Received: 2 July 2020 | Accepted: 15 July 2020

DOI: 10.1002/cav.1939

SPECIAL ISSUE PAPER

WILEY

Sketch-based modeling with a differentiable renderer

Nan Xiang[1] | **Ruibin Wang**[1] | **Tao Jiang**[2] | **Li Wang**[1] | **Yanran Li**[1] | **Xiaosong Yang**[1] | **Jianjun Zhang**[1]

[1]National Centre for Computer Animation, Bournemouth University, Dorset, UK

[2]Centre for Vision, Speech and Signal Processing, University of Surrey, Surrey, UK

Correspondence
Xiaosong Yang, National Centre for Computer Animation, Bournemouth University, Dorset, UK.
Email: xyang@bournemouth.ac.uk

Present address
Yanran Li, Bournemouth University, Fern Barrow, TA123, Tolpuddle Annex 2, Poole, Dorset, BH12 5BB, UK.

Abstract

Sketch-based modeling aims to recover three-dimensional (3D) shape from two-dimensional line drawings. However, due to the sparsity and ambiguity of the sketch, it is extremely challenging for computers to interpret line drawings of physical objects. Most conventional systems are restricted to specific scenarios such as recovering for specific shapes, which are not conducive to generalize. Recent progress of deep learning methods have sparked new ideas for solving computer vision and pattern recognition issues. In this work, we present an end-to-end learning framework to predict 3D shape from line drawings. Our approach is based on a two-steps strategy, it converts the sketch image to its normal image, then recover the 3D shape subsequently. A differentiable renderer is proposed and incorporated into this framework, it allows the integration of the rendering pipeline with neural networks. Experimental results show our method outperforms the state-of-art, which demonstrates that our framework is able to cope with the challenges in single sketch-based 3D shape modeling.

KEYWORDS

deep learning, shape prediction, sketch-based modeling

1 | INTRODUCTION

Sketching is an efficient and intuitive way of graphically demonstrating ideas. It plays an important role in areas of artistic creation, product engineering and industrial design due to its succinctness and efficiency. However, there is a huge gap between sketch and the product with concrete three-dimensional (3D) shape. Bringing two-dimensional (2D) sketch to the 3D world is the goal of sketch-based 3D shape prediction. This stimulating topic has been widely discussed in computer vision and pattern recognition area for many years.

Humans are good at perceiving 3D shapes and spatial positions from 2D sketches via prior knowledge, while it is a challenging task for computers. *"How can computers understand and interpret sketches in three dimensions?"*[1] is the question that computer scientists have been pondering for decades. Many researches define extra rules to get adequate information for converting 2D sketch to 3D model,[2-5] but these methods are extremely restricted to specific shapes and preconditions, the shape recovering became much more cumbersome when there are too much irregular lines exist.[6] Recovering a complete 3D shape from the sketch is a problem remains unsolved,[1] especially when the recovering is based on a single sketch image on account of the multitude of ambiguities in single-view line drawings.

Recent progress in deep neural networks has sparked a growing research interest in using deep learning methods for image-based 3D shape reconstruction.[7-15] Indeed, a few recent works explored the potential of learning from 3D model priors for predicting the 3D shapes of sketches,[16-19] but the issues are still there, such as the output models lack sharp features,[17,18] require multiview for refining[17,19] or need category-specific 3D templates for training[16] which can be seen in Figure 1. In this paper, we consider the problem of 3D shape prediction from a single sketch image. To address the problems mentioned above, an end-to-end learning framework with a differentiable renderer is presented. Figure 2 shows partial experimental results of our approach.

Our approach recovers 3D mesh for a single sketch without any 3D supervision by introducing a differentiable renderer. Renderer is an engineered program that projects the 3D models onto the 2D screen and then generates shaded images via *rasterization*,[20] this process is also known as Rendering. Literally, recovering a 3D shape from a single image can be seen as an inverse process of the rendering. Unfortunately, the rendering is not invertible due to the loss of vital data like 3D spatial information. Deep learning methods can be potentially used to cope with this difficulty, the key obstacle is the rasterization which is a discrete operation, while neural networks rely upon back-propagation by gradient descent to update weights, which means the gradients cannot be back-propagated in the rendering process.

Several differentiable renderers have been proposed and applied to 3D shape prediction.[7-9] However, these methods are focus on shaded image-based modeling, cannot handle that of sketch image. Kato et al.[8] introduce a hand-crafted linear interpolation manner for calculating the approximated gradients, it only considers gradients on image plane so that the predicted 3D shapes lack concave surface features. Liu et al.[7] and Chen et al.[9] claim that they achieve a fully differentiable rendering pipeline by introducing a probabilistic rasterization, while it brings a higher computational cost. Although the prior differentiable renderers have achieved promising results in 3D shape recovering for the single shaded image, they cannot be applied to sketch-based 3D shape prediction due to the sparsity and irregularity of the sketch.

In this work, we incorporate a differentiable renderer into a deep learning framework for 3D shape prediction from a single sketch image. Inspired by the work of normal map generation in References 21,22, we use Conditional Generative Adversarial Networks (CGANs) architecture[23] to train a normal image generator for the sketch. The generated normal image is then transferred to encoder-decoder convolutional neural networks (CNNs) for 3D shape prediction. The learning process requires 2D supervision exclusively by employing the differentiable renderer. The overview of the system is shown in Figure 3. The normal image contains both silhouette and surface geometric information of the 3D mesh, thus the utilization of normal image allows us to recover the complete 3D shape by the 2D supervision. To demonstrate the advantages of our approach, we compare with the state-of-art in both sketch and shaded image based 3D shape prediction methods, the results are shown in Section 4.2.

Our main contributions can be summarized as:

FIGURE 1 State-of-art of sketch-based modeling methods. (a) Delanoy et al.[17] and (b) Lun et al.[19] use multiview sketches to refine the three-dimensional (3D) shape. (c) The method of Smirnov et al.[16] requires category-specific 3D templates that extremely restricts its universality. (d) Pix2Vox[18] shows a result lacking sharp features, though it can recover 3D shape from a single sketch

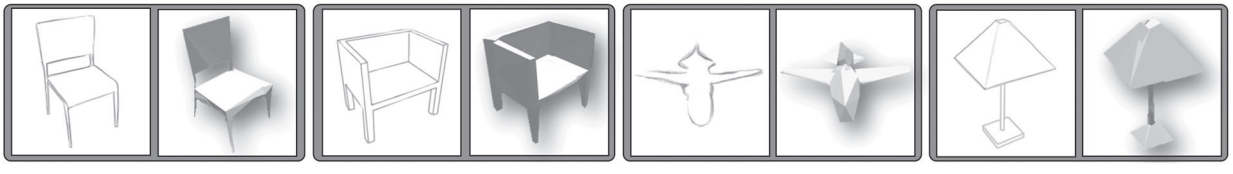


FIGURE 2 The results of our method. Given a single-hand drawing sketch, even it has broken and irregular lines, our framework generates a complete three-dimensional shape

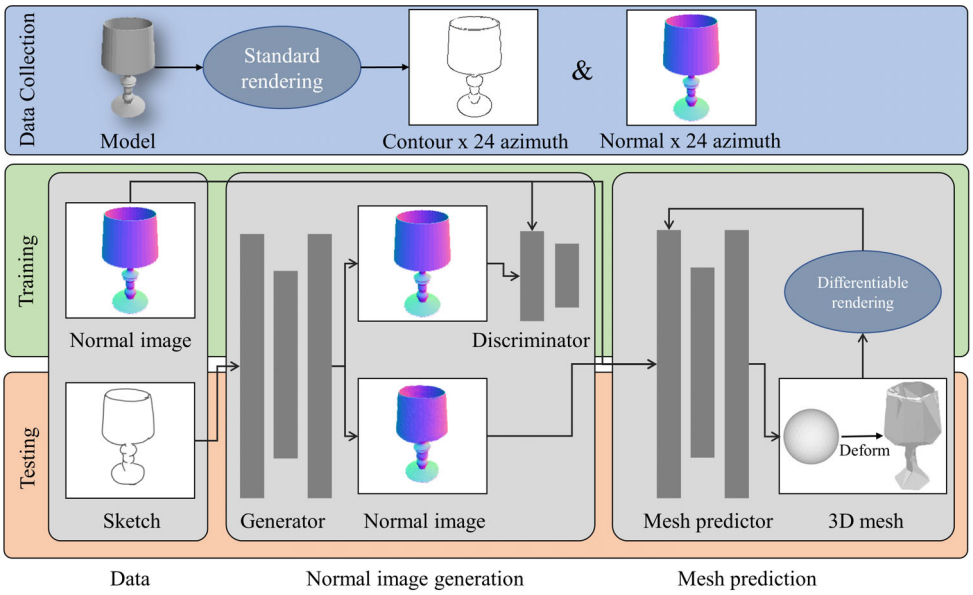


FIGURE 3 The overview of the framework. It consists of a Conditional Generative Adversarial Networks architecture for normal image generation, and an encoder-decoder convolutional neural networks architecture for mesh prediction. The training data includes sketch images and normal images, they are rendered from three-dimensional models, under 24 azimuth angles for each. Data collection detail is demonstrated in Section 4.1

- We present an end-to-end learning framework to recover 3D shape from a single sketch image. A differentiable renderer is proposed and incorporated into the framework, it provides approximate gradients for rendering process, which allows the integration of the rendering pipeline with neural networks.
- We introduce a novel two-step strategy for single sketch 3D modeling. Instead of directly recovering 3D shape from line drawings, we consider the problem as normal image generation and normal image based shape prediction. The generated normal image provides both silhouette and surface geometric information, to some extent, it facilitates the disentanglement of ambiguities in the sketch.

2 | RELATED WORK

2.1 | Sketch-based 3D modeling

Recovering 3D shape from 2D line drawings has been an active research area for more than two decades.[1,2] The previous work can be broadly categorized into two types in terms of whether it is learning-based approach.

Traditional *constrained-based* methods require specific shape features that can be achieved by defining extra rules, to get adequate information for converting 2D sketches to 3D model. Malik and Jitendra[2] introduced line labeling rules to classify the 2D lines, then interpreted 3D information such as depth, position and orientation of the line drawing scenes. Based on the line labelling rules, Malik et al.[3] presented a framework to partition the global constraints into constraint sets corresponding to the faces, edges, and vertices for easier optimization was presented. Shao et al.[4] estimated the gemoetric information of surface according to the cross-sections of the sketches. Then Cordier et al.[24] described a system for inferring the 3D shape for mirror-symmetric curves. These methods are commonly restricted to specific shapes and clean curves. *Incremental-based* methods allow users to interactively add new strokes for dynamically reducing ambiguities in the reconstruction process.[25-28] These kind of systems require extra user guidance to obtain supplementary curves from one or multiple views, which impedes their spread and application. In contrast, we aim to recover the 3D shape from a single freehand sketch without extra annotation or modification.

This work is motivated by the recent progress of deep learning methods in solving computer vision issues specifically on sketch-based modeling. Delanoy et al.[17] introduced an interactive CNNs-based reconstruction engine that can refine a voxel model by multiview sketch inputs, a post-process that converts the voxel to the polygon mesh was brought into the pipeline. Lun et al.[19] trained CNNs for generating normal images from sketches, then fused the multiview normal images as 3D point clouds followed with a mesh converting process. The interactive system Pix2Vox[18] provided a graphical interface for users to generate the 3D voxel shape in real-time, the shape was updating with respect to the incremental changes of the input sketches. Polygon mesh is a more popular 3D representation compared to other types such as voxel

and point clouds,[8,20,21] while because of the peculiar data structure of the mesh, the prior works cannot model a polygon mesh from sketches directly using neural networks, they introduced a postprocess to convert the intermediate 3D representations into mesh instead.[17,19] In addition, Smirnov et al.[16] learned a special shape representation, a deformable parametric template composed of Coons patches.[29] Though Smirnov et al. captured the piecewise smooth geometry of shapes, category-specific 3D templates were required which extremely limits its generalization. Our approach recovers the single sketch image to its 3D polygon mesh immediately without category-specific templates, the results are competitive even surpasses the state-of-art.

2.2 | Differentiable rendering-based 3D shape prediction

Recently, a number of works were dedicated to predicting 3D polygon mesh for single shaded image by introducing differentiable rendering pipeline.[7-9] The core technique of the differentiable rendering is to find a way to change the standard discrete rasterization into a continuous manner, which allows both forward and backward propagation.

OpenDR,[30] known as the first differentiable rasterization-based general-purpose renderer, approximated gradients of the projected pixels using first-order Taylor expansion. Neural (3D) Mesh Renderer (NMR)[8] hand-designed a linear interpolation-based scheme for gradients approximation, which was applied to the single image 3D mesh reconstruction. Both OpenDR and NMR followed the standard rendering pipeline in forward pass, and their approximated gradients were operated on the 2D image domain. Then Liu et al.[7] introduced a probabilistic formulation that treated the rendering as a probabilistic process, where every pixel was assigned to all faces. Subsequently, Chen et al.[9] proposed to specify the foreground pixel to the most front faces, such that it can alleviate the highly computational cost induced by the probabilistic rendering. Although the different differentiable renderer has been employed for single shaded image 3D shape prediction.[7-9] It has not yet been applied for single sketch image. Our approach converts the sketch to normal image first, then use the single normal image to generate 3D shape, these two steps are unified in an end-to-end learning framework with a differentiable renderer.

3 | METHOD

3.1 | Differentiable rendering pipeline

Standard rendering pipeline. Rendering pipeline is the process of drawing the 3D model into what the computer monitor displays, the popular graphics application programming interfaces such as Direct3D and OpenGL provide unify workflow for modern rendering pipeline.

We consider the rendering as starting from 3D vertex to 2D shaded plane. As shown in Figure 4, *vertex shader* takes vertex data as input, transforms it into normalized device coordinates.[20] *Shape assembly*stage assembles all the points in the specific shape primitive, triangle for example. *Geometry shader* is an optional shader, has ability to generate new vertices for updating the shape. *Rasterization* is the central operation in the pipeline, it enumerates the pixels that are covered by the shape primitive,[20] the output is a set of fragments that will be transferred to *fragment shader*, to calculate the final color of each pixel.

In this work, the geometry shader follows default settings, the vertex and fragment shaders are easily defined in entirely differentiable manners. However, the rasterization is not differentiable due to the discrete sampling operation. In the following, a differentiable rasterization formulation is demonstrated.

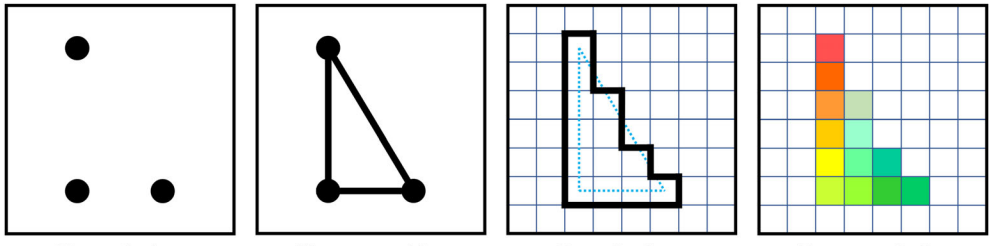


FIGURE 4 The standard rendering pipeline

Our rasterization formulation. Inspired by Reference 8, let $A_i(x_i,y_i)$ be a single pixel of an image, its color is denoted as I_i, then the gradient can be represented as $(\frac{\partial I_i}{\partial x_i}, \frac{\partial I_i}{\partial y_i})$. Assume that pixel A_i is outside the projected face f_j, when f_j move to collide with A_i, its color changes to $\overline{I_i}$. In standard discrete rasterization, the gradient is zero even the face is moved to cover the pixel A_i due to the sudden change of the color, which is not a continuous process.

Here we denotes $D : d(A_i, f_j)$ as the differentials between A_i and f_j along with the x and y coordinates, such that we define the derivate of I_i as:

$$\frac{\partial I_i}{\partial A_i} = \frac{\overline{I_i} - I_i}{\delta D}. \tag{1}$$

δ is a parameter that controls the strength of the gradients.

3.2 | Mesh prediction

The first step is to convert sketch to normal image, which facilitates the disentanglement of ambiguities in the sketch. This process will be demonstrated in Section 3.3. On this basis, the second step is to generate 3D shape from the normal image.

Previous works have proved that shaded image based mesh prediction can be realized without 3D supervision by incorporating the differentiable rendering pipeline.[7-9,21] Inspired by these series of work, we leverage the idea of generating 3D mesh by deforming a predefined sphere mesh, with the *topological genus 0*, rather than category-specific 3D templates as used in Reference 16. It can deform to the shape with the same genus level. The deformation process is formulated as $v_i + \triangle v_i^l + \triangle v^g$, where v_i is the vertex of the mesh, $\triangle v_i^l$ is the local bias for each vertex, and $\triangle v^g$ is a global bias. These two bias vectors are the outputs of the mesh predictor. Following losses are used for supervising the reconstruction networks:

In each iteration of the training process, surface normals of the mesh are calculated, and mapped into RGB range [0,1], then render the values into a normal image $\hat{N}$ by our differentiable renderer, the ground-truth of the normal image is N. In consideration of the normal image contains both 2D silhouette information and 3D mesh surface details, and L1 distance keeps more sharpen features than L2 distance,[31] we calculate the L1 distance between $\hat{N}$ and N as the normal loss:

$$\mathcal{L}_n = ||\hat{N} - N||_1, \tag{2}$$

Then let $\hat{S}$ and S, respectively, denote the predicted and ground-truth silhouette. We use the Intersection-Over-Union (IOU)[32] as the silhouette loss, it is defined as:

$$\mathcal{L}_s = 1 - \frac{||\hat{S} \otimes S||_1}{||\hat{S} + S - \hat{S} \otimes S||_1}, \tag{3}$$

where the symbol $\otimes$ presents an element-wise product.

In addition, the As-Rigid-As-Possible energy is adopted as the edge loss to regularize the edges:

$$\mathcal{L}_e = \frac{1}{n} \sum_{e_i \in E} ||\hat{e}_i - e_i||_2, \tag{4}$$

where e_i denotes one original edge in the edge set E of the mesh, while $\hat{e}_i$ is the current edge corresponding to e_i, and n is the amount of edges.

A smoothness loss[8,9,21] is also employed, it acts on the predicted mesh directly and ensures the consistency of the surface:

$$\mathcal{L}_m = \sum_{(f_i,f_j) \in F} (1 + \cos < f_i, f_j >)^2, \tag{5}$$

Here $<f_i, f_j>$ is the dihedral angle of two adjacent faces, and F denotes the set of all adjacent face pairs.

The final loss for the mesh prediction is a weighted sum of above losses:

$$\mathcal{L} = \lambda_n \mathcal{L}_n + \lambda_s \mathcal{L}_s + \lambda_e \mathcal{L}_e + \lambda_m \mathcal{L}_m. \tag{6}$$

3.3 | Normal image generation

Isola et al.[33] explored image-to-image translation problem using CGANs.[23] On this basis, Su et al.[22] trained a normal image generator that be able to convert sketch images to normal images. Inspired by these works, we treat the normal image generation for a single sketch image as the image-to-image translation. We use CGANs mix a global L1 distance and a local sharp feature sampling regularizer to optimize the normal image generator.

The CGANs architecture is a variant model of GANs,[34] GANs consist of a generator G and a discriminator D, G maps a random vector z to an image y, $G : z \to y$. Based on GANs, CGANs conditions on extra information x,[23] in this work, x is the sketch image. The standard objective function of CGANs[21,22,33] is defined as:

$$\mathcal{L}_{\text{CGANs}}(G, D) = \mathbb{E}_{x,y}[\log D(x, y)] + \mathbb{E}_{x,y}[\log (1 - D(x, G(x, z)))], \tag{7}$$

where G and D represent the Generator and Discriminator, respectively, x is the input image and y is the normal image, z is the random vector. A qualified Generator can be represented as: $\hat{G} = \arg\min_G \max_D \mathcal{L}_{\text{CGANs}}(G, D)$.

We calculate the distance between the generated normal image and the ground-truth to regularize the global image distribution. Compare to the L1 norm, L2 norm encourages image blurring,[31] hence, the global distance is:

$$\mathcal{L}_g(G) = \mathbb{E}_{x,y,z}[||G(x, z) - y||_1], \tag{8}$$

Additionally, in each iteration of training, we sample a few pixels from the areas with sharp geometric features such as corners and edges via applying a Sobel Filter to the normal image, to enhance the constraints of local features:

$$\mathcal{L}_l(G) = \mathbb{E}_{\hat{\varphi} \subset G(x,z), \varphi \subset y}[||\hat{\varphi} - \varphi||_1], \tag{9}$$

where $\hat{\varphi}$ is the sampled pixels from the generated normal image, φ is the corresponding pixels of the ground truth.

The normal image generator g is obtained by optimizing the final objective:

$$g = \arg\min_G \max_D \mathcal{L}_{\text{CGANs}}(G, D) + \lambda_g \mathcal{L}_g(G) + \lambda_l \mathcal{L}_l(G). \tag{10}$$

4 | EXPERIMENTS

4.1 | Experimental setup

Data preparation. In this work, mass of sketches are required for training the networks, however, there is no large-scale database of hand-drawn sketches with matched 3D meshes available. Thus, we generate synthetic training data from *ShapeNet*,[35] a popular opensouce 3D dataset which is used in prior works.[7-9,16,21] Our approach generates both sketch and normal images simultaneously, the overview is shown in Figure 5.

We build a standard rendering pipeline to draw the normal and suggestive contour[36] images under 24 azimuth angles with 30-degree elevation angle for each model. To imitate the ambiguities presented in hand-drawn sketches, we employ a synthetical method[16] to augument the contours with features such as broken lines as shown in Figure 5. The resolution of the images is 256×256.

Network settings. The overview of the system is shown in Figure 3. Specifically, to validate the performance of our approach, the shape predictor is an encoder-decoder architecture that identical to that of.[7-10] In the training stage, we first down-sample the normal images to 64×64, then feed the resized images to the network. A predefined sphere with 642 vertices was used as the underlying mesh, which is similar to that of previous works.[7-9] We set the weights with $\lambda_n = 0.002$, $\lambda_s = 0.9$, $\lambda_e = 0.9$, $\lambda_m = 0.01$. The Adam optimizer with $\alpha = 0.0001$, $\beta_1 = 0.9$ and $\beta_2 = 0.999$. For each category, we set the batch size to 64, iteration to 20,000.

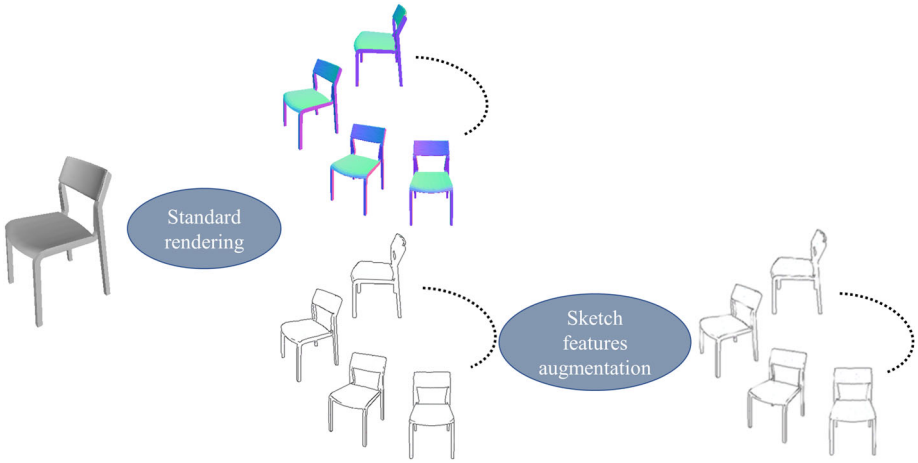


FIGURE 5 Training data preparation. Render the normal and contour images from 24 azimuth angles for each mesh. Then augument the hand-drawn features for the contour images

For the normal image generator, the CGAN architecture is a common choice for image-to-image translation,[22,23,33] we follow the structure in Reference 22. In every training iteration, we feed 256×256 pixels synthetic sketch image to the Generator, the ground-truth and the generated intermediate normal images are transferred to the Discriminator. We set the learning rate of the RMSProp optimizer to 5e-5.

4.2 | Results and comparisons

Our method achieves encouraging results that can be seen in Figure 6. To verify the superiority of the proposed approach, we also compare results with that of state-of-art both in learning sketch-based modeling[16-18] and shaded image-based modeling.[8,10]

Compared to previous single-view approaches. We compare our method to other single sketch-based methods.[16,18] The voxel-based method[18] can recover the 3D voxel for the given single sketch image, but the result is mediocrity as shown in Figure 7, while our method shows a more promising result.

Another state-of-art approach[16] produces a parametric 3D model (Coons patches), but it is restricted to specific shapes, due to the factor that it requires different templates for each category, in contrast, our approach uses only one template mesh for all categories of shapes. Furthermore, our results show more sharp and accurate features than that of Smirnov et al.,[16] the comparison result can be seen in Figure 8.

Compared to the multiview approaches. In Figure 9, we compare with multiview-based methods. In Reference 17, an interactive system is introduced to obtain multiview sketches, they utilize the incremental sketches to refine the 3D voxel shape. Lun et al.[19] takes multiview sketches as input to generate dense point cloud, both References 17 and 19 unable to generate 3D mesh but introducing a postprocess to convert voxel[17] or point cloud[19] to mesh instead. Our approach takes only single sketch as the input, and generates promising 3D mesh online without any post process.

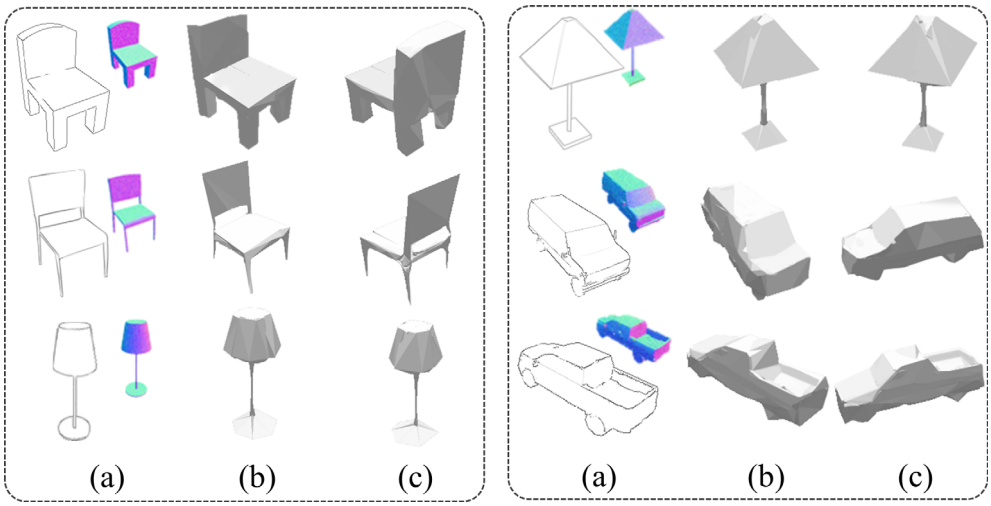


FIGURE 6 Some experimental results. (a) is the input sketch, and the generated normal image. (b) and (c) are the generated three-dimensional mesh rendered from two perspectives. The normal image provides geometric surface information that is beneficial to the concave features recovering

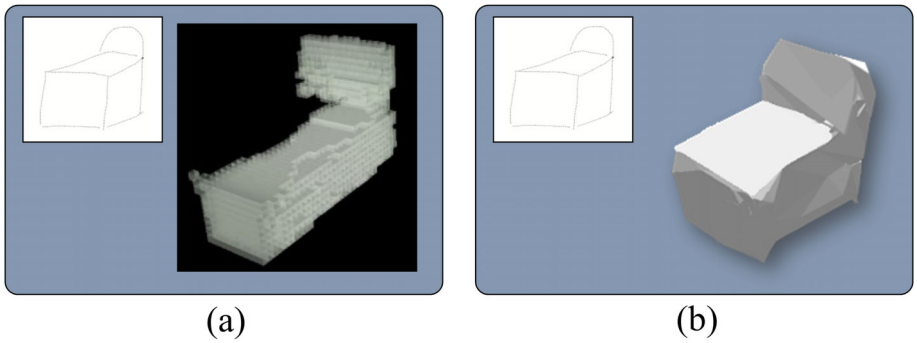


FIGURE 7 Compared to the voxe-based approach. (a) is the voxel-based method[18] and (b) is our method

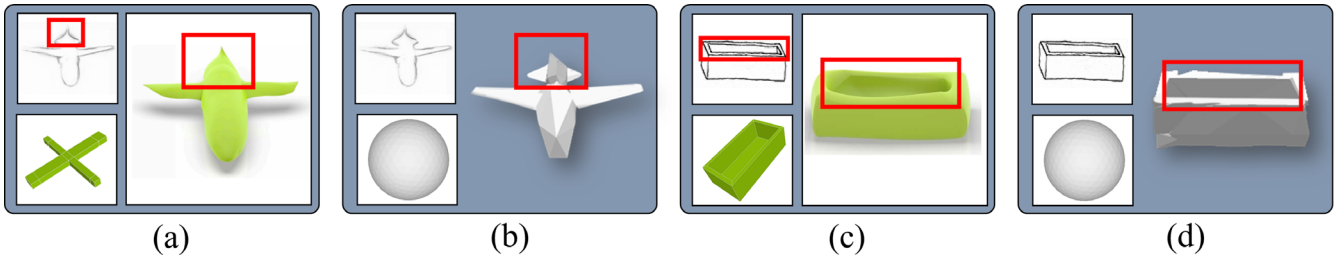


FIGURE 8 Compared to the previous single-view approach. Though the method of Smirnov et al.[16] (a and c) using single sketch as input while specific shapes of template are required for different categories. Our approach (b and d) only needs a sphere as the template for all categories, and recovers more sharp and accurate features as the areas marked by the red boxes above

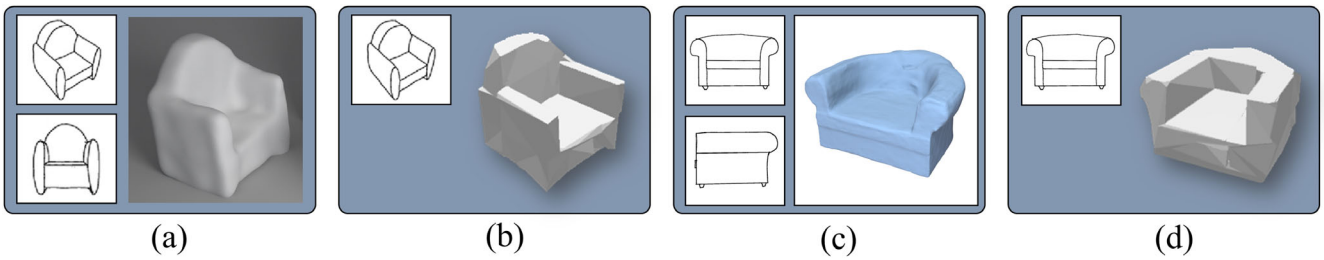


FIGURE 9 Compared to the multi-view approaches. (a) Reference 17 and (c) Reference 19 require multiview sketches to refine the shape, and a post-process to convert voxel[17] or point cloud[19] to mesh. Our approach (b and d) generates the mesh with promising shape directly from a single sketch

Compared to single shaded image-based reconstruction approaches. To demonstrate the shape accuracy of our results, we compare with the 3D unsupervised reconstruction methods. These methods take the image with completely filled contour as input, generate 3D shape in voxel or mesh representation. We use IOU,[32] the most commonly metric for shapes' similarity, to evaluate the comparison result. As shown in Table 1, our approach even achieves better IOU scores with the single sketch image as input. Meanwhile, the result of comparison between ours and NMR[8] is shown in Figure 10, it illustrates that the normal image-based constrain allows our method to recover more accurate and detailed surface structures than that of silhouette based method.

TABLE 1 Comparison of Intersection-Over-Union scores (the higher the better)

Category	Airplane	Bench	Dresser	Car	Chair	Display	Lamp	Speaker	Rifle	Sofa	Table	Phone	Vessel	Mean
Retrieval[10]	0.5564	0.4875	0.5713	0.6519	0.3512	0.3958	0.2905	0.4600	0.5133	0.5314	0.3097	0.6696	0.4078	0.4766
Voxel[10]	0.5556	0.4924	0.6823	0.7123	0.4494	0.5395	0.4223	0.5868	0.5987	0.6221	**0.4938**	0.7504	0.5507	0.5736
NMR *et al.*[8]	**0.6172**	0.4998	0.7143	0.7095	0.4990	0.5831	0.4126	0.6536	0.6322	0.6735	0.4829	0.7777	0.5645	0.6016
Our	0.6025	**0.5132**	**0.7212**	**0.7436**	**0.5841**	**0.5852**	**0.4804**	**0.6700**	**0.6521**	**0.6944**	0.4658	**0.7876**	**0.6037**	**0.6233**

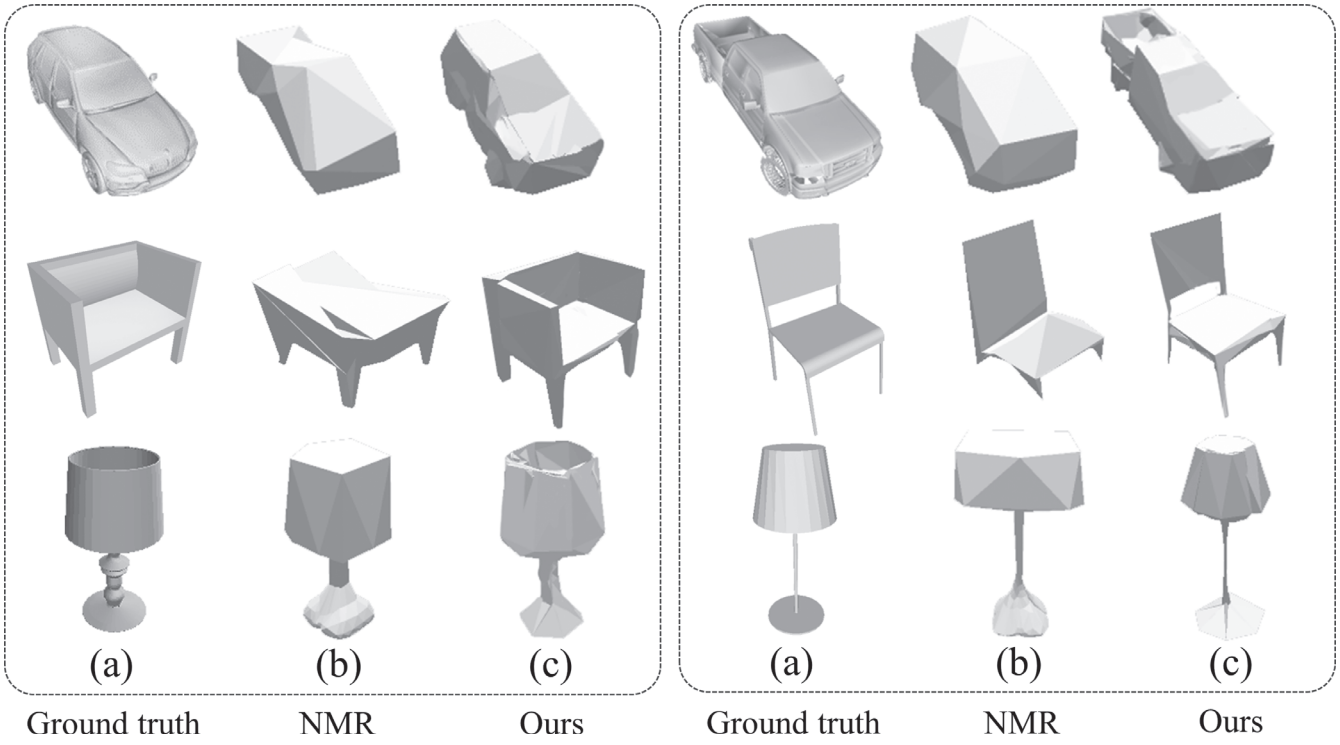
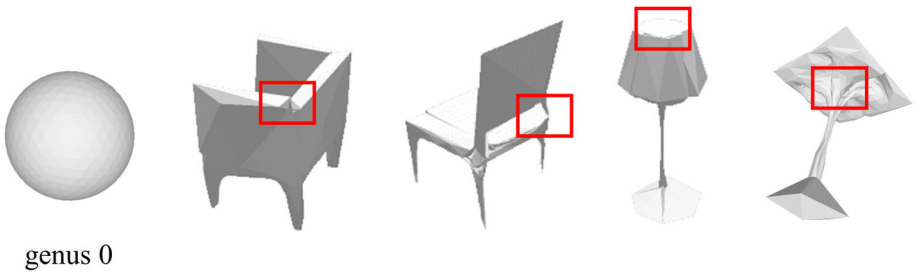


FIGURE 10 Compared to the state-of-art single shaded image-based shape reconstruction method NMR.[8] The results of NMR are lack of concave features, while ours show more accurate and detailed surface structures

FIGURE 11 The *genus* of predefined sphere is 0, therefore, the hollow structure cannot be fully recovered



4.3 | Limitation

Although our method has achieved promising results for the single sketch-based modeling, there are flaws in recovering various topologies, especially when the topological genus is differ from the predefined mesh. As shown in Figure 11, the edges of chair are not fully connected, even though the concave features are recovered, the hollow structure especially in Lamp is ignored. Theoretically, our approach is suitable for the shapes with different topologies, as long as the predefined mesh is changed to shape with the corresponding topological genus. In addition, incorrect normal image generation will cause uncertain failure in final shape prediction.

5 | CONCLUSION

Research in sketch-based modeling has never been stopped since it has significant practical usage. However, the ambiguous features of line drawing like a Grand Canyon between sketch and its 3D shape, which has attracted lots of researchers to adventure . In this paper, we has explored the possibility of deep learning-based method, a unified learning framework with a differentiable renderer incorporated was presented. The promising experimental results demonstrate that the proposed framework is able to cope with the challenging in single sketch-based 3D shape reconstruction. In future, improve

the efficiency of the differentiable rendering, automatic identify the *genus* of the objective shape would be deserved to try. The proposed approach also shows potential in sketch-based 3D retrieval, it might be another fun adventure in the Grand Canyon.

ACKNOWLEDGEMENTS

The authors would like to appreciate the open dataset *ShapeNet* and open deep learning framework *Pytorch*. Author Nan Xiang would like to acknowledge financial support from China Scholarship Council (CSC, No.201707050015). The research leading to these results has been supported by European Regional Development Funds - VISTA AR project (funded by the Interreg France [Channel] England).

ORCID

Nan Xiang https://orcid.org/0000-0003-4028-2287
Yanran Li https://orcid.org/0000-0003-1385-7604

REFERENCES

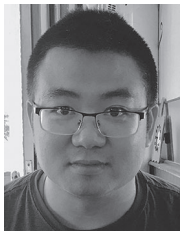
1. Bessmeltsev M. Recovering 3D shape from concept and pose drawings. Canada: University of British Columbia, 2016.
2. Malik J. Interpreting line drawings of curved objects. Int J Comput Vis. 1987;1(1):73–103.
3. Malik J, Maydan D. Recovering three-dimensional shape from a single image of curved objects. IEEE Trans Pattern Anal Mach Intell. 1989;11(6):555–566.
4. Shao C, Bousseau A, Sheffer A, Singh K. CrossShade: Shading concept sketches using cross-section curves. ACM Trans Graph. 2012;31(4):1–11.
5. Jung A, Hahmann S, Rohmer D, Begault A, Boissieux L, Cani MP. Sketching folds: Developable surfaces from non-planar silhouettes. ACM Trans Graph. 2015;34(5):155.
6. Iarussi E, Bommes D, Bousseau A. Bendfields: Regularized curvature fields from rough concept sketches. ACM Trans Graph. 2015;34(3):24.
7. Liu S, Li T, Chen W, Li H. Soft rasterizer: A differentiable renderer for image-based 3d reasoning. Proceedings of the IEEE International Conference on Computer Vision, Seoul, Korea; 2019. p. 7708–7717.
8. Kato H, Ushiku Y, Harada T. Neural 3d mesh renderer. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, Utah, USA; 2018. p. 3907–3916.
9. Chen W, Ling H, Gao J, et al. Learning to predict 3d objects with an interpolation-based differentiable renderer. Advances in Neural Information Processing Systems. 2019:9605–9616. https://github.com/nv-tlabs/DIB-R.
10. Yan X, Yang J, Yumer E, Guo Y, Lee H. Perspective transformer nets: Learning single-view 3d object reconstruction without 3d supervision. In: Lee DD, Sugiyama M, Luxburg UV, Guyon I, Garnett R. Advances in Neural Information Processing Systems 29. Curran Associates, Inc.; 2016:1696–1704. http://papers.nips.cc/paper/6206-perspective-transformer-nets-learning-single-view-3d-object-reconstruction-without-3d-supervision.pdf.
11. Pan J, Li J, Han X, Jia K. Residual MeshNet: Learning to deform meshes for single-view 3D reconstruction. Proceedings of the 2018 International Conference on 3D Vision (3DV). Verona, Italy: IEEE; 2018. p. 719–727.
12. Wang N, Zhang Y, Li Z, Fu Y, Liu W, Jiang YG. Pixel2mesh: Generating 3d mesh models from single RGB images. Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany; 2018. p. 52–67.
13. Tatarchenko M, Dosovitskiy A, Brox T. Octree generating networks: Efficient convolutional architectures for high-resolution 3d outputs. Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy; 2017. p. 2088–2096.
14. Groueix T, Fisher M, Kim VG, Russell BC, Aubry M. A papier-mâché approach to learning 3D surface generation. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, Utah, USA; 2018. p. 216–224.
15. Fan H, Su H, Guibas LJ. A point set generation network for 3d object reconstruction from a single image. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, Hawaii, USA; 2017. p. 605–613.
16. Smirnov D, Bessmeltsev M, Solomon J. Deep sketch-based modeling of man-made shapes; 2019. arXiv preprint arXiv:190612337;.
17. Delanoy J, Aubry M, Isola P, Efros AA, Bousseau A. 3D sketching using multi-view deep volumetric prediction. Proc ACM Comput Graph Interact Techn. 2018;1(1):21.
18. Moriya T. Pix2vox: Sketch-based 3D exploration with stacked generative adversarial networks. San Francisco, USA: Github; ; 2017. Retrieved from: https://github.com/maxorange/pix2vox/.
19. Lun Z, Gadelha M, Kalogerakis E, Maji S, Wang R. 3D shape reconstruction from sketches via multi-view convolutional networks. Proceedings of the 2017 International Conference on 3D Vision (3DV). Qingdao, China: IEEE; 2017. p. 67–77.
20. Marschner S, Shirley P. Fundamentals of computer graphics. Boca Raton, FL: CRC Press, 2015.
21. Xiang N, Wang L, Jiang T, Li Y, Yang X, Zhang J. Single-image mesh reconstruction and pose estimation via generative normal map. Proceedings of the 32nd International Conference on Computer Animation and Social Agents, Paris, France; 2019. p. 79–84.
22. Su W, Du D, Yang X, Zhou S, Fu H. Interactive sketch-based normal map generation with deep neural networks. Proc ACM Comput Graph Interact Techn. 2018;1(1):1–17.
23. Mirza M, Osindero S. Conditional generative adversarial nets; 2014. arXiv preprint arXiv:14111784.
24. Cordier F, Seo H, Melkemi M, Sapidis NS. Inferring mirror symmetric 3D shapes from sketches. Comput Aid Des. 2013;45(2):301–311.

25. Chen T, Zhu Z, Shamir A, Hu SM, Cohen-Or D. 3-sweep: Extracting editable objects from a single photo. ACM Trans Graph. 2013;32(6):1–10.
26. Shtof A, Agathos A, Gingold Y, Shamir A, Cohen-Or D. Geosemantic snapping for sketch-based modeling. Comput Graph Forum. 2013;32(2):245–253.
27. Nealen A, Igarashi T, Sorkine O, Alexa M. FiberMesh: Designing freeform surfaces with 3D curves. ACM SIGGRAPH 2007 papers. New York, NY: Association for Computing Machinery; 2007; p. 41–es.
28. Cherlin JJ, Samavati F, Sousa MC, Jorge JA. Sketch-based modeling with few strokes. Proceedings of the 21st Spring Conference on Computer Graphics, Budmerice, Slovakia; 2005. p. 137–145.
29. Jain AK, Zhong Y, Dubuisson-Jolly MP. Deformable template models: A review. Signal Process. 1998;71(2):109–129.
30. Loper MM, Black MJ. OpenDR: An approximate differentiable renderer. Proceedings of the European Conference on Computer Vision. New York, NY: Springer; 2014. p. 154–169.
31. Pathak D, Krahenbuhl P, Donahue J, Darrell T, Efros AA. Context encoders: Feature learning by inpainting. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, Nevada, USA; 2016. p. 2536–2544.
32. Rezatofighi H, Tsoi N, Gwak J, Sadeghian A, Reid I, Savarese S. Generalized intersection over union: A metric and a loss for bounding box regression. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, California, USA; 2019. p. 658–666.
33. Isola P, Zhu JY, Zhou T, Efros AA. Image-to-image translation with conditional adversarial networks. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, Hawaii, USA; 2017. p. 1125–1134.
34. Goodfellow I, Pouget-Abadie J, Mirza M, et al. Generative adversarial nets. Advances in Neural Information Processing Systems. Palais des Congrés de Montréal, Montréal Canada; 2014:2672–2680.
35. Chang AX, Funkhouser T, Guibas L, et al. Shapenet: An information-rich 3d model repository; 2015. arXiv preprint arXiv:151203012.
36. DeCarlo D, Finkelstein A, Rusinkiewicz S, Santella A. Suggestive contours for conveying shape. ACM SIGGRAPH 2003 Papers. New York, NY: Association for Computing Machinery; 2003; p. 848–855.

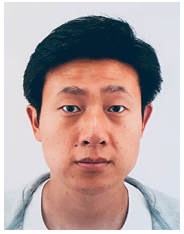
AUTHOR BIOGRAPHIES

Nan Xiang Nan Xiang received his BS degree from the School of Software, Nanchang University, China, in 2012, MA degree from School of Animation and Digital Arts, Communication University of China, in 2017. He is currently a PhD student at the National Centre for Computer Animation, Bournemouth University, UK. His research interests include virtual reality, 3D reconstruction and deep learning.

Ruibin Wang Ruibin Wang is currently a PhD candidate at National Centre for Computer Animation, Bournemouth University, UK. He received his BS and ME degree in Applied Mathematics and Vehicle Operation Engineering from Southwest Jiaotong University (China) in 2016 and 2019, respectively. His research interests include Natural Language Processing and 3D reconstruction.



Tao Jiang Tao Jiang received the BSc and MSc degrees from the University of Electronic Science and Technology of China (UESTC) and his PhD from Bournemouth University. Now, he is a PostDoc in CVSSP at University of Surrey. His research interests include deep learning, computer vision and computer graphics, specifically, human pose estimation, SLAM and shape registration.



Li Wang Li Wang is currently a PhD candidate at National Centre for Computer Animation, Bournemouth University, UK. He received his BS and ME degree in Computer Science from Jilin University (China) in 2013 and 2016, respectively. His research interests include deep learning, computer vision and computer graphics.

Yanran Li Yanran Li is currently a PhD Researcher in the National Centre for Computer Animation, Bournemouth University, UK. She received her MSc degree in Mathematics from the University of Science and Technology of China and Bachelor degree in Applied Mathematics from Xian Jiaotong University. Her research interests include computer graphics, computer vision, deep learning, motions and images.

Xiaosong Yang Xiaosong Yang is currently an Associate Professor in the National Centre for Computer Animation, Bournemouth University, UK. He received his bachelor (1993) and master degree (1996) in computer science from Zhejiang University (P. R. China) and PhD (2000) in computing mechanics from Dalian University of Technology (P. R. China). He worked as PostDoc (2000-2002) in the Department of Computer Science and Technology of Tsinghua University for two years and as Research Assistant (2001-2002) at Chinese University of Hong Kong. His research interests include deep learning, computer vision, computer animation, motion capture & synthesis, VR & AR, special effects & game development, digital health, data mining, medical visualization.

Jianjun Zhang Jianjun Zhang is currently a Professor of Computer Graphics at the National Centre for Computer Animation, Bournemouth University, and leads the Computer Animation Research Centre. His research focuses on a number of topics relating to 3D computer animation, including virtual human modeling and simulation, geometric modeling, motion synthesis, deformation and physics-based animation. He is also interested in virtual reality and medical visualization and simulation. Prof. Zhang has published over 200 peer-reviewed journal and conference publications. He has chaired over 30 international conferences and symposia and serves on a number of editorial boards.

How to cite this article: Xiang N, Wang R, Jiang T, et al. Sketch-based modeling with a differentiable renderer. *Comput Anim Virtual Worlds*. 2020;e1939. https://doi.org/10.1002/cav.1939