreme Mechanics Letters* 56 (2022): 101857, https://doi.org/10.1016/j.eml.2022.101857.

36. S. Nair, T. F. Walsh, G. Pickrell, and F. Semperlotti, "GRIDS-Net: Inverse Shape Design and Identification Of Scatterers via Geometric Regularization and Physics-Embedded Deep Learning," *Computer Methods in Applied Mechanics and Engineering* 414 (2023): 116167, https://doi.org/10.1016/j.cma.2023.116167.

37. M.-C. Chen, Y.-C. Li, Y.-Y. Yu, et al., "Deep Learning-Based Inverse Design and Forward Prediction of Bi-Material 4D-Printed Facial Shells," *Soft Science* 5, no. 4 (2025): 57, https://doi.org/10.20517/ss.2025.81.

38. Y. Ren, X. Deng, Z. You, and X. Zheng, "1-D Multi-Channel CNN With Transfer Functions for Inverse Electromagnetic Behaviors Modeling and Design Optimization of High-Dimensional Filters," *Applied Intelligence* 54, no. 1 (2024): 503–521, https://doi.org/10.1007/s10489-023-05200-4.

39. Y. Wang, X. Huang, S. Zhang, Z. Yao, and L. Gao, "CNN-Based Inverse Design of Coding Metamaterial Absorbers," *Optics Communications* 592 (2025): 132251, https://doi.org/10.1016/j.optcom.2025.132251.

40. X. Yang, M. Liu, B. Zhang, et al., "Hierarchical Tessellation Enables Programmable Morphing Matter," *Matter* 7, no. 2 (2024): 603–619, https://doi.org/10.1016/j.matt.2023.11.002.

Supporting Information

Additional supporting information can be found online in the Supporting Information section.

Supporting File 1: adma75796-sup-0001-SuppMat.pdf.

Supporting File 2: adma75796-sup-0002-MovieS1.mp4.

15214095, 0, Downloaded from https://advanced.onlinelibrary.wiley.com/doi/10.1002/adma.75796, Wiley Online Library on [24/11/2026]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License