Hindawi
Wireless Communications and Mobile Computing
Volume 2021, Article ID 5577530, 9 pages
https://doi.org/10.1155/2021/5577530

Research Article

Reconstructing 3D Model from Single-View Sketch with Deep Neural Network

Fei Wang,[1] **Yu Yang**,[2] **Baoquan Zhao**,[3] **Dazhi Jiang**,[1] **Siwei Chen**,[1] **and Jianqiang Sheng**[4]

[1]*Shantou University, Shantou, China*
[2]*Shenzhen Securities Information Co., Ltd, Shenzhen, China*
[3]*Guilin University of Electronic Technology, Guilin, China*
[4]*Shenzhen Institute of Information Technology, Shenzhen, China*

Correspondence should be addressed to Baoquan Zhao; zbqsys@gmail.com

Received 8 January 2021; Revised 5 March 2021; Accepted 26 March 2021; Published 27 April 2021

Academic Editor: Amr Tolba

In this paper, we introduce a novel 3D shape reconstruction method from a single-view sketch image based on a deep neural network. The proposed pipeline is mainly composed of three modules. The first module is sketch component segmentation based on multimodal DNN fusion and is used to segment a given sketch into a series of basic units and build a transformation template by the knots between them. The second module is a nonlinear transformation network for multifarious sketch generation with the obtained transformation template. It creates the transformation representation of a sketch by extracting the shape features of an input sketch and transformation template samples. The third module is deep 3D shape reconstruction using multifarious sketches, which takes the obtained sketches as input to reconstruct 3D shapes with a generative model. It fuses and optimizes features of multiple views and thus is more likely to generate high-quality 3D shapes. To evaluate the effectiveness of the proposed method, we conduct extensive experiments on a public 3D reconstruction dataset. The results demonstrate that our model can achieve better reconstruction performance than peer methods. Specifically, compared to the state-of-the-art method, the proposed model achieves a performance gain in terms of the five evaluation metrics by an average of 25.5% on the man-made model dataset and 23.4% on the character object dataset using synthetic sketches and by an average of 31.8% and 29.5% on the two datasets, respectively, using human drawing sketches.

1. Introduction

3D shape reconstruction as one of the most fundamental problems in computer graphics is playing an increasingly important role in a wide variety of fields such as virtual/augmented reality, computer-aided geometric design, gaming, and medical imaging. However, manually reconstructing 3D models from scratch is a nontrivial task. This is because the procedure generally involves intensive interactions, which will take a designer a lot of time and effort to craft an exquisite 3D model. To alleviate this situation, a large body of approaches have been developed to facilitate the 3D shape reconstruction process. Among them, automatically reconstructing 3D models from sketch images is gaining more and more popularity due to its high efficacy and simplicity of interaction.

To harvest 3D models from free-hand sketches, a crucial factor is how to accurately understand their semantic meaning. Towards this end, conventional sketch-based 3D shape reconstruction methods like [1] rely on hand-crafted features. With the recent advance in deep learning technology, deep neural network-based 3D shape reconstruction has achieved remarkable progress. Despite these achievements, there are still many challenging issues in this area that have not been effectively addressed, which are seriously hindering the adoption of this technique in many domains. These issues are mainly featured in the following three aspects. Firstly, there is a great semantic gap between sketch images and 3D models. Compared with a nature image, a sketch is a visual representation form with a high level of abstraction and can easily cause ambiguity. This character could bring great challenges to the understanding of sketches and will affect the performance of 3D reconstruction. Besides, drawing skills and painting styles of different users vary greatly, which further exacerbate the difficulties of sketch semantic analysis. Secondly, deep learning-based 3D reconstruction models are generally highly dependent on sufficient training data. With the diversity of users' painting styles, it is very time-consuming and costly to collect tens of thousands of well-labelled sketch images that can be used to feed into deep neural networks for effective training. Last but not least, the dimensionality and features of sketches and 3D models are quite different. How to effectively exploit the very limited visual clues in sketch images to reconstruct 3D shape accurately is another challenging task that has not been adequately addressed by existing studies.

To tackle the aforementioned issues, in this paper, we introduce a novel 3D shape reconstruction framework from a sketch image using a deep neural network. Unlike conventional methods that need several sketch images from multiple views, the proposed approach only takes a single-view sketch image as input. This can significantly reduce the interaction during the reconstruction process, which is a very important factor that is highly concerned by practitioners in real-world applications. However, compared with multiview-based reconstruction methods, extracting meaningful features from a single-view sketch may be insufficient for accurate 3D shape reconstruction. This is because sketches from multiple views are more likely to convey more useful features to infer the underlying structure of a 3D shape. To gain the merit of multiview sketch-based reconstruction frameworks, we formulate the problem as a three-stage task, i.e., we first segment an input sketch into a series of basic units in the first stage and use the units to build a transformation template and create multifarious sketches in the second stage before reconstructing 3D shape in the third stage. Such a strategy paves the way to harvesting high-quality 3D models with less input and human-computer interaction. This work is an extension of our conference paper [2]. In this extended version, we introduce a new network optimization component and conduct more experiments to evaluate the effectiveness of our method.

The rest of this paper is organized as follows. In the next section, we briefly introduce existing researches on 3D shape reconstruction from sketch images. The details of the proposed framework are presented in Section 3. We conduct an extensive experiment to demonstrate the effectiveness of the proposed method in Section 4. Section 5 concludes the paper.

2. Related Work

In this section, we briefly review existing researches that are closely related to our work from two aspects: (1) sketch understanding and (2) sketch-based composition and reconstruction.

2.1. Sketch Understanding. Sketch understanding is an emerging branch in the field of artificial intelligence [3–5], aimed at recognizing and extracting the semantic knowledge from sketch images in a fully/semiautomatic fashion. It mainly encompasses two parts, i.e., sketch recognition and semantic understanding [6]. A pioneer work on large-scale sketch recognition is "How do users draw sketches?" [1], in which 20 k sketch images in total were collected. It adopts a Gaussian derivative to estimate the direction of lines and utilizes a bag-of-words model to encode a local curve direction as the feature vector of sketches. Then, the support vector machine (SVM) is employed to classify sketch images. With the great achievement of the deep neural network (DNN) on natural image recognition, the convolutional neural network (CNN) has also been applied to sketch recognition. However, taking sketch images as a 2D pixel array, CNN-based methods usually need to learn a huge amount of network parameters, which is very inefficient to train the network. Compared with natural images, the features of sketch images are sparse. Such spare data can significantly improve the compactness of networks.

2.2. Sketch-Based Composition and Reconstruction. A large body of studies have been carried out to extract features from sketch images and use them to perform a variety of tasks such as 3D model retrieval and shape reconstruction. For example, Chen and Fang [7] proposed to retrieve a 3D model using a sketch by constructing two individual deep CNN and metric networks. One network is for sketch images, and the other one is for 3D models. Interleaved active metric learning (IAML) is used to learn specific features from these two modalities, which is capable of mining important features from samples for training and learning discriminative feature representation effectively. Besides, to reduce the cross-modality difference between sketch features and 3D shapes, it also introduced a modality transformation network to convert sketch features into the feature space of 3D models, which achieves better retrieval performance.

This technique also paves the way for the development of sketch-based 3D shape reconstruction [8, 9]. Wang et al. [10] developed a label-free sketch neural network 3D-GAN. This model is integrated with an embedding latent vector space to harvest a similar feature vector distribution between sketch image and 2D rendered image. Then, it obtains final results by retrieving the k most similar 3D models with a sketch as the prior knowledge. Lun et al. [11] mapped a sketch to 3D shape by training a ConvNet to infer the structure and reconstructed a 3D model using a multiview framework. Unlike conventional methods that adopt voxels to represent 3D models, the shape of 3D objects can be represented with surface-based forms (for example, polygon mesh), which can achieve more accurate prediction by combining feedforward frameworks.

3. The Proposed Deep Neural Network for 3D Shape Reconstruction

3.1. Overview of the Proposed Method. The proposed pipeline is mainly composed of three components: (1) sketch component segmentation based on multimodal DNN fusion, (2) multifarious sketch generation based on nonlinear transformation network, and (3) deep 3D shape reconstruction using

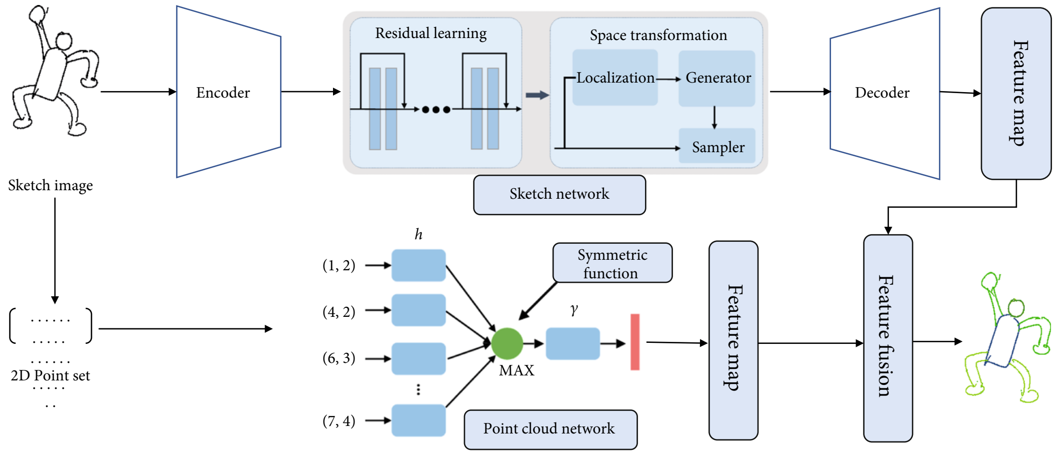

Figure 1: Sketch component segmentation based on multimodal DNN fusion.

multifarious sketches. The first component is a 2D point cloud-based sketch segmentation model, which is used to segment a given sketch into a series of basic units and build a transformation template by the knots between them. The advantage of this component is that it only relies on a small amount of sample data to achieve the learning task of the transformation network. The second component is developed for generating multifarious sketches with the aforementioned transformation template. It creates the transformation representation of a sketch by extracting the shape features of an input sketch and transformation template samples, which can avoid the problems existing in conventional models such as unitary transformation structure and distortion. The third component takes the obtained sketches as input to reconstruct 3D shapes with a generative model. It fuses and optimizes features of multiple views and thus is more likely to generate high-quality 3D shapes. Details of each component will be introduced in the following three subsections.

3.2. Sketch Component Segmentation Based on Multimodal DNN Fusion. Sketch component segmentation net is inspired by work [6]. The sketch network mainly consists of two parts: encoder and decoder. On the one hand, the network obtains the global feature of a sketch through the encoder. As shown in Figure 1, feature representation is learned and extracted using spatial invariance enhanced residual (SIER), which is composed of two modules: residual learning module and spatial transformation module. These features will be combined together in the decoding phrase to generate a pixel-level feature segmentation image. On the other hand, the coordinate information of sketch contour is an important geometry structure. Therefore, a 2D point cloud network can obtain the feature of a sketch by representing each point with 2D coordinates (x, y). Let $P = \{p_i \mid i = 1, \cdots, n\}$ be the coordinate set of sketch contour, where p_i represents the coordinate of each sample point; the 2D point cloud network takes P as input and gets the global features by gathering point features with maximum function MAX. Then, the probability of each point in P associated with all semantic units can be obtained by connecting local and global features through a segmentation network. More specifically, let $f : X \rightarrow R$ be a continuous set function regarding Hausdorff distance $d_H(\cdot, \cdot)$. For $\forall \varepsilon > 0$, there exists a continuous function $g(x_1, \cdots, x_n) = \gamma \circ$ MAX such that for arbitrary $x_i \in X$,

$$|f(\{x_i, \cdots, x_n\}) - \gamma\left(\text{MAX}\left(h\left(x_q\right), h(x_2), \cdots, h(x_n)\right)\right)| < \varepsilon, \quad (1)$$

where γ and h are continuous functions and MAX is a vector maximum description operator. Therefore, the general function of the arbitrary 2D point set can be represented as

$$f(\{x_1, \cdots, x_n\}) \approx g(h(x_1), \cdots, h(x_n)). \quad (2)$$

As shown in Figure 1, h can be learned with multilayer perception (MLP), and $g = \gamma \circ \text{MAX}$ can be obtained with single variable function and max pooling.

After performing sketch segmentation, we use the transformation templates of the original sketch to train the network. As illustrated in Figure 2, a sketch is segmented into a series of semantic units. The knot between units is represented with $\{(p_i, q_i)\}$, which is highlighted with a red circle in the figure.

3.3. Multifarious Sketch Generation Based on Nonlinear Transformation Network. As shown in Figure 3, the proposed pipeline is mainly composed of two modules, i.e., multiview sketch generation module and 3D model generation module. The aim of segmentation is to obtain small samples of sketches for the use of training the network in Section 3.2. Multiview sketches are generated by the Sketch-VAE network with the input sketch image. The obtained multiview sketches are then fed into the encoder and decoder to harvest a depth image and a normal image, with which we can generate a 3D point cloud. And finally, a 3D polygon model can be obtained by performing remeshing on the point cloud.

In this step, we first carry out transformation feature extraction and representation. For $m (m \geq 2)$ sketch images, each of which has n vertexes, let p_i be the v_ith vertex of an input sketch and p'_i be the one in the transformation

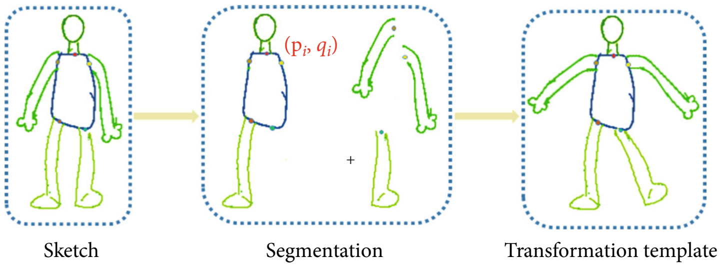

Figure 2: Sketch segmentation, knots, and transformation template.

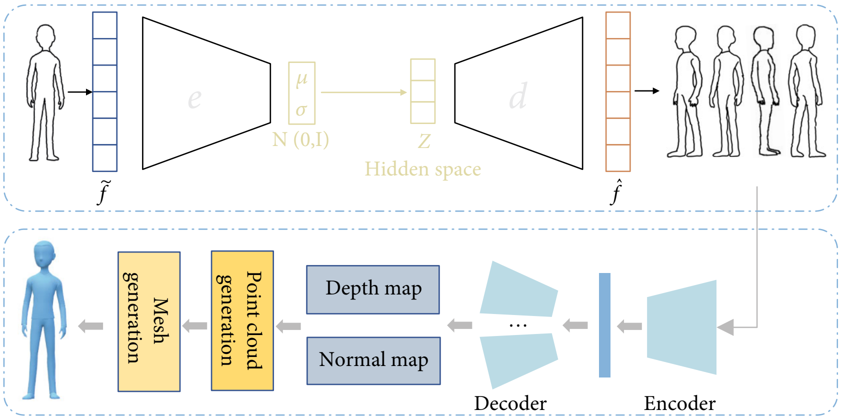

Figure 3: Deep 3D shape reconstruction using multifarious sketches.

template; the transformation gradient T_i can be obtained by minimizing the energy function below:

$$E(T_i) = \sum_{j\in N_i} c_{ij} \|e'_{ij} - T_i e_{ij}\|^2, \tag{3}$$

where N_i is the neighborhood set of v_i, $e'_{ij} = p'_i - p'_j$, $e_{ij} = p_i - p_j$, and $c_{ij} = \cot \alpha_{ij} + \cot \beta_{ij}$ is cotangent weight. The affine transformation matrix can be decomposed into rotation part and scaling part $T_i = R_i S_i$. The rotation difference from v_i to v_j is given by

$$dR_{ij} = R_i^T R_j. \tag{4}$$

Thus, the energy function is redefined as

$$E(T_i) = \sum_{j\in N_i} c_{ij} \sum_{t\in N_i} c'_i \|e'_{ij} - R_i dR_{ti} S_i e_{ij}\|^2, \tag{5}$$

where $c'_i = 1/|N_i|$ and $|N_i|$ is the number of neighborhood of v_i. Finally, the feature of the transformation template on v_i can be represented as

$$f_i^j = \left\{\log dR_{ij}\, ; S_i\right\} (\forall i, j \in N_i). \tag{6}$$

The Sketch-VAE network is aimed at finding an encoder and a decoder, where the goal of the encoder is to map the posterior distribution of x to hidden vector z, while that of the decoder is to generate a credible x. The loss function of the Sketch-VAE model is defined as

$$L_{\mathrm{VAE}} = \sum_{j=1}^{M} \sum_{i=1}^{K} \left(f\wedge_i^j - \tilde{f}_i^j\right)^2 + D_{KL}\left(q\left(z \mid \tilde{f}\right) \| p(z)\right), \tag{7}$$

where $\tilde{f}_i^j$ is the transformed feature after preprocessing and $\widehat{f}_i^j$ is the output of the Sketch-VAE model. z is a hidden vector; $p(z)$ and $q(z \mid \tilde{f})$ are prior and posterior probability, respectively; and D_{KL} is KL divergence.

To avoid incorrect output caused by the separation between sketch components, we add a constrain condition and define the loss function of knots as

$$L_{\mathrm{joints}} = \sum_{i=1}^{n} \|p_i - q_i\|^2. \tag{8}$$

Besides, we also add a regularization constrain to the network optimization network to avoid distortion and define the loss function as

$$L_{\mathrm{reg}} = \sum_{i=1}^{|V|} \sum_{j\in N_i} w_{ij} \left\| \left(\widehat{v}_i - \widehat{v}_j\right) - R_i\left(v_i - v_j\right)\right\|^2, \tag{9}$$

where v_i and $\widehat{v}_i$ are the vertexes in the original and transformed sketches, respectively.

Table 1: Segmentation accuracy (%) in terms of P-metric under different parameter settings on the SketchSeg dataset.

Loss weight λ_1	Airplane	Bicycle	Candelabra	Chair	Four legs	Human	Lamp	Rifle	Table	Vase	Average
$\lambda_1 = 0.1$	89.3	89.2	91.5	90.4	87.3	84.2	89.4	89.2	**93.4**	**95.8**	89.8
$\lambda_1 = 0.3$	**92.1**	**91.6**	**93.1**	**91.8**	**87.6**	**84.3**	**89.6**	**90.4**	88.7	92.1	90.1
$\lambda_1 = 0.5$	90.3	91.0	91.8	91.7	85.9	82.9	90.2	88.8	87.9	91.3	89.2
$\lambda_1 = 0.7$	88.7	89.6	89.4	85.3	83.6	80.6	86.9	83.5	81.6	88.5	85.8
$\lambda_1 = 0.9$	84.0	84.5	85.7	81.6	82.0	74.9	83.0	84.0	76.6	80.7	81.7

3.4. Deep 3D Shape Reconstruction Using Multifarious Sketches. The proposed deep 3D shape reconstruction framework is illustrated in Figure 3. For the sketches obtained in the last step, the encoder first extracts the feature of each sketch. It consists of a series of convolution, and all layers use ReLU as an active function. Then, the decoder transforms these features into depth and normal images, which will be fused into a 3D point cloud subsequently. The first step of fusion is to map all foreground pixels to 3D points. If the prediction probability of a pixel is larger than 50%, we consider it as a foreground pixel. Let the depth of a foreground pixel p be $p_{p.v}$, the coordinate set of graph space in the ith sketch image be $\{p_x, p_y\}$, and the position of a 3D point $q_{p,i}$ can be calculated as

$$q_{p,i} = R_v \left[\kappa p_x \kappa p_y d_{p,i}\right]^T + e_i, \tag{10}$$

where κ is a scaling coefficient, representing the distance between adjacent pixels and the center. Finally, the 3D model can be obtained by transforming it into polygon mesh with the Poisson surface reconstruction algorithm [12].

3.5. Network Optimization. To constrain the two feature vectors, we use the relative cross-entropy to evaluate their similarity and select sigmoid normalization-based cross-entropy as the loss function of the proposed sketch pixel network:

$$L_{\text{sketch pixel}} = -\frac{1}{K}\sum_{k}^{K}\left[y_k \log\left(\frac{1}{1+e^{-c^t}}\right) + (1-y_k)\log\left(\frac{e^{-c^t}}{1+e^{-c^t}}\right)\right], \tag{11}$$

where y_k indicates whether the label k exists in the obtained segmentation result. If so, we set $y_k^{cls} = 1$; otherwise, $y_k = 0$. c_k is the prediction probability that the label k appears in the results. Likewise, $L_{\text{point cloud}}$ is the feature constrain of the 2D point cloud. We can perform the feature fusion via multiple ways such as cascade and weighted sum. Meanwhile, the weights of different networks have a direct impact on the results. We will study the impact of the normalization of different network components on the results in Table 1. The objective function is formulated as

$$L = \lambda_1 L_{\text{sketch pixel}} + \lambda_2 L_{\text{point cloud}}. \tag{12}$$

4. Experiment and Discussion

4.1. Experiment Setup. To evaluate the effectiveness of the proposed method for 3D shape reconstruction, we conduct a comparative experiment on a public dataset. To train our neural network, we use the dataset presented in [11], which mainly consists of three different types of 3D models, i.e., human/humanoid, airplanes, and chairs. Among them, human/humanoid involves human models, aliens, and virtual cartoon characters, which come from *The Models Resource* dataset [13], while airplane and chair models are mainly from *3D ShapeNet* [14], which has a large variety in shape geometry. There are 120 sketch images in total in the test dataset. Among them, 90 are synthetic sketches, which are generated from test images with line painting techniques, while the rest 30 sketch images are drawn by two professional artists. They were asked to draw 10 sketch images for each category. The sketch images were scaled to a size of 800 × 800. We train our model for 50 epochs with an SGD optimizer. The size of minibatch, initial learning rate, and momentum was set to 5, 0.01, and 0.9, respectively. The weight parameters of cross-entropy loss are 0 for background and 0.1 for others, which can avoid the interference of the background. The experiment was conducted on a PC equipped with an Intel i7 CPU, 32 GB RAM, and GTX 2080ti GPU. For the weight parameters λ_1 and λ_2 in Equation (12), we formulate $\lambda_1 + \lambda_2 = 1$ and set λ_1 to 0.3 according to the parameter analysis described in Table 1.

Peer sketch-based 3D shape reconstruction methods selected for comparison include ShapeMVD [11], nearest retrieval, Tatarchenko et al. [15], U-net [16], volumetric decoder, and R2N2 [17], which are state-of-the-art models and widely used by existing studies for performance evaluation. For the nearest-neighbor baseline, we extract the representation of the input test sketches based on our encoder. This is used as a query representation to retrieve the training shape whose sketches have the nearest encoder representation based on the Euclidean distance. We additionally implemented a variant of Tatarchenko et al.'s decoder by adding U-net connections between the encoder and their decoder. The volumetric decoder consisted of five transpose 3D convolutions of stride 2 and kernel size 4 × 4 × 4. The number of filters starts with 512 and is divided by 2 at each layer. Leaky ReLU functions and batch normalization were used

Table 2: Man-made objects (synthetic).

	Ours	ShapeMVD	Nearest retrieval	Tatarchenko et al. [15]	[15]+U-net	Volumetric decoder	R2N2
Hausdorff distance	**0.076**	0.092	0.165	0.142	0.121	0.113	0.144
Chamfer distance	**0.011**	0.015	0.025	0.022	0.017	0.021	0.026
Normal distance	**26.45**	30.66	42.57	35.58	32.32	49.40	48.78
Depth map error	**0.013**	0.026	0.049	0.039	0.030	0.038	0.045
Volumetric distance	**0.276**	0.344	0.501	0.442	0.374	0.432	0.512

Table 3: Character models (synthetic).

	Ours	ShapeMVD	Nearest retrieval	Tatarchenko et al. [15]	[15]+U-net	Volumetric decoder	R2N2
Hausdorff distance	**0.065**	0.089	0.200	0.119	0.092	0.152	0.148
Chamfer distance	**0.010**	0.015	0.036	0.025	0.016	0.026	0.032
Normal distance	**26.47**	30.61	44.93	34.98	31.00	53.84	53.13
Depth map error	**0.014**	0.018	0.040	0.030	0.019	0.031	0.036
Volumetric distance	**0.248**	0.313	0.541	0.428	0.329	0.437	0.493

Table 4: Man-made objects (human drawing).

	Ours	ShapeMVD	Nearest retrieval	Tatarchenko et al. [15]	[15]+U-net	Volumetric decoder	R2N2
Hausdorff distance	**0.094**	0.116	0.176	0.153	0.153	0.130	0.149
Chamfer distance	**0.011**	0.017	0.031	0.024	0.025	0.022	0.028
Normal distance	**21.058**	27.04	40.96	32.40	30.45	48.32	48.12
Depth map error	**0.011**	0.021	0.042	0.033	0.032	0.032	0.042
Volumetric distance	**0.202**	0.311	0.544	0.405	0.403	0.405	0.500

Table 5: Character models (human drawing).

	Ours	ShapeMVD	Nearest retrieval	Tatarchenko et al. [15]	[15]+U-net	Volumetric decoder	R2N2
Hausdorff distance	**0.102**	0.117	0.188	0.139	0.136	0.178	0.168
Chamfer distance	**0.013**	0.021	0.036	0.025	0.024	0.032	0.036
Normal distance	**28.22**	33.44	43.81	36.11	34.74	54.91	54.29
Depth map error	**0.012**	0.026	0.040	0.031	0.027	0.037	0.040
Volumetric distance	**0.217**	0.298	0.458	0.342	0.307	0.420	0.436

after each layer. We note that we did not use skip connections (U-net architecture) in the volumetric decoder because the size of the feature representations produced in the sketch image-based encoder is incompatible with the ones produced in the decoder.

4.2. Overall Reconstruction Performance Comparison against Peer Methods. Following the common practice, we also compare the similarity between the reconstructed 3D models and input sketches using the following five distance metrics: Chamfer distance, Hausdorff distance, surface normal distance, depth map error, and volumetric Jaccard distance. The comparison results are illustrated in Tables 2–5, where Tables 2 and 3 are the results of synthetic sketches while Tables 4 and 5 are those of human drawing sketches. Bold values in the tables indicate the best results among all methods. From Tables 2 and 3, we can see that the proposed method achieves smaller distances than peer ones in terms of the five evaluation metrics on both man-made objects and character models. Specifically, compared to the state-of-the-art model ShapeMVD, our model achieves a performance gain in terms of Hausdorff distance, Chamfer distance, normal distance, depth map error, and volumetric distance by 17.4%, 26.7%, 13.7%, 50.0%, and 19.8% on the man-made objects dataset and by 27.0%, 33.3%, 13.5%, 22.2%, and 20.8% on the character model dataset, respectively. Overall, the proposed method improves the performance in terms of the five distance metrics by an average of 25.5% and 23.4% on the man-made objects and character models, respectively, which demonstrates that our method can generate more accurate 3D models. We can find that ShapeMVD performs better than conventional methods like U-net. However, its performance gain is just marginally higher. Our model outperforms all of these methods, and performance gain is significant. The superiority of the proposed method can also be reflected on the human drawing sketch

Table 6: Segmentation performance of MIFNet against state-of-the-art methods with the P-metric.

Method	U-net	LinkNet	FCN	PointNet	MIFNet
Airplane	68.9	78.0	78.2	81.0	**92.9**
Bicycle	68.1	65.3	71.4	78.0	**93.5**
Candelabra	89.3	88.3	**90.8**	81.1	93.0
Chair	84.0	89.1	86.9	81.0	**88.1**
Four legs	74.1	76.7	80.3	75.5	**89.3**
Human	71.9	74.5	75.6	69.2	**85.5**
Lamp	92.2	91.2	92.8	86.2	**87.8**
Rifle	54.8	59.9	65.2	83.2	**89.7**
Table	79.6	82.5	81.4	82.0	**88.6**
Vase	89.9	93.8	**94.4**	84.8	92.8
Average	77.3	79.9	81.7	80.2	**90.1**

Table 7: Segmentation performance of MIFNet against state-of-the-art methods with the C-metric.

Method	U-net	LinkNet	FCN	PointNet	MIFNet
Airplane	52.6	67.7	66.5	67.3	**86.5**
Bicycle	49.7	55.7	**59.2**	50.9	85.5
Candelabra	90.3	89.0	**94.5**	67.9	94.8
Chair	81.9	89.2	84.8	77.6	**85.1**
Four legs	54.5	67.2	73.5	60.9	**84.2**
Human	62.6	67.9	**72.1**	56.6	81.2
Lamp	92.4	92.4	92.5	86.1	**87.0**
Rifle	38.9	44.5	54.7	59.7	**82.1**
Table	70.1	80.3	75.3	67.5	**82.6**
Vase	90.7	96.6	**98.1**	78.9	93.1
Average	68.4	75.0	77.1	67.3	**86.2**

dataset, as shown in Tables 4 and 5. Likewise, the distance reduction compared with other methods is also prominent. For example, compared to ShapeMVD, the proposed method reduces the Chamfer distance by 33.3% and 38.1% on the man-made object objects and character models, respectively. Overall, our method achieves a performance gain on these two human drawing sketch datasets by an average of 31.8% and 29.5%, respectively. These experimental results demonstrate the effectiveness of the proposed method in reconstructing 3D shape from single-view sketches.

4.3. Evaluation of the Effectiveness of the Proposed Sketch Segmentation Method. To further verify the effectiveness of the proposed sketch segmentation method, we conducted an experiment on a public sketch segmentation database, i.e., SketchSeg. The two evaluation metrics used in this paper are pixel-based accuracy (P-metric) and component-based accuracy (C-metric), which is first proposed by Huang et al. [18] and widely used by peer work. Among them, P-metric is targeted for the evaluation of a whole sketch image, i.e., the percentage of the pixels of components that are correctly segmented to the pixels of the entire sketch, while C-metric is defined as the percentage of the number of components that are correctly segmented to the total number of components of the sketch. We treat a component as a correctly segmented one if more than 75% of its pixels are correctly predicted. The comparison between multisource information fusion (MIFNet) and peer methods is shown in Table 6. We can see from the table that the performance of MIFNet is superior to other methods. More specifically, it achieves an accuracy of 90.1% on average, while those of U-Net, LinkNet, FCN, and PointNet are 77.3%, 79.9%, 81.7%, and 80.2%, respectively. In other words, the proposed method improves the performance by 12.8%, 10.2%, 8.4%, and 9.9%, respectively. Besides, the component-based accuracy of peer methods is 68.4%, 75.0%, 77.1%, and 67.3%, respectively. MIFNet outperforms the FCN by 9.1% in terms of segmentation accuracy. Particularly, we can observe that the performance improvement on an *airplane*, *chair*, *human*, *gun*, and *desk* is more significant than others. This is because the average number of pixels of these five categories is higher than other ones. As for C-metric, the proposed method also shows an advantage over peer ones. As can be seen from Table 7, MIFNet improves the accuracy by 9.1%, compared to FCN, which demonstrates that MIFNet is more effective in capturing the structural information of sketch components.

4.4. The Selection of Loss Function Weight. Table 1 illustrates the comparison of different loss function weight λ_1 where λ_1 is the weight of pixel-based network while 1-λ_1 represents that of the point cloud-based network. We can see from the table that the segmentation results are less satisfactory when λ_1 is large, and the performance gets better gradually with the decrease of λ_1, which demonstrates that the point cloud-based network plays a more important role than the pixel-based one. Particularly, when $\lambda_1 = 0.3$, the performance of the pixel-based network reaches its peak and achieves an accuracy of 90.1%. In light of this, we thus train the model with this setting in our experiment.

5. Conclusion

In this paper, we have introduced a novel 3D shape reconstruction method from a single-view sketch image using a deep neural network. Our model is general and can be easily extended to other applications, such as biomedical and intelligent computing [19, 20]. The proposed method first generates a series of sketch images from different viewpoints by analysing the semantic information of the input sketch image. Then, the obtained sketch images are fed into a deep neural network to reconstruct the 3D shapes. Compared with multiview-based approaches, the proposed method only takes a single sketch image as input, which can significantly reduce time used for drawing sketches and remarkably improve the reconstruction efficiency. Besides, using the input sketch image as visual clues to generate multiview sketch images is helpful to reconstruct 3D shapes more accurately, which is superior to conventional single sketch image-based 3D shape reconstruction methods. Extensive experiments on a public 3D shape reconstruction dataset have demonstrated the efficacy of the proposed model.

One of the most challenging issues related to sketch-based 3D shape reconstruction is that the painting skills and styles of different users vary greatly, which makes it difficult to develop a versatile model to successfully extract meaningful features and infer semantic information from all kinds of sketches. A limitation of the proposed method is that it may fail to accurately reconstruct a high-quality 3D model if the input sketch is painted at a very abstract level or can easily cause ambiguity. Therefore, more powerful deep neural networks and machine learning techniques such as [21, 22] will be a promising way to address the challenges and further improve the reconstruction performance. Besides, the proposed method can be extended to sketch-based dynamic 3D model creation, which is used to be a time-consuming and labour-intensive task in the field of cartoon animation.

Data Availability

The source codes used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research is supported by the National Natural Science Foundation of China (61902087 and 61902232), Natural Science Foundation of Guangxi (2018GXNSFAA294127), General Universities and Colleges Young Innovative Talents Project of Guangdong Province (2019GKQNCX120 and 2019GKQNCX121), Scientific Research Start-up Fund of Shantou University (09420021), Natural Science Foundation of Guangdong Province (2021A1515012302, S2018A030313420, and 2019A1515010943), Key Project of Basic and Applied Basic Research of Colleges and Universities in Guangdong Province (Natural Science) (2018KZDXM035), Basic and Applied Basic Research of Colleges and Universities in Guangdong Province (Special Projects in Artificial Intelligence) (2019KZDZX1030), and 2020 Li Ka Shing Foundation Cross-Disciplinary Research Grant (2020LKSFG05D and 2020LKSFG04D).

References

[1] M. Eitz, J. Haysy, and M. Alexa, "How do humans sketch objects?," *Acm Transactions on Graphics*, vol. 31, no. 4CD, pp. 1–10, 2012.

[2] F. Wang, Y. Yu, B. Zhao et al., "Deep 3D shape reconstruction from single-view sketch image," in *The 8th International Conference on Digital Home*, Dalian, China, 2020.

[3] D. Jiang, G. Tu, D. Jin et al., "A hybrid intelligent model for acute hypotensive episode prediction with large-scale data," *Information Sciences*, vol. 546, pp. 787–802, 2021.

[4] D. Jiang, K. Wu, D. Chen et al., "A probability and integrated learning based classification algorithm for high-level human emotion recognition problems," *Measurement*, vol. 150, article 107049, 2020.

[5] D. Jiang, Z. Tian, Z. He, G. Tu, and R. Huang, "A framework for designing of genetic operators automatically based on gene expression programming and differential evolution," *Natural Computing*, vol. 6, 2021.

[6] F. Wang, S. Lin, H. Wu et al., "SPFusionNet: sketch segmentation using multi-modal data fusion," in *2019 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1654–1659, Shanghai, China, 2019.

[7] J. Chen and Y. Fang, "Deep cross-modality adaptation via semantics preserving adversarial learning for sketch-based 3d shape retrieval," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 605–620, Munich, Germany, 2018.

[8] F. Wang, S. Lin, H. Li et al., "Multi-column point-CNN for sketch segmentation," *Neurocomputing*, vol. 392, pp. 50–59, 2020.

[9] F. Wang, S. Lin, X. Luo, B. Zhao, and R. Wang, "Query-by-sketch image retrieval using homogeneous painting style characterization," *Journal of Electronic Imaging*, vol. 28, no. 2, article 023037, 2019.

[10] L. Wang, C. Qian, J. Wang, and Y. Fang, "Unsupervised learning of 3D model reconstruction from handdrawn sketches," in *Proceedings of the 26th ACM international conference on Multimedia*, pp. 1820–1828, Seoul, Republic of Korea, 2018.

[11] Z. Lun, M. Gadelha, E. Kalogerakis, S. Maji, and R. Wang, "3D shape reconstruction from sketches via multi-view convolutional networks," in *2017 International Conference on 3D Vision (3DV)*, pp. 67–77, Qingdao, China, 2017.

[12] M. Kazhdan and H. Hoppe, "Screened Poisson surface reconstruction," *ACM Transactions on Graphics (ToG)*, vol. 32, no. 3, pp. 1–13, 2013.

[13] T. M. Resource, 2017, https://www.models-resource.com/.

[14] A. X. Chang, T. Funkhouser, L. Guibas et al., "ShapeNet: an information-rich 3D model repository," 2015, https://arxiv.org/abs/1512.03012.

[15] M. Tatarchenko, A. Dosovitskiy, and T. Brox, "Multi-view 3D models from single images with a convolutional network," in *Computer Vision – ECCV 2016: 14th European Conference*, vol. 9911, pp. 322–337, Amsterdam, The Netherlands, 2016.

[16] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention*, vol. 9351, pp. 234–241, Springer International Publishing, 2015.

[17] C. B. Choy, D. Xu, J. Gwak, K. Chen, and S. Savarese, "3d-r2n2: a unified approach for single and multi-view 3d object reconstruction," in *Computer Vision – ECCV 2016: 14th European Conference,*, vol. 9912, pp. 628–644, Amsterdam, The Netherlands, 2016.

[18] Z. Huang, H. Fu, and R. W. H. Lau, "Data-driven segmentation and labeling of freehand sketches," *ACM Transactions on Graphics (TOG)*, vol. 33, no. 6, pp. 1–10, 2014.

[19] D. Jiang, Z. He, Y. Lin, Y. Chen, and L. Xu, "An improved unsupervised single channel speech separation algorithm for processing speech sensor signals," *Wireless Communications and Mobile Computing*, vol. 2021, Article ID 6655125, 13 pages, 2021.

[20] D. Jiang, D. Jin, J. Zhuang, D. Tan, D. Chen, and Y. Liang, "A computational model of emotion based on audio-visual stimuli understanding and personalized regulation with concurrency," *Concurrency and Computation: Practice and Experience*, vol. 17, 2021.

[21] C. Ieracitano, A. Paviglianiti, M. Campolo, A. Hussain, E. Pasero, and F. C. Morabito, "A novel automatic classification system based on hybrid unsupervised and supervised machine learning for electrospun nanofibers," *IEEE/CAA Journal of Automatica Sinica*, vol. 8, no. 1, pp. 64–76, 2021.

[22] Q. Deng, L. Ma, A. Jin, H. Bi, B. H. Le, and Z. Deng, "Plausible 3D face wrinkle generation using variational autoencoders," *IEEE Transactions on Visualization & Computer Graphics*, vol. 1, 2021.

[23] F. Wang, S. Lin, X. Luo, and R. Wang, "Coupling computation of density-invariant and divergence-free for improving incompressible SPH efficiency," *IEEE Access*, vol. 8, pp. 135912–135919, 2020.

[24] L. Cai, Y. Yu, S. Zhang, Y. Song, Z. Xiong, and T. Zhou, "A sample-rebalanced outlier-rejected k -nearest neighbor regression model for short-term traffic flow forecasting," *IEEE Access*, vol. 8, pp. 22686–22696, 2020.

[25] H. Lu, D. Huang, Y. Song, D. Jiang, T. Zhou, and J. Qin, "St-trafficnet: a spatial-temporal deep learning network for traffic forecasting," *Electronics*, vol. 9, no. 9, pp. 1474–1517, 2020.

[26] H. Lu, Z. Ge, Y. Song, D. Jiang, T. Zhou, and J. Qin, "A temporal-aware lstm enhanced by loss-switch mechanism for traffic flow forecasting," *Neurocomputing*, vol. 427, pp. 169–178, 2021.

[27] C. Li, S. Tang, H. K. Kwan, J. Yan, and T. Zhou, "Color correction based on cfa and enhancement based on retinex with dense pixels for underwater images," *IEEE Access*, vol. 8, pp. 155732–155741, 2020.