Sketch based Modeling via Manifold Regularization

Juncheng Liu* Zhouhui Lian Jie Feng Bingfeng Zhou
Institute of Computer Science and Technology, Peking University

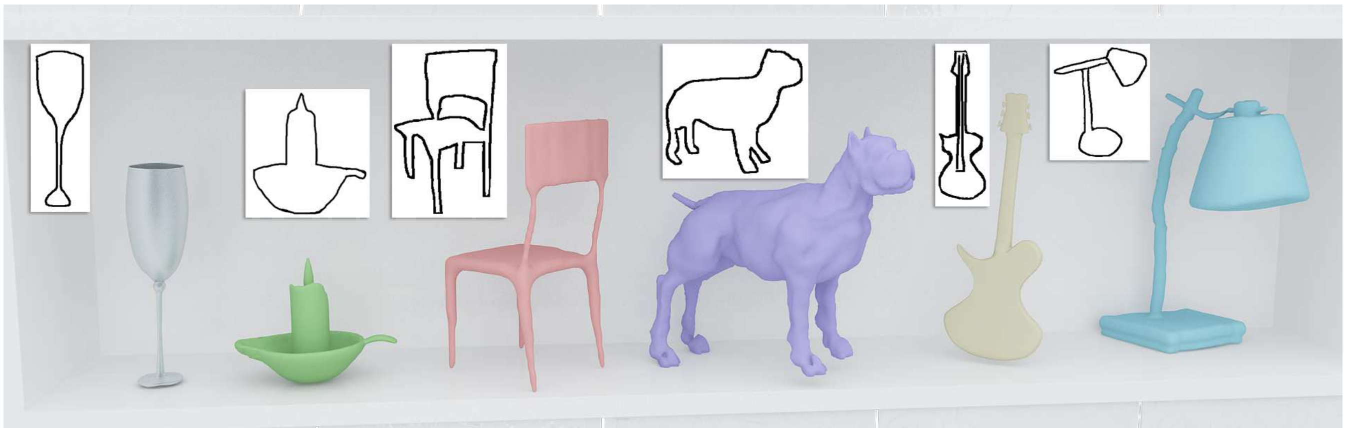


Figure 1: *Examples of results obtained using our sketch based shape modeling approach.*

Abstract

This paper proposes a novel method to automatically generate the realistic 3D model for a 2D free-hand sketch drawn by a user. Specifically, the proposed method first retrieves 3D shapes based on the sketch, and then implements a deformation procedure to make the most similar model retrieved more consistent with the provided sketch. In the retrieval stage, a locality preserving view selection scheme is adopted to generate views that are well-suited to create sketch images for the 3D object. Our method predicates the sketch views accurately while significantly reduces the amount of rendering views. In the deformation stage, retrieved models are modified according to the input sketches. Since free-hand sketches always contain various kinds of drawing errors such as stroke jittering and asymmetry, extracting plausible deforming information from sketches while discarding undesirable drawing errors is difficult. To address these issues, we obtain the plausible deformation for the corresponding sketch by exploring a shape manifold trained on a collection of similar 3D models. Experimental results show that the proposed method can generate 3D shapes that correspond quite well with 2D sketches.

CR Categories: I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Geometric algorithms;

Keywords: Sketch-based modeling, Shape manifold, User interface

*E-mail:{liujuncheng,lianzhouhui,feng_jie,cczbf}@pku.edu.cn. This work was supported by National Natural Science Foundation of China (Grant No.: 61472015, 61170206, 61370112 and 61202230), Beijing Natural Science Foundation (Grant No.: 4152022), and Key Laboratory of Machine Perception (Ministry of Education), Peking University.

SA'15 Technical Briefs, November 02 – 06, 2015, Kobe, Japan.
© 2015 ACM. ISBN 978-1-4503-3930-8/15/11…$15.00
DOI: http://dx.doi.org/10.1145/2820903.2820906

1 Introduction

With the rapid growth of large repositories of 3D models, effective ways of exploring and modeling 3D shapes are desired. As an intuitive way of searching and creating complex 3D shapes, sketch-based modeling has been studied for many years and a number of interactive tools have been designed for free-hand sketch designing [Yang et al. 2005; Lee and Funkhouser 2008; Xie et al. 2013; Fan et al. 2013]. Generally, existing approaches can be classified into the following two categories: 1) Methods that directly refer 3D geometric information from a well-drawn sketch; 2) Methods which retrieve the shape within a dataset that holds the best visual similarity with the provided sketch.

Geometry inferring from sketch. Directly inferring geometric information by exploring the sketch is an intuitive approach [Xu et al. 2014]. As a non-data driven method, this family of modeling methods often require realistic sketches that contain perspective and projective geometric information. For instance, [Mitani et al. 2002] used three vanishing points to find the projection center. Furthermore, they are often designed as progressively refined algorithms which are computationally expensive.

Sketch-based shape retrieval. Sketch based shape retrieval has been widely studied since the release of Princeton Shape Benchmark [Shilane et al. 2004]. Many efforts have been made to get more effective shape descriptors for sketches. In 2012, [Eitz et al. 2012] proposed a comprehensive framework for sketch based 3D shape retrieval that gives the state-of-the-art results. However, retrieval based methods typically require a large collection of shapes, while in many real applications usually none of models in the database is identical as the provided sketch.

To solve the problems mentioned above, in this paper, we present a framework that integrates the retrieval method with a sketch based deformation step to make the modeling results more consistent with the input sketch. The deformation step adopted in our approach has two advantages: 1) The shape manifold produces large numbers of variants by exploring a small collection of similar shapes (Figure 4). 2) The deformation step compensates the incorrect retrieval results to some extent.

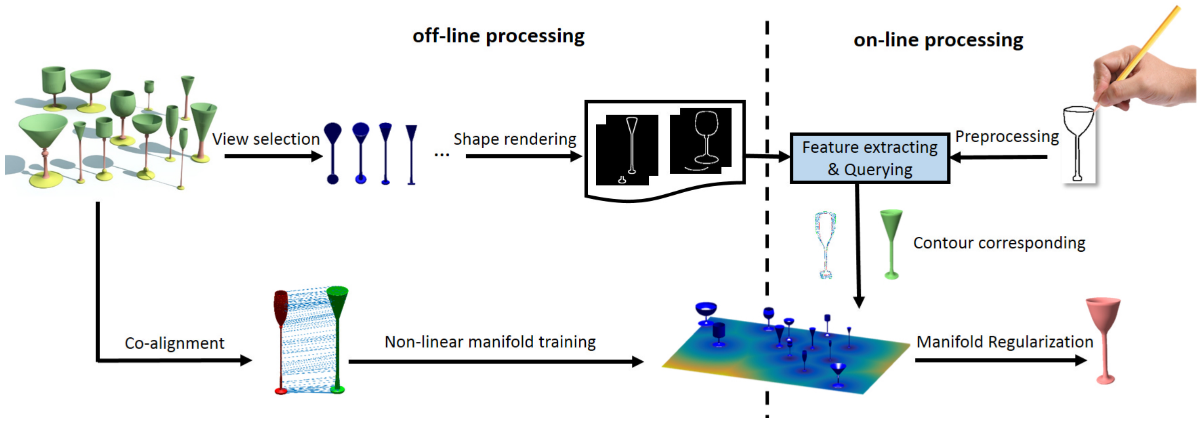


Figure 2: *Pipeline of our sketch-based modeling approach.*

More specifically, our approach consists of two major procedures: sketch based shape retrieval and deformation via manifold regularization. In the first step, we use the Histogram of Gradients (HOG) to describe 2D sketches and then perform sketch based shape retrieval. In the second step, we extract meaningful and plausible deformation information by exploring a nonlinear manifold trained on a collection of similar shapes using a non-linear dimensionality reduction technique GPLVM [Lawrence 2004]. The proposed method can be divided into off-line processing and on-line processing as shown in Figure 2. To sum up, the main contributions of this paper are threefold: 1) A novel sketch modeling framework is designed that integrates both shape retrieval and deformation; 2) A locality preserving view selection method is proposed; 3) Plausible deformation is extracted from sketch by employing manifold regularization.

2 Sketch based shape retrieval with locality preserving view selection

Contrasting to most of existing camera view sampling methods, our locality preserving view selection scheme employs an analytical method which preserves the locality details without sampling uniformly on the view direction sphere. Specifically, our method uses a particular dimensionality reduction method: locality preserving projections(LPP) [Niyogi 2004] which is more robust than approaches using conventional covariance analysis. LPP is suitable for view selection due to its linearity and locality preserving ability. As demonstrated in [Secord et al. 2011], surface visibility is considered to be the most influential metric when selecting a good viewing direction. Typically, we favor views that maximally reveal 3D shape information [Xu et al. 2014]. This implies the rationality of our view selection method. Moreover, the proposed method has a closed-form solution without invoking optimization in contrast to other machine learning based view selection schemes.

Given a set of discrete points X in mesh $\mathcal{M}$, $X = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N\}, \mathbf{x}_i \in \mathcal{M}$, $\mathcal{M}$ is a nonlinear manifold embedded in $\mathcal{R}^3$. We aim to find the optimized projection directions that best preserve the details. The projection matrix is given by collecting the first two largest eigenvalues and corresponding eigenvectors obtained by solving the generalized eigenvector problem

$$XLX^T\mathbf{a} = \lambda XDX^T\mathbf{a}\,,$$

where W is the weighted adjacency matrix and D is a diagonal matrix whose entries are column sums of W, $D_{ii} = \sum_j W_{ji}$. $L = D - W$ is the Laplacian matrix. Given the two projection direction vectors $\mathbf{a}_1$ and $\mathbf{a}_2$, the projection matrix is given by $A = [\mathbf{a}_1 \;\; \mathbf{a}_2]$. The projected point in $\mathcal{R}^2$ is denoted as $\mathbf{y}_i, i = 1, 2, \dots, N$. The explicit projecting transform is computed as

$$\mathbf{y}_i = A^T\mathbf{x}_i.$$

Instead of using vertices on the triangle mesh directly, we sample points uniformly within each triangle to avoid views that preserve too much information for high curvature regions while ignore flat regions. Given the locality preserving directions $\mathbf{a}_1$ and $\mathbf{a}_2$, the camera view direction is obtained by the cross product of these two vectors

$$\mathbf{v} = \mathbf{a}_1 \times \mathbf{a}_2\,. \tag{1}$$

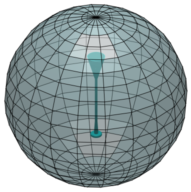
The camera coordinates system is thus determined by the three basis vectors $\mathbf{z}_c = \mathbf{v}, \mathbf{y}_c = \mathbf{z_c} \times \mathbf{u}, \mathbf{x}_c = \mathbf{y_c} \times \mathbf{z_c}$, where $\mathbf{u}$ represents the up vector.

To improve the freedom of the possible drawing views, we add a δ-view interval to the calculated locality preserving view. In spherical coordinate system, that is: $\boldsymbol{\theta} = \theta\mathbf{1} + \boldsymbol{\delta}_\theta$ and $\boldsymbol{\phi} = \phi\mathbf{1} + \boldsymbol{\delta}_\phi$, where $\mathbf{1}$ is a column vector with all its entries being one. $\boldsymbol{\delta}_\theta$ and $\boldsymbol{\delta}_\phi$ are incremental (or decremental) vectors indicating an interval of θ and ϕ in δ-view as illustrated in Figure 3. After rendering the selected views, we extracts HOG feature vectors from these view images. Note that after retrieval with a query sketch, both the most similar shape and its estimated view are obtained.

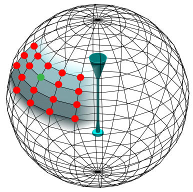
3 Sketch based deformation via manifold regularization

3.1 Shape collections co-alignment

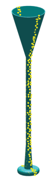
Methods that jointly optimize the maps among the shapes in a collection are considered to perform much better than approaches which directly implement matching on pairs of shapes [Huang and Guibas 2013]. Here we use a part-based template learning method [Kim et al. 2013] to co-align a collection of similar shapes. This algorithm starts with an initial template model and then jointly optimizes for part segmentation. Finally, 3D shapes can be well aligned in canonical domain. We use this as input and simply use ICP algorithm to obtain point-to-point correspondence.



(a) *uniformly sampled views*



(b) *locality preserving views with δ-views*



(c) *view slicing*

Figure 3: *Locality preserving views and uniformly sampled views. (a) uniformly sampled 400 rendering views. (b) locality preserving view(green dot) + δ-views(red dots), 12 in total. (c) view slicing, marked in yellow dots.*

3.2 Manifold regularization

To extract plausible deformation while discarding undesirable drawing errors, we adopt a manifold regularization method inspired by the success of recent work on font generation [Campbell and Kautz 2014]. After co-alignment, we are able to train a generative shape manifold by considering each shape as a vector in high-dimensional space. The trained manifold offers smooth interpolations between similar shapes and hence provides a valid constraint. Let Ω represent the pixels that have nonzero values in the rendered image of $\mathcal{M}$ in a given view and $\partial\Omega$ be the boundary of Ω by invoking edge detecting operator in Ω. The so-called view-slicing $\boldsymbol{q} \subset \mathcal{M}$ consists of vertices that contribute to the pixels in $\partial\Omega$ (marked as yellow dots in Figure 3c).

The first step of deformation is to align the sketch $\mathcal{S}$ with $\partial\Omega$. Here we employ a shape context descriptor for the correspondence recovery [Belongie et al. 2000]. Let $\mathcal{D}$ be the deformation vectors, namely, $\mathcal{D}_i = \mathcal{S}_i - \partial\Omega_i$. This procedure is illustrated in Figure 2. We apply these deformation vectors to the view slicing, that is, $\boldsymbol{q}_i \leftarrow \boldsymbol{q}_i + \mathcal{D}_i$.

Given the shape manifold trained through GPLVM, we aim to find a position $\mathbf{x}_p$ in the low dimensional space, whose corresponding vector $\boldsymbol{p}$ in high dimensional space favors the plausible shape deformation described by $\mathcal{D}$. Therefore, we exploit the shape manifold and aim to find a point $\mathbf{x}^*$ that minimizes the following energy function

$$E = \lambda_1 E_1 + \lambda_2 E_2 + \lambda_3 E_3 + \lambda_4 E_4, \tag{2}$$

where $\lambda_1, \lambda_2, \lambda_3, \lambda_4$ denote the corresponding regularization parameters. Terms in this function are discussed explicitly as follows.

The Euclidean distance term. Given the above definition, we are now able to give the first Euclidean distance term formulated as

$$E_1 = \sum_{i=1}^{m} \|\boldsymbol{q}_i - \boldsymbol{p}_i\|^2 = \|\boldsymbol{q} - \boldsymbol{p}\mathbf{S}\|_{\mathcal{F}}^2 \,, \tag{3}$$

where $\|\cdot\|_{\mathcal{F}}$ denotes the frobenius norm. $\mathbf{S} \in \{0,1\}^{n\times m}$ is a binary selective matrix with exact one entry being one within each column, i.e. $\mathbf{S}^T\mathbf{1} = \mathbf{1}$. $\mathbf{S}_{ij} = 1$ means that the i^{th} point in $\boldsymbol{p}$ is selected as the j^{th} point. Thus multiplying $\mathbf{S}$ is equivalent to apply the view slicing operation to the point set $\boldsymbol{p}$.

The local tangent vector term. To describe the contour, we introduce the second energy term named local tangent vector. We perform covariance analysis within a small neighborhood of each point $\boldsymbol{q}_i$ in $\boldsymbol{q}$. Note that view slicing points of a mesh do not necessarily lie in a plane. Therefore, in this step we project $\boldsymbol{q}$ to the estimated view plane beforehand. For simplicity, we will still use $\boldsymbol{q}$ to denote the projected $\boldsymbol{q}$. The second term is defined as

$$E_2 = \sum_{i=1}^{m} f(\boldsymbol{\mu}_i^q, \boldsymbol{\mu}_i^p) \,, \tag{4}$$

where $\boldsymbol{\mu}$ represents the local tangent vector. For a given point $\mathbf{x}_i \in \boldsymbol{q}$ and its neighborhood $\mathcal{N}(\mathbf{x}_i)$, $\mathbf{U\Sigma V}^T$ denotes the singular value decomposition (SVD) of $\mathcal{N}(\mathbf{x}_i)$, where $\mathbf{\Sigma} = diag(\sigma_1, \sigma_2)$ and $\mathbf{U} = [\boldsymbol{\mu}\ \boldsymbol{\nu}]$ are the two basis vectors in this local coordinate system. The function f evaluates the weighted cosine angular distance between two angles

$$f(\boldsymbol{v}_1, \boldsymbol{v}_2) = (1 - |\boldsymbol{v}_1^T \boldsymbol{v}_2|) e^{-\varepsilon \cdot \frac{\sigma_1}{\sigma_2}}.$$

The difference of Fourier shape descriptors. After projecting $\boldsymbol{p}\mathbf{S}$ and $\boldsymbol{q}$ into the view plane, we calculate the third term as

$$E_3 = \|\mathbf{F}_p - \mathbf{F}_q\|^2 \,, \tag{5}$$

where $\mathbf{F}_p$ and $\mathbf{F}_q$ are Fourier coefficients extracted from point sets $\boldsymbol{p}\mathbf{S}$ and $\boldsymbol{q}$, respectively.

The minimum distance regularization. As observed in Figure 4, the point that is too far away from training samples may result in undesired variants such as excessive stretch and distortion. Therefore, we add a minimum distance term to prevent the optimized point going too far away from training samples in the manifold

$$E_4 = \min_i \|\mathbf{x}_p - \mathbf{x}_i\|^2, \ \ \forall i = 1, 2, \ldots, N. \tag{6}$$

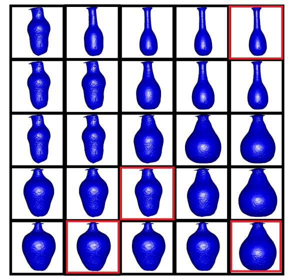


Figure 4: *A vase manifold generated by four samples (marked in red square). The dimensionality is reduced to 2 for visualization. From top to bottom, the vase body becomes wider. From left to right, the bulge position becomes lower. This manifold produces a large number of variants by exploring a small collection of similar shapes.*

4 Results

The proposed method is evaluated in the shape COSEG dataset[1] with 6 selected small subsets: candelabra, goblet, chair, lamp, guitar and fourlegs. Table 1 shows detailed configurations of each subset and the specific implementation time. Our final results are shown in Figure 5 from which we can see that important features such as symmetry and high-frequency details are well preserved.

For edge detection, we use sobel operator with a threshold chosen as 0.01. When rendering, we produce another image for recording the view slicing indices. Each pixel is assigned with an integer which denotes the indexing number of triangles that contribute to the pixel's brightness. The regularization parameters are chosen

[1] `http://web.siat.ac.cn/~yunhai/ssl/ssd.htm`

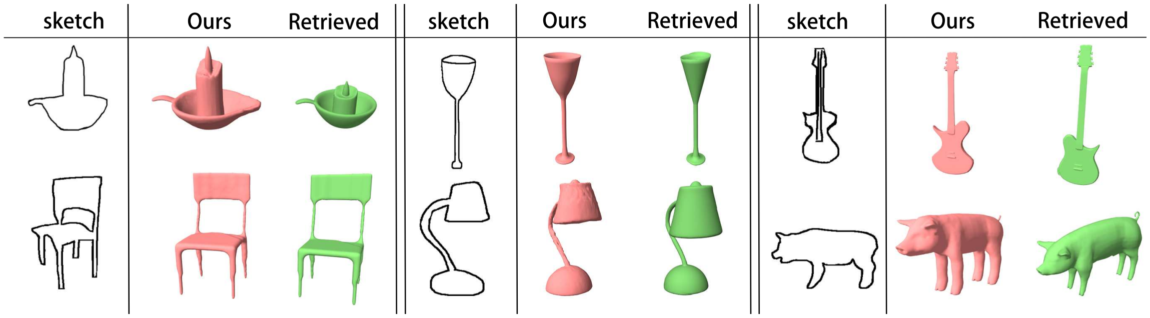


Figure 5: *Examples of sketch modeling results obtained by a normal retrieval system and our approach. The proposed method is evaluated in 6 datasets: candelabra, goblets, chairs, lamps, guitars and fourlegs.*

datasets	# shapes	#views	alignment(s)	total(s)
Goblets	12	11	35.13	46.88
Candelabra	20	11	34.85	48.23
Chairs	20	54	76.50	95.94
Lamps	20	15	52.93	110.75
Guitars	44	33	39.72	55.76
Fourleg	20	55	38.80	64.89

Table 1: *Details of datasets and the computational time of the proposed method. The context-based sketch alignment is the most time-consuming step.*

as $\lambda_1 = 2, \lambda_2 = 0.01, \lambda_3 = 0.01, \lambda_4 = 0.01$. For δ-view, we use different intervals for each subset due to the different geometric features of each dataset. For instance, goblets and candelabra are quite symmetric across the mean axis, so we stretch the interval of ϕ while shrink the interval of θ. Note that this is conducted for a further reduction of the amount of rendering images. We also observe that a large σ in GPLVM can lead to a more tremendous deformation among training samples. Therefore, we adopt $\sigma = 1$ in average to get smoother shape morphing. For the HOG feature, we utilize a window of 64×64 as sketches contain a majority of margins. It should also be point out that here the optimized solution of Equation 2 is found by searching for sampled points on the manifold.

5 Conclusion

This paper presented a sketch based modeling framework that integrates both shape retrieval and sketch based deformation. On contrast with other existing methods, projection realistic drawings are not required for our algorithm and a further consistency with the provided sketch is achieved by implementing a deformation step using manifold regularization. This paper also proposed an analytical view selection method by preserving the local details in shapes. Experimental results demonstrated the effectiveness of our method in sketch based shape modeling. One major limitation of the proposed method is that the shape manifold should be built on the dataset consisting of 3D models with similar shapes. We are planning to solve this problem in the future.

References

BELONGIE, S., MALIK, J., AND PUZICHA, J. 2000. Shape context: A new descriptor for shape matching and object recognition. In *NIPS*, vol. 2, 3.

CAMPBELL, N. D., AND KAUTZ, J. 2014. Learning a manifold of fonts. *ACM Transactions on Graphics (TOG) 33*, 4, 91.

EITZ, M., RICHTER, R., BOUBEKEUR, T., HILDEBRAND, K., AND ALEXA, M. 2012. Sketch-based shape retrieval. *ACM Trans. Graph. 31*, 4, 31.

FAN, L., WANG, R., XU, L., DENG, J., AND LIU, L. 2013. Modeling by drawing with shadow guidance. In *Computer Graphics Forum*, vol. 32, Wiley Online Library, 157–166.

HUANG, Q.-X., AND GUIBAS, L. 2013. Consistent shape maps via semidefinite programming. In *Computer Graphics Forum*, vol. 32, Wiley Online Library, 177–186.

KIM, V. G., LI, W., MITRA, N. J., CHAUDHURI, S., DIVERDI, S., AND FUNKHOUSER, T. 2013. Learning part-based templates from large collections of 3d shapes. *ACM Transactions on Graphics (TOG) 32*, 4, 70.

LAWRENCE, N. D. 2004. Gaussian process latent variable models for visualisation of high dimensional data. *Advances in neural information processing systems 16*, 3, 329–336.

LEE, J., AND FUNKHOUSER, T. A. 2008. Sketch-based search and composition of 3d models. In *SBM*, 97–104.

MITANI, J., SUZUKI, H., AND KIMURA, F. 2002. 3d sketch: sketch-based model reconstruction and rendering. In *From geometric modeling to shape modeling*. Springer, 85–98.

NIYOGI, X. 2004. Locality preserving projections. In *Neural information processing systems*, vol. 16, MIT, 153.

SECORD, A., LU, J., FINKELSTEIN, A., SINGH, M., AND NEALEN, A. 2011. Perceptual models of viewpoint preference. *ACM Transactions on Graphics (TOG) 30*, 5, 109.

SHILANE, P., MIN, P., KAZHDAN, M., AND FUNKHOUSER, T. 2004. The princeton shape benchmark. In *Shape modeling applications, 2004. Proceedings*, IEEE, 167–178.

XIE, X., XU, K., MITRA, N. J., COHEN-OR, D., GONG, W., SU, Q., AND CHEN, B. 2013. Sketch-to-design: Context-based part assembly. In *Computer Graphics Forum*, vol. 32, Wiley Online Library, 233–245.

XU, B., CHANG, W., SHEFFER, A., BOUSSEAU, A., MCCRAE, J., AND SINGH, K. 2014. True2form: 3d curve networks from 2d sketches via selective regularization. *ACM Transactions on Graphics 33*, 4.

YANG, C., SHARON, D., AND VAN DE PANNE, M. 2005. Sketch-based modeling of parameterized objects. In *EG Workshop on Sketch-Based Interfaces and Modeling*, 63–72.