DeepMetaHandles: Learning Deformation Meta-Handles of 3D Meshes with Biharmonic Coordinates

Minghua Liu[1] Minhyuk Sung[2] Radomir Mech[3] Hao Su[1]

[1]University of California San Diego [2]KAIST [2]Adobe Research

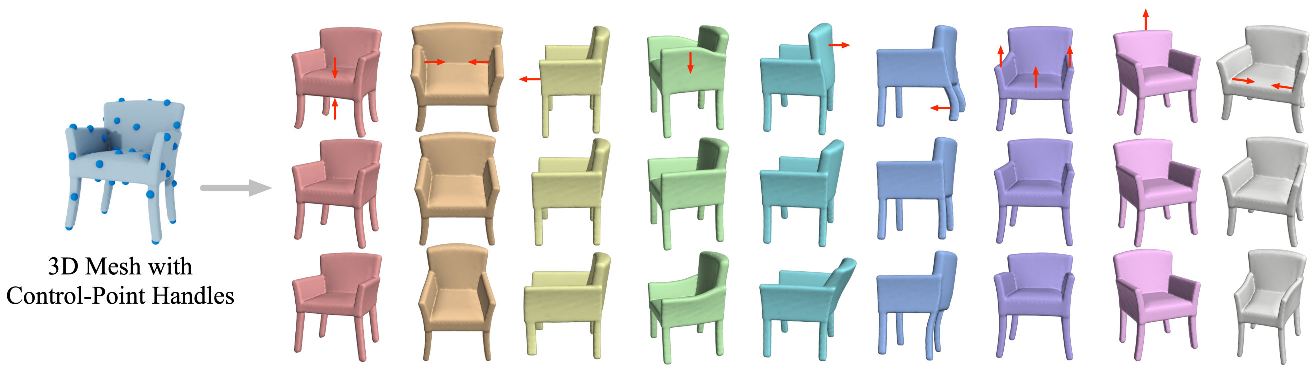

Figure 1: Learned meta-handles for a single chair. Each column indicates a meta-handle and shows three deformations along the direction of that meta-handle, with red arrows highlighting the deformed region. Our method learns intuitive and disentangled meta-handles in an unsupervised fashion, which factorize all the plausible deformations for the shape.

Abstract

*We propose **DeepMetaHandles**, a 3D conditional generative model based on mesh deformation. Given a collection of 3D meshes of a category and their deformation handles (control points), our method learns a set of meta-handles for each shape, which are represented as combinations of the given handles. The disentangled meta-handles factorize all the plausible deformations of the shape, while each of them corresponds to an intuitive deformation. A new deformation can then be generated by sampling the coefficients of the meta-handles in a specific range. We employ biharmonic coordinates as the deformation function, which can smoothly propagate the control points' translations to the entire mesh. To avoid learning zero deformation as meta-handles, we incorporate a target-fitting module which deforms the input mesh to match a random target. To enhance deformations' plausibility, we employ a soft-rasterizer-based discriminator that projects the meshes to a 2D space. Our experiments demonstrate the superiority of the generated deformations as well as the interpretability and consistency of the learned meta-handles. The code is available at* `https://github.com/Colin97/DeepMetaHandles`.

1. Introduction

3D Meshes can store sharp edges and smooth surfaces compactly. However, Learning to generate 3D meshes is much more challenging than 2D images due to the irregularity of mesh data structures and the difficulty in designing loss functions to measure geometrical and topological properties. For such reasons, to create new meshes, instead of generating a mesh from scratch, recent work assumes that the connectivity structure of geometries is known so that the creation space is restricted to changing the geometry without altering the structure. For example, [17, 16] create new shapes by *deformations* of one template mesh. They, however, limit the scope of the shape generation to possible variants of the template mesh. We thus propose a 3D *conditional* generative model that can take any existing mesh as input and produce its plausible variants. Our approach integrates a *target-driven* fitting component and a conditional generative model. At test time, it allows both deforming the input shape to fit the given target shape and exploring plausible variants of the input shape without a target.

Our main design goals are two-fold: improving the *plausibility* of the output shapes and enhancing the *interpretability* of the learned latent spaces. To achieve the goals, the key is to choose a suitable parameterization of deformations. One option is to follow the recent target-driven deformation network [18, 4, 20, 15], which parameterizes the deformation as new positions of all the mesh vertices. However, such a large degree of freedom often results in the loss of fine-grained geometric details and tends to cause undesirable distortions. Instead of following the above works, we leverage a classical idea in computational geometry, named *deformation handles*, to parameterize smooth deformations with a low degree of freedom. Specifically, we propose to take a small set of *control points* as deformation handles and utilize a deformation function defined on the control points and their *biharmonic coordinates* [19].

Not all the translations of the control points lead to plausible deformations. Based on the control-point handles, we aim to learn a low-dimensional deformation subspace for each shape, and we expect the structure of this subspace to exhibit *interpretability*. In contrast to typical generative models, where shape variations are embedded into a latent space implicitly, our method explicitly factorizes all the plausible deformations of a shape with a small number of interpretable deformation functions. Specifically, for each axis of our input-dependent latent space, we assign a deformation function defined with the given set of control points and offset vectors on them so that each axis corresponds to an intuitive deformation direction. Since each axis is explicitly linked to multiple control-point handles, we thus call them *meta-handles*. We enforce the network to learn *disentangled* meta-handles, in the sense that a meta-handle should not only leverage the correlations of the control-point handles, but also correspond to a group of parts that tend to deform altogether according to the dataset. We hope that the disentangled meta-handles allow us to deform each part group independently in downstream applications.

Beyond choosing the parameterization of deformations, we have to overcome the challenge of examining the plausibility. In the popular adversarial learning framework, a straightforward approach would be converting the output mesh to voxels or point clouds and exploiting voxel or point cloud based discriminators. The conversions, however, may discard some important geometric details. In our method, we instead project the shapes into a 2D space with a differentiable *soft rasterizer* [12] and employ a 2D discriminator. We found that this architecture can be trained more robustly, and it captures local details of plausible shapes.

Our deformation-based conditional generative model, named **DeepMetaHandles**, takes random pairs of source and target shapes as input during training. For the source shape, the control points are sampled from its mesh vertices by farthest point sampling, and the biharmonic cooridnates [19] for control-point handles are pre-computed. Our network consists of two main modules: MetaHandleNet and DeformNet. The MetaHandleNet first predicts a set of meta-handles for the source shape, where each meta-handle is represented as a combination of control-point offsets. A deformation range is also predicted for each meta-handle, describing the scope of plausible deformations along that direction. The learned meta-handles, together with the corresponding ranges, define a deformation subspace for the source shape. Then, DeformNet predicts coefficients multiplied to the meta-handles, within the predicted ranges, so that the source shape deformed with the coefficients can match the target shape. To ensure the plausibility of variations within the learned subspace, we then randomly sample coefficients within the predicted ranges and apply both geometric and adversarial regularizations to the corresponding deformations.

Fig. 1 shows examples of the learned meta-handles, which interestingly resemble natural deformations of *semantic* parts, such as lifting the armrests or bending the back of a chair. Our experiments also show that the learned meta-handles are consistent across various shapes and well disentangle the shape variation space. Finally, we compare our approach with other target-driven deformation techniques [5, 18, 4, 20] and demonstrate that our method produces superior fitting results.

Key contributions:

- We propose DeepMetaHandles, a 3D *conditional* generative model based on mesh deformation.
- We employ a few control points as deformation handles. Together with their biharmonic coordinates, we can produce smooth but flexible enough deformations.
- We propose to factorize the deformation space with a small number of disentangled meta-handles, each of which provides an intuitive deformation by leveraging the correlations between the control points.
- We improve the plausibility of the deformations by exploiting a differentiable renderer and a 2D discriminator.

2. Method

In this section, we will first briefly review the control-point-based deformation and the biharmonic coordinates [19] technique we use, and introduce how the meta-handles are defined with the control-point handles (Section 2.1). We will then present how we learn the meta-handles in an unsupervised fashion and our neural network architectures (Section 2.2). Lastly, we will introduce our loss functions that guide the emergence of plausible deformations and intuitive factorizations (Section 2.3).

2.1. Biharmonic Coordinates and Meta-Handles

Mesh deformation through directly moving individual vertex is cumbersome and may easily lead to unwanted distortions. We thus leverage deformation handles to parameterize the deformations with a low degree of freedom. The key in the handle-based deformation is to define a proper deformation function that features several desired properties. For instance, no change of handles should result in no deformation; each handle should produce local and smooth deformation; the deformation function should be expressed in closed form. Numerous previous work has introduced different handle-based deformation functions. Many of them are based on solving the *biharmonic* equation defined over the mesh with boundary constraints (given from handles).

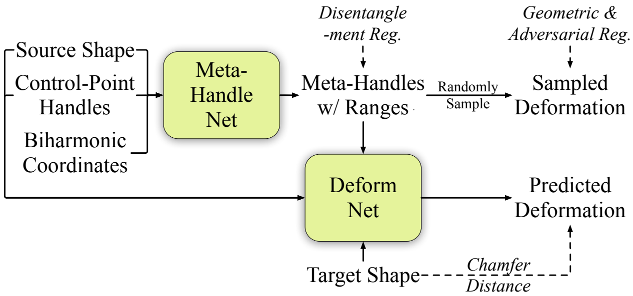

Figure 3: Overview of our method. We learn the meta-handles in an unsupervised fashion.

The resulting deformation functions of these approaches satisfy many desired properties [6, 8]. Also, closed-form expressions with respect to the handles can be easily calculated after a pre-computation. (Please refer to Jacobson *et al*. [7] for more details.)

In our method, we employ a subset of mesh vertices as the deformation handles (*control points*) and restrict the transformations of the handles to pure translations. Given the mesh vertices $\mathbf{V} \in \mathbb{R}^{n\times 3}$ (n vertices) and a set of c control points $\mathbf{C} \in \mathbb{R}^{c\times 3}$, the *linear* map $\mathbf{W} \in \mathbb{R}^{n\times c}$ between them ($\mathbf{V} = \mathbf{WC}$) is often called 'generalized barycentric coordinates' [14, 10, 9, 13]. Wang *et al*. [19] proposed one way to define $\mathbf{W}$ based on the biharmonic functions, which is thus dubbed *biharmonic coordinates*, and we utilize it as our deformation function. Without requiring that control points form a *cage* enclosing the input shape, our deformation handles are flexible and intuitive.

Specifically, we sample c control points from the mesh vertices by farthest point sampling (FPS) over the geodesic distances. The biharmonic coordinates $\mathbf{W}$ are also precomputed. However, the deformation function $f : \mathbb{R}^{c\times 3} \to \mathbb{R}^{n\times 3}$ defined over the given control points $\mathbf{C}$, $f(\mathbf{C}) = \mathbf{WC}$, has $3c$ degrees of freedom. It may overparameterize the plausible shape variation space, which means there may be lots of implausible deformations, if we randomly translate the control points (see Fig. 2). Also, there may exist strong correlations across the deformations from moving individual control points. For a specific shape (e.g., a chair), all the plausible variants may reside in a lower-dimensional subspace and can be factorized into several meaningful deformation directions (e.g., scaling all chair legs and bending the chair back).

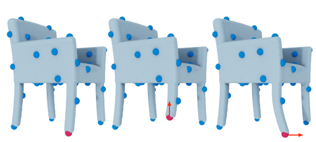

Figure 2: Two deformations resulted from moving the red control point along the arrow directions.

To this end, we propose to find a smaller number of meta-handles to factorize the subspace covering all the plausible deformations. Specifically, each meta-handle $\mathbf{M}_i \in \mathbb{R}^{c\times 3}$ is represented as *offsets* over the c control points:

$$\mathbf{M}_i = [\vec{t}_{i1}, \cdots, \vec{t}_{ic}]^T, \quad (1)$$

where $\vec{t}_{ij} \in \mathbb{R}^3$ indicates the offset of the j-th control point for the i-th meta-handle. In contrast to a single control point that mainly affects a local region of the mesh, each meta-handle is expected to provide a more intuitive deformation direction, which may even correspond to some semantic meanings (See Figs. 1).

We now use the linear combination of the meta-handles to represent a deformation. Specifically, a new deformation function $g : \mathbb{R}^m \to \mathbb{R}^{n\times 3}$ is defined with respect to the meta-handles $\{\mathbf{M}_i\}_{i=1\cdots m}$ and their linear combination coefficients $\mathbf{a} = [a_1, \cdots, a_m]$:

$$g\left(\mathbf{a}; \{\mathbf{M}_i\}_{i=1\cdots m}\right) = \mathbf{W}(\mathbf{C}_0 + \sum_{i=1}^{m} a_i \mathbf{M}_i), \quad (2)$$

where $\mathbf{C}_0 \in \mathbb{R}^{c\times 3}$ denotes the rest positions of the given control points. In the context of the conditional generative model, it can be interpreted as that each shape has a m-dimensional input-dependent latent space, where each axis corresponds to a meta-handle describing a specific deformation function in 3D space. A latent code $\mathbf{a}$ can thus be directly decoded to a deformation of the input mesh as a linear combination of the meta-handles.

Along with the meta-handles, our method also predicts *ranges* $\{[L_i, R_i]\}_{i=1\cdots m}$ of the coefficients associated with each meta-handle. The ranges describe the scope of plausible deformations along the direction of each meta-handle. Any set of coefficients within the *ranges* $\{[L_i, R_i]\}_{i=1\cdots m}$ is thus expected to produce a plausible deformation.

We utilize a small number of meta-handles to learn a low-dimensional compact deformation space. The degrees of freedom of the deformation function g is typically much smaller than that of the deformation function f, i.e., $m \ll 3c$. As a result, the meta-handles are required to not only leverage the correlations of the control-point handles, but also discover the underlying properties of the shape structure (e.g., chair legs are symmetric and should thus be deformed together).

2.2. Network Architecture

We propose to learn the meta-handles in an unsupervised fashion without taking semantic annotations or correspondences across the shapes as input or supervision. As shown in Fig. 3, our method mainly includes three networks: MetaHandleNet, DeformNet, and a discriminator network (discussed in Section 2.3). Taking a pair of randomly sampled shapes within the same category as input, the method predicts a deformation space for the source shape, and finds a deformation within the space to match the target shape. Specifically, MetaHandleNet takes a source shape, its control points, and the precomputed biharmonic coordinates as input and predicts a set of meta-handles as well as the corresponding coefficient ranges. DeformNet then predicts coefficients of the meta-handles so that the resulting deformation of the source shape matches the target shape.

2.3. Loss Functions

We consider three objectives when training our network: 1) the deformed input (source) shape matches the given target shape; 2) any deformation sampled from the learned ranges is plausible; 3) the learned meta-handles properly disentangle the deformation space. We thus train our network with the following joint loss function:

$$\mathcal{L} = \mathcal{L}_{fit} + \mathcal{L}_{geo} + \mathcal{L}_{adv} + \mathcal{L}_{disen}. \tag{3}$$

Among the four terms, the fitting loss $\mathcal{L}_{fit}$ corresponds to the first objective and minimizes the Chamfer distance [2] between the deformed source point cloud and the target point cloud.

$\mathcal{L}_{geo}$ and $\mathcal{L}_{adv}$ are geometry loss and adversarial loss, respectively, added for the second objective. In each iteration, we randomly sample a deformation within the predicted ranges, and apply these two losses to penalize implausible deformations.

Specifically, $\mathcal{L}_{geo}$ is further decomposed into:

$$\mathcal{L}_{geo} = \mathcal{L}_{symm} + \mathcal{L}_{nor} + \mathcal{L}_{Lap}, \tag{4}$$

where $\mathcal{L}_{symm}$ is symmetry loss minimizing the Chamfer distance [2] between the deformed point cloud and its reflection along the x-axis (also used in previous works [18, 20]). Given the mesh connectivity, normal loss $\mathcal{L}_{nor}$ and Laplacian loss $\mathcal{L}_{Lap}$ are computed to prevent distortions. $\mathcal{L}_{nor}$ minimizes the angle difference between the face normals of the source mesh and the deformed mesh. $\mathcal{L}_{Lap}$ minimizes $l1$-norm of the difference of Cotangent Laplacian.

It is not enough to guarantee plausible deformation with only geometric regularization. We thus leverage an adversarial loss $\mathcal{L}_{adv}$, which is defined with a soft rasterizer and a 2D discriminator. (A similar adversarial training idea using 2D projection is also introduced by Li *et al.* [11].) We feed both randomly deformed shapes and shapes without deformation into a soft rasterizer [12]. The renderer captures a soft silhouette image for each shape from a random view. The images are then fed into a simple 2D convolution neural network to predict whether they come from a deformed shape or not. The 2D discriminator network is jointly trained with MetaHandleNet and DeformNet with a classification loss function. The output probabilities for deformed shapes are used to penalize implausible deformations.

For the third objective, we introduce a disentanglement loss $\mathcal{L}_{disen}$. Inspired by Aumentado-Armstrong *et al.* [1], $\mathcal{L}_{disen}$ is defined with four terms:

$$\mathcal{L}_{disen} = \mathcal{L}_{sp} + \mathcal{L}_{cov} + \mathcal{L}_{ortho} + \mathcal{L}_{SVD}. \tag{5}$$

Specifically, $\mathcal{L}_{sp}$ encourages the meta-handles $\mathbf{M}_i$ and the coefficient vector $\mathbf{a}$ to be sparse by penalizing their $l1$-norm. $\mathcal{L}_{cov}$ penalizes the covariance matrix (calculated for each batch) of the coefficients $\mathbf{a}$. $\mathcal{L}_{ortho}$ encourages meta-handles to cover different parts of the control-point offsets by penalizing "dot products" between the meta-handles. $\mathcal{L}_{SVD}$ encourages the control points to translate in a single direction within each meta-handle. Please refer to the supplementary materials for the details of $\mathcal{L}_{disen}$.

3. Experiments

One of the main contributions of our method is that, for each shape, we learn a set of interpretable meta-handles with the corresponding coefficient ranges, which factorize all the plausible deformations for the shape.

Fig. 1 demonstrates some learned meta-handles of a single shape. Each column shows the deformations along the direction of a meta-handle, with the deformation scale uniformly sampled within the corresponding coefficient range. The red arrows highlight the deformation direction of each meta-handle. As shown in the figure, the learned meta-handles are disentangled and factorize all the plausible deformations for the shape. Although we do not take any semantic annotation or correspondences across different shapes as input or supervision, our method is able to learn some intuitive meta-handles. Specifically, the learned meta-handles are not limited to global scaling. Many of them align with some local semantic parts, such as adjusting the thickness of the chair seat (first column), the height of armrests (fourth column), the length of four chair legs (seventh column), and the height of the chair back (eighth column). Also, many of them involve non-rigid deformation of some parts, such as bending the chair back (fifth column) and two back legs (sixth column), which cannot be achieved through the rigid bounding-box handles proposed by previous methods [3, 15]. To construct a low-dimensional compact deformation space, the learned meta-handles not only leverage correlations between the control-point handles, but also discover the underlying hard constraints (e.g., symmetry) of the shape structure. Meanwhile, the coefficient ranges learn the underlying soft priors (e.g., ratios of part scales) and provide reasonable deformation scopes for meta-handles.

4. Conclusion

We presented **DeepMetaHandles**, a 3D conditional generative model based on mesh deformation. Our method takes automatically-generated control points with biharmonic coordinates as deformation handles, and learns a latent space of deformation for each input mesh. Each axis of the space is explicitly associated with multiple deformation handles, and it's thus called a meta-handle. The disentangled meta-handles factorize all the plausible deformations of the shape, while each of them conforms to an intuitive deformation. We learn the meta-handles unsupervisely by incorporating a target-driven deformation module. We also employ a differentiable render and a 2D discriminator to enhance the plausibility of the deformation.

References

[1] T. Aumentado-Armstrong, S. Tsogkas, A. Jepson, and S. Dickinson. Geometric disentanglement for generative latent shape models. In *ICCV*, 2019.

[2] Haoqiang Fan, Hao Su, and Leonidas Guibas. A point set generation network for 3d object reconstruction from a single image. In *CVPR*, 2017.

[3] Matheus Gadelha, Giorgio Gori, Duygu Ceylan, Radomir Mech, Nathan Carr, Tamy Boubekeur, Rui Wang, and Subhransu Maji. Learning generative models of shape handles. In *CVPR*, 2020.

[4] Thibault Groueix, Matthew Fisher, Vladimir G. Kim, Bryan C. Russell, and Mathieu Aubry. Deep self-supervised cycle-consistent deformation for few-shot shape segmentation. In *Eurographics Symposium on Geometry Processing*, 2019.

[5] Haibin Huang, Evangelos Kalogerakis, Siddhartha Chaudhuri, Duygu Ceylan, Vladimir G. Kim, and Ersin Yumer. Learning local shape descriptors from part correspondences with multiview convolutional networks. *ACM Transactions on Graphics*, 2017.

[6] Alec Jacobson, Ilya Baran, Jovan Popović, and Olga Sorkine. Bounded biharmonic weights for real-time deformation. In *ACM SIGGRAPH*, 2011.

[7] Alec Jacobson, Zhigang Deng, Ladislav Kavan, and JP Lewis. Skinning: Real-time shape deformation. In *ACM SIGGRAPH 2014 Courses*, 2014.

[8] Alec Jacobson, Tino Weinkauf, and Olga Sorkine. Smooth shape-aware functions with controlled extrema. In *Eurographics Symposium on Geometry Processing*, 2012.

[9] Pushkar Joshi, Mark Meyer, Tony DeRose, Brian Green, and Tom Sanocki. Harmonic coordinates for character articulation. In *ACM SIGGRAPH*, 2007.

[10] Tao Ju, Scott Schaefer, and Joe Warren. Mean value coordinates for closed triangular meshes. In *ACM SIGGRAPH*, 2005.

[11] Xiao Li, Yue Dong, Pieter Peers, and Xin Tong. Synthesizing 3d shapes from silhouette image collections using multi-projection generative adversarial networks. In *CVPR*, 2019.

[12] Shichen Liu, Tianye Li, Weikai Chen, and Hao Li. Soft rasterizer: A differentiable renderer for image-based 3d reasoning. In *ICCV*, 2019.

[13] Josiah Manson and Scott Schaefer. Moving least squares coordinates. In *Eurographics Symposium on Geometry Processing*, 2010.

[14] Mark Meyer, Alan Barr, Haeyoung Lee, and Mathieu Desbrun. Generalized barycentric coordinates on irregular polygons. *J. Graph. Tools*, 2002.

[15] Minhyuk Sung, Zhenyu Jiang, Panos Achlioptas, Niloy J. Mitra, and Leonidas J. Guibas. DeformSyncNet: Deformation transfer via synchronized shape deformation spaces, 2020.

[16] Qingyang Tan, Lin Gao, Yu-Kun Lai, Jie Yang, and Shihong Xia. Mesh-based autoencoders for localized deformation component analysis. In *AAAI*, 2018.

[17] Qingyang Tan, Lin Gao, Yu-Kun Lai, and Shihong Xia. Meshvae: Variational autoencoders for deforming 3d mesh models. In *CVPR*, 2018.

[18] Weiyue Wang, Duygu Ceylan, Radomir Mech, and Ulrich Neumann. 3dn: 3d deformation network. In *CVPR*, 2019.

[19] Yu Wang, Alec Jacobson, Jernej Barbič, and Ladislav Kavan. Linear subspace design for real-time shape deformation. In *ACM SIGGRAPH*, 2015.

[20] Wang Yifan, Noam Aigerman, Vladimir Kim, Siddhartha Chaudhuri, and Olga Sorkine-Hornung. Neural cages for detail-preserving 3d deformations. In *CVPR*, 2020.