Research on 3D Reconstruction of Furniture Based on Differentiable Renderer

YALIN MIAO, HUI JIANG, LIN JIANG, AND MENG TONG
Department of Information Science, Xi’an University of Technology, Xi’an 710048, China
Corresponding author: Hui Jiang (1320764607@qq.com)

ABSTRACT Due to the self-obscuration, traditional 3D reconstruction algorithms have difficulty in recovering the 3D structure of an object from a single image. With the rapid development of convolutional neural networks, 3D reconstruction based on deep learning has attracted a wide range of attention from researchers. However, it is expensive to obtain the 3D supervised data corresponding to the objects. To solve the above problems, we combine convolutional neural networks with differentiable renderer and propose the Mesh_CA in this paper, which enables reconstruction of a single image without 3D supervision data. Specifically, an ellipsoid is first initialized for each input single view, and then the features extracted by the convolutional neural network are used to guide the deformation of the ellipsoid to obtain the generated 3D object; After that, the generated object is passed into the differentiable renderer and its corresponding contour information is output; finally, calculating the error between the predicted contour and the real one, and the final 3D object is obtained after training and testing. By training and testing on five types of furniture objects on a large-scale public dataset ShapeNet, the performance of the proposed Mesh_CA surpasses current classical methods.

INDEX TERMS Deep learning, 3D reconstruction, differentiable renderer, attention mechanism.

I. INTRODUCTION
With the development of Virtual Reality(VR), Augmented Reality(AR), Mixed Reality(MR) and computer vision technology, 3D reconstruction of furniture is becoming more and more important in e-commerce product display and promotion, game design, virtual home and so on. It is important to generate furniture objects from single or multiple furniture images.

Restoring 3D object structure from 2D vision has been one of the main goals in the field of computer vision. Image-based 3D reconstruction usually refers to inferring the 3D geometry and structure of objects from one or more images, which is more widely used in many domains[1], [2], [3].

Traditional 3D reconstruction methods address the problem of dimensional loss in 2D images from a geometric perspective, focusing on mathematically understanding and formalizing the projection process from 3D to 2D, with the aim of proposing algorithmic solutions to this ill-posed problems. They usually require calibration of the camera’s internal and external parameters, which is a tedious process and the reconstruction results are affected by many factors (e.g., light).

The remarkable performance of Convolutional Neural Network (CNN) in the fields of target detection [4], natural language processing [5], [6] and image recognition [7], [8]. Besides, the release of a large number of 3D databases[9], [10], [11], more and more researchers are focusing on data-driven object 3D reconstruction field. Data-driven 3D shape synthesis is one of the challenging problems in computer vision and computer graphics, where the goal is to learn a model and subsequently generate 3D shape templates suitable for new shape synthesis, interpolation and editing.

Although reconstruction based on multiple views can achieve better reconstruction results than single view (e.g., 3D-R2N2), however, multiple images corresponding to...
objects are not easily available sometimes. In addition, the 3D structure corresponding to the images are not easily available due to the high cost of modeling. In this paper, inspired by the work of Kato et al. [12] and Liu et al. [13], we combine a differentiable renderer with convolutional neural network and realize 3D reconstruction of the input single furniture image in a mesh fashion with less information.

In the 3D reconstruction of furniture objects, since the input images are rendered images of furniture objects in different views, the same structure will have different positions and sizes in different views (e.g., chair legs have different sizes and positions in different views), and it is important to understand the positions and sizes of the same structure of objects in different views. Therefore, two modules are introduced in this paper, namely select kernel (SK) [14] and coordinate attention (CA) [15].

SK module consists of three components: Split, Fuse and Select. Split uses different kernel sizes ($3 \times 3$, $5 \times 5$) to process the same feature map respectively, corresponding to different receptive field sizes of neurons. Fuse aggregates the information from different kernel processing, aiming to obtain a global and integrated representation of the selection weights. Select aggregates the feature maps of different kernel branches based on the selection weight information. It was found that most neurons collect more information from the larger kernel branches when the image size remains constant and the target object becomes larger.

CA module aggregates the input feature maps along the vertical and horizontal directions respectively, generating a pair of direction-aware feature maps. In this way, long-range dependencies can be captured along one spatial direction while the other spatial direction preserves precise location information, thus allowing the network to locate objects of interest more accurately.

The main contributions can be summarized as follows:

1) We combine convolutional neural networks with a differentiable renderer and propose Mesh_CA. We equip Mesh_CA with well-designed encoder and loss functions, which show powerful reconstruction ability to the input single furniture image without 3D supervision.

2) For the problem that the same structure (e.g., chair legs) in different views has different sizes and positions leading to insufficient utilization of the input image features, we introduce SK and CA respectively to improve the reconstruction performance of the model. Further, smoothing loss and Laplace loss are also introduced to improve the visual effect of the generated objects.

3) Experimental results on the ShapeNet datasets demonstrate that the proposed approaches outperform the classical methods in terms of both accuracy and visual effect. Additional experiments also show its strong generalization abilities in reconstructing unseen 3D furniture objects.

II. RELATED WORKS

There are many representations of 3D objects, section A “3D Reconstruction Based on Deep Learning” focuses on the three common representations, including voxels, point clouds and mesh [16], [17]. Finally, the generation method of the grid is investigated in our work. In addition, the related applications of attention mechanisms are also described in section B “Attentional Mechanisms”.

A. 3D RECONSTRUCTION BASED ON DEEP LEARNING

Voxel reconstruction. Choy et al. [18] first proposed a 3D reconstruction method based on voxel representation, the 3D recurrent reconstruction neural network(3D-R2N2), which is now widely used as a baseline for comparison. 3D-R2N2 uses RNNs to fuse the feature maps of the input images, solving the problem of inconsistent single-view and multi-views reconstruction methods, enabling refinement of the generated 3D shapes when more views are input. However, it also has some drawbacks. One is that for a given sequence of images, the reconstruction results of RNNs-based methods are not consistent when their input order is different; second, due to the long-term memory loss of RNNs, they cannot fully utilize the input images, making the reconstruction results poor. Third, 3D-R2N2 has high computational complexity and can only generate low-resolution voxels due to the limitation of computer hardware, thus losing much detail information.

To make the quality of the generated shapes better while avoiding multi-stage training, Liu et al. [19] proposed Variational Shape Learner (VSL). VSL uses short connections to combine some local latent variables to form a global latent variable. Finally, all the local variables are cascaded with the global variables to represent the encoded shape. In their work, the reconstruction results of VSL trained jointly on all species are significantly worse than those after training on individual species, which indicates that VSL does not learn the potential representation of objects well. Xie et al. [20] proposed Pix2Vox++, their contribution is to propose a context-aware fusion network, which refines the detail of the generated voxels based on the learned fusion scores. Pix2Vox++ relies on the object-centered coordinates to align multi-view features. However, object-centered coordinates encourage the network to memorize observed meshes, which may lead to poor generalization abilities. In order to use the shape prior to learn the generic shape representation for agnostic categories, Zhang et al. [21] proposed the generalized reconstruction framework GenRe (Generalizable Reconstruction). Although GenRe hallucinates the unseen parts of these shape primitives, it fails to exploit global shape symmetry to produce correct predictions. This is not surprising given that their network design does not explicitly model such regularity. A possible future direction is to incorporate priors that facilitate learning high-level concepts such as symmetry. To solve the problem of topology perception in 3D shape reconstruction, Chen et al. [22] addressed it by adding two topological
properties to the reconstruction pipeline, namely connectivity and genus (number of holes), and proposed the topology-aware autocoder TPWCoder. The ShapeNet dataset does not have much topological information, therefore TPWCoder possibly does not well refine local details of shapes, which shows that TPWCoder is more dependent on the topological information. Yan et al. [23] explored single-view 3D reconstruction by a learning agent’s perspective and proposed the Perspective Transformer Network (PTN). PTN uses 2D supervision, 3D supervision achieve slightly better performance than it. This can be attributed to the fact that 2D-based supervision methods use loss functions that are based on 2D binary masks and silhouettes. However, multiple 3D objects can explain the same 2D projections. The work of Tulsiani et al. [24] is similar to Yan, however, relies on multi-view supervision. However, sometimes multi-view corresponding to an object are not easily obtained. Moreover, while their approach allows to bypass the availability of ground-truth 3D information for training, a benchmark dataset is still required for evaluation which may be challenging for scenarios like scene reconstruction.

The voxel-based approach is simple and intuitive, and can use the knowledge of deep learning to analyze and process the model directly. However, as the size of the object increases and the resolution increases, deep learning models can usually only handle low-resolution voxel due to the hardware conditions of the computer, and thus many geometric details are lost. In addition, voxels inside the shape occupy memory and computational resources without affecting the final appearance. There are limitations in voxel-based generation, therefore, the research community is actively exploring other forms of representation, such as point clouds.

**Point cloud reconstruction.** Fan et al. [25] proposed point set generation network (PSG) for the first explicit single-view 3D reconstruction in terms of point cloud representation. In PSG, the role of random variables is used to model the uncertainty of the reconstruction, so multiple candidate 3D point clouds are generated. Then, from the multiple candidate 3D point clouds, the one closest to the real 3D data is selected as the final reconstruction result. But how to represent disordered data and how to deal with ambiguity poses difficulties for generating 3D point cloud. Kurenkov et al. [26] obtained the 3D structure of the target object by deforming the template shape. However, their method does not work well for reconstructing circular and less detailed objects, and in some cases it is not even possible to infer the 3D shape of the object. To generate dense point clouds, Lin et al. [27] used generators and proposed a method to generate 3D shapes in the form of dense point cloud representation based on a 2D convolutional network. The model does not use the 3D structure of the object as supervision, but uses the contour information of the object and the depth map as supervision. But their method can be more problematic when objects contain very thin structures (e.g. lamps). Similar to Lin’s work, Insaafutdinov and Dosovitskiy [28] proposed an architecture that also uses 2D projections for supervision. But this architecture is proposed based on specific classes and learns from a set of unlabeled images. In contrast to Lin, Mandikal and Radhakrishnan [29] proposed a hierarchical approach to reconstruct dense point clouds using deep pyramidal networks. Due to outlier points in the sparse point cloud get aggregated in the dense reconstruction, certain predictions have artifacts consisting of small cluster of points around some regions. Lu et al. [30] recently proposed an attention-based approach to generate dense point clouds from the input single view, and this framework is considered as an improvement to the work of Mandikal. To fit more categories of objects, they need to improve the attention mechanism in the future.

The advantage that point cloud only captures the surface features of shape provides a effective representation for 3D shape. However, the unstructured nature, lack of connectivity, and irregularity of point cloud leads to it cannot be easily processed by applying deep learning models. Besides, the generated point cloud needs post-processing before it can be applied to virtual reality, robotics and other fields. However, the representation of mesh can be applied to various fields without post-processing, which attracts researchers’ attention to mesh-based representation 3D reconstruction.

**Mesh Reconstruction.** The model proposed by Kar et al. [31] is one of the pioneering works in 3D reconstruction from a single image in a mesh representation. The model is proposed based on specific class and from several annotated images to learn a deformable shape model that captures shape changes within the class. Some of their major failure modes include not being able to capture the correct scale and pose of the object and thus badly fitting to the silhouette in some cases. Their subtype prediction also fails on some instances (e.g. CRT vs flat screen “tvmonitors”) leading to incorrect reconstructions. Groueix et al. [32] proposed AtlasNet, their goal is to learn directly to reconstruct 3D meshes from a single image or point cloud, and the main contribution is the design of a decoder that feeds potential representations into the decoder and outputs parametric surface elements. But AtlasNet has poor generalization performance, such as when not trained on chairs, AtlasNet seems to struggle to define clear thin structures, like legs or armrests, especially when they are associated to a change in the topological genus of the surface. Wang et al. [33] first used Graph Convolution Networks (GCN) for 3D reconstruction based on mesh representation and proposed Pix2Mesh. The work of GEOMetrics proposed by Smith et al. [34] is similar to Pix2Mesh, which improves on the work of Wang in terms of loss functions, mesh adaption and vertex information update. Both methods are restricted to generating meshes with the same topology as the initial mesh, which limits the utilization of them, future research directions include addressing the restrictive constant topology prescribed by the initial mesh object through reconstruction and generation methods.

In recent years, many researchers turn to the field of combining differentiable renderer with neural network [35], [36], [37]. In the unsupervised shape reconstruction work of
Rezende et al. [38], they incorporated the OpenGL renderer into a neural network for 3D mesh reconstruction and computed the gradients of the OpenGL renderer using REINFORCE [39]. However, the gradients obtained in this way are not too accurate. Henderson and Ferrari [40] proposed the first probabilistic generation framework for 3D mesh reconstruction that uses only 2D supervision, which need less information to complete the reconstruction. But the reconstruction results of this method are not as good as those of the 3D supervised-based method. Kato et al. [12] used hand-crafted functions to approximate the inverse gradient, while using the standard graphics renderer directly during forward propagation, this approach may lead to uncontrolled optimization. The SoftRas proposed by Liu et al. [13] can be used for both forward and backward propagation and is dedicated to provide a more accurate differentiable renderer that truly approaches the discrete and non-differentiable rasterization steps. However, their approach cannot handle shadows and topology changes, which are worth investigation in the future.

Voxel-based representation make it difficult to generate subvoxel-level 3D objects, and thus some detailed information about the objects is lost. In addition, the processing of empty voxels occupies memory and computational resources. Point-based representation suffers from unstructured nature, lack of connectivity and irregularity, which leads to the point-based generation method not being easily processed by directly applying deep learning models. In addition, the generated point clouds need to be post-processed before they can be applied to real scenes. The mesh-based representation, on the other hand, requires less number of parameters and less memory requirements. In addition, meshes are suitable for geometric transformations, where the rotation, translation and scaling of objects are represented by simple operations on the vertices. It also works better in terms of overall smoothness and quality of the shape when representing objects. Therefore, this paper investigates the method of furniture object generation based on mesh representation.

B. ATTENTIONAL MECHANISMS

Selective visual attention in humans greatly improves the efficiency and accuracy of visual information processing. Attention in deep convolutional neural networks is a simulation of selective visual attention, and the core purpose is to expect the convolutional neural network to pay attention to feature information that is more useful for the task being performed, while not allocating much attention to other non-critical information, and sometimes even simply ignoring other unimportant information.

The effectiveness of the attention mechanism has been demonstrated in many tasks. In machine translation, since one or a few words of one language are input corresponding to one or a few words of another language. Therefore, it is not reasonable to encode each input as a fixed-length vector and assign the same weight to different inputs. To solve this problem, Bahdanau et al. [41] introduced an extended model to improve the accuracy of translation by performing translation and alignment simultaneously. Instead of encoding the entire input sentence all as a fixed-length context vector, their work encodes the input sentence as a sequence of vectors and adaptively selects a subset of these vectors at decoding time.

To address the question that visual question and answer models neglect the modeling of object relationships in image, Cheng et al. [42] proposed the relational reasoning mode Graph Attention Network Relation Reasoning(GAT2R) by introducing a graph attention mechanism. GAT2R model includes question feature extraction, scene graph generation, scene graph update part, multimodal fusion and answer prediction. Among them, the scenario graph update part uses a question-guided graph attention network to updated graph node representation dynamically. Thus, providing more accurate predictions for the subsequent answer prediction.

Understanding the position and size of the same structure of an object in different views is important for the 3D reconstruction of furniture objects. Therefore, the SK module and CA module are introduced to make fuller use of the input image features and thus improve the reconstruction performance of the model in this paper.

III. METHOD OVERVIEW

Inspired by the work of NMR [12] and Liu et al. [13], this paper introduces the differentiable SoftRas and proposes the Mesh_CA model, whose overall framework is shown in Figure 1. It consists of two parts: the mesh generator and the differentiable renderer. The mesh generator generates the corresponding 3D structure of the object. Then it is passed into the differentiable renderer SoftRas, and the loss calculation of the rendered output silhouette with the real silhouette is performed using the contour loss function, so that the reconstruction of furniture objects can be achieved by 2D supervision only.

![FIGURE 1. Single view mesh reconstruction flowchart.](image)

A. MESH GENERATOR

Firstly, an initialized ellipsoid is predefined for each input furniture image, and the ellipsoid is an isotropic sphere with 642 vertices; then, the 3D object of the target is generated by deforming the ellipsoid. The mesh generator consists of two parts: encoder and decoder. The structure of the mesh generator is shown in Figure 2.

**Encoder.** The encoder used in this paper is the same as ResNet18 except that the last fully connected layer is removed. The input images are processed by the encoder, and the output feature vector is $1 \times 1 \times 512$. 
**Decoder.** The decoder consists of two fully connected layers with channel numbers of 1024, 2048 respectively. The features extracted by the encoder are fed to the decoder and the output is a displacement vector for each vertex, thus deforming a template mesh to the target object.

**B. DIFFERENTIABLE RENDERER**

Inspired by the work of Liu et al. [13], this paper introduces the differentiable renderer SoftRas to build the model structure. Compared to NMR [12], SoftRas has consistency between forward and backward propagation, thus providing a high-quality gradient flow to realize this process from 2D to 3D. The structure of SoftRas is shown in Figure 3.

As shown in figure 3, where P is the camera, L is the lighting condition, M is the grid, A is the appearance of the vertices, N is the grid normal, U is the image space coordinates, Z is the visual correlation depth, D is the probability map, and I is the rendered output 2D image. As in OpenDR [43], the gradient from the rendered image I to the vertices in the grid M is computed as:

\[
\frac{\partial I}{\partial M} = \frac{\partial I}{\partial U} \frac{\partial U}{\partial M} + \frac{\partial I}{\partial Z} \frac{\partial Z}{\partial M} + \frac{\partial I}{\partial N} \frac{\partial N}{\partial M} \quad (1)
\]

The \(\frac{\partial U}{\partial M}\), \(\frac{\partial Z}{\partial M}\), \(\frac{\partial N}{\partial M}\) and \(\frac{\partial I}{\partial Z}\) are easily obtained by inverting the projection matrix and the illumination model. \(\frac{\partial U}{\partial M}\) and \(\frac{\partial I}{\partial Z}\) do not exist in the conventional rendering channel. By introducing the probability map D as an intermediate representation, \(\frac{\partial I}{\partial U}\) is decomposed into \(\frac{\partial I}{\partial M}\) and \(\frac{\partial I}{\partial Z}\), so \(\frac{\partial I}{\partial U}\) is differentiable. Further, \(\frac{\partial I}{\partial Z}\) is obtained by the aggregation function.

1) **PROBABILITY MAP**

The influence of the triangular surface slice \(f_i\) on the image plane is simulated using the probability map \(D_j\), it is defined as in equation (2).

\[
D_j = \text{sigmoid} \left[ \frac{\delta_j^d (i,j)}{\sigma} \right] \quad (2)
\]

where \(\sigma\) is a positive scalar that controls the sharpness of the probability distribution, \((i,j)\) is the coordinate of pixel \(P_i\), and \(d(i,j)\) is the closest distance from triangular face slice \(f_j\) to pixel \(P_i\). \(\delta_j^d\) is a symbolic representation that maps pixels inside and outside \(f_j\) to (0.5,1) and (0,1) respectively. The value of \(\delta_j^d\) is 1 when \(P_i\) lies inside \(f_j\).

The Euclidean distance is used for compute \(d(i,j)\), let \(t_j^i \in R^3\) be the center of gravity coordinates of the point on the edge of \(f_i\) closest to \(P_j\). Then the signed Euclidean distance \(D_E(i,j)\) from \(P_i\) to the edge of \(f_j\) is calculated as follows.

\[
D_E(i,j) = \delta_j^d \left\| U_j t_j^i - P_i \right\|^2_2 \quad (3)
\]

where \(\delta_j^d\) is the symbolic denotation and has a value of 1 when \(P_i\) lies within the triangular surface slice \(f_i\).

Therefore, \(\frac{\partial D_E(i,j)}{\partial U_j}\) can be obtained by the following equation:

\[
\frac{\partial D_E(i,j)}{\partial U_j} = 2\delta_j^d (U_j t_j^i - P_i) (t_j^i)^T 
\]

where \(t_j^i = U_j^{-1} P_i\),

\[
U_j = \begin{bmatrix} x_1 & x_2 & \cdots & x_3 \\ y_1 & y_2 & \cdots & y_3 \\ 1 & 1 & \cdots & 1 \end{bmatrix}, \quad P_i = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}, \quad P_i
\]

FIGURE 2. Structure of mesh generator.

FIGURE 3. Structure of SoftRas.
2) AGGREGATION FUNCTION

The aggregation function $A_i$ is used to merge the color map $C_j$ to obtain the rendered output $I$ based on the probability map $D_j$ and the depth $Z_j$, which bridges the relationship between the 2D plane and the 3D space. The aggregation function is calculated as in equation (5):

$$I^i = A_i\left(\{C_j\}\right) = \sum_j w_j^i C_j + w_b^i C_b$$

(5)

where $w_j^i$ is calculated as in equation (6):

$$w_j^i = \frac{D_j^i \exp \left( \frac{z_j^i}{\gamma} \right)}{\sum_k D_k^i \exp \left( \frac{z_k^i}{\gamma} \right) + \exp \left( \frac{z_b}{\gamma} \right)}$$

(6)

In the above two equations: $C_b$ is the background color; $w$ is the weight, satisfying $\sum w_j^i + w_b^i = 1$; $z_j^i$ is the normalized depth of the 3D point on the triangular surface slice $f_i$ with 2D projection $p_i$. $z_j^i = \frac{z_{j\text{surf}} - z_{j\text{base}}}{z_{\text{surf}} - z_{\text{base}}}$; $\epsilon$ is a constant; $\gamma$ is the sharpness of the control aggregation function, the default value is 0.0001; $k=1,2,3,\ldots,j$.

\[ \frac{\partial I}{\partial D_j} \text{ and } \frac{\partial I}{\partial D_b} \text{ can obtain by the following equation:} \]

\[ \left\{ \begin{array}{l}
\frac{\partial I}{\partial D_j} = w_j^i \left( C_j - I \right) \\
\frac{\partial I}{\partial D_b} = w_b^i \left( C_b - I \right)
\end{array} \right. \]

(7)

The contour of an object is independent of its color and depth map, so the aggregation function $A_O$ of the contour is further explored based on equation (5), is calculated as below:

$$I^i_c = A_o\left(\{D_j\}\right) = 1 - \prod_j \left(1 - D_j^i\right)$$

(8)

Equation (8) models the contour as the probability of having at least one triangular surface covering pixel $p_i$. $\frac{\partial I^i_c}{\partial D_j}$ is calculated as follows:

$$\frac{\partial I^i_c}{\partial D_j} = \frac{1 - I^i_c}{1 - D_j^i}$$

(9)

C. LOSS FUNCTION

The generating function $G(x)$ consists of three parts: contour loss $L_c$, the smoothing loss $L_{sm}$ and the Laplace loss $L_{lap}$. $G(x)$ is calculated as in equation (10):

$$L = L_c + a L_{lap} + \beta L_{sm}$$

(10)

where: contour loss $L_c$ represents the difference between the reconstructed contour and the true contour, $L_c = 1 - \frac{\sum i \Delta x_i}{\sum i \Delta x_i + \sum i \Delta y_i}$; smoothness loss $L_{sm}$ is used to ensure that the intersection angle of all faces is close to 1800 and acts as a regularizer, $L_{sm} = \sum_{\theta_i \in \epsilon} (\cos \theta_i + 1)^2$, $\theta_i$ is the angle between two faces, and $\epsilon$ is the set of all edges in $G(x)$.

IV. EXPERIMENTS

This section firstly introduces the experimental environment, data sets and evaluation metrics; secondly, it validates the effectiveness of the introduced module and compares the proposed model with some current classical methods; finally, it investigates the generalization ability of the model.

A. EXPERIMENTAL SETUP

The dataset used in this paper is five classes of furniture objects in the ShapeNet dataset, bench, cabinet, chair, sofa, and table respectively. With the same camera settings and illumination settings, the objects are rendered from 24 azimuthal angles and a fixed height (30°) using blender. The output image size is $64 \times 64$ pixels. As in Yan [23] work, we have used a training set of 70%, a validation set of 10% and a test set of 20%.

In order to quantitatively evaluate the reconstruction performance of the network, the real and generated meshes are voxelized and the voxel size is set to $32^3$ to calculate the Intersection over Union (IOU) between voxels, which is calculated as in equation (11):

$$IOU = \frac{\sum_{i,j,k} [1(p_{i,j,k} > \beta) I(y_{i,j,k})]}{\sum_{i,j,k} [1(p_{i,j,k} > \beta) I(y_{i,j,k})]}$$

(11)

where: $(i,j,k)$ denotes the location of the voxel, $p_{i,j,k}$ denotes the probability value of the presence of the voxel and obeys the Bernoulli distribution $\{1 - p_{i,j,k}, p_{i,j,k}\}$, $y_{i,j,k}$ denotes the corresponding true output with values belonging to $0, 1$. $I(.)$ is the indicator function and $t$ is the set voxelization threshold. If $p_{i,j,k}$ is greater than $t$, a voxel exists at that position; conversely, the position is an empty voxel. A higher IOU value indicates a better reconstruction.

B. TRAINING DETAILS

The activation function used is Leaky-relu in this paper. Compared with sigmoid and tanh activation function, it is easy to compute and converge faster because there is no power function. In addition, compared with Relu, Leaky-relu does not lose the negative information of the input.

For a fair comparison, the number of iterations is 250,000, which is consistent with the work of Liu et al. The parameters of the hybrid loss function are set to $\alpha = 5 \times 10^{-4}$, $\beta = 5 \times 10^{-4}$ (Same as Kato et al.). The optimizer used is ADAM, which uses the official recommended value (learning rate $\alpha = 0.0001$, $\beta_1 = 0.9$, $\beta_2 = 0.999$). The network runs on Ubuntu 18.04 with a GeForce GTX 1080Ti.

C. MODULE VALIDATION

In order to verify the effectiveness of the introduced modules, three sets of comparison experiments are conducted and the results are shown in Table 1. In the table: Mesh_SK indicates the introduction of SK module in the encoder; Mesh_CA indicates the introduction of CA module in the encoder. The number of modules used is 2 and the positions are located on both sides of ResNetLayer1. Mesh_SC indicates the introduction of SK and CA modules in the encoder, and the
positions are located after Layer1 and layer4 respectively; mIOU indicates the mean Intersection over Union (mIOU) of the five categories of the corresponding model. The bold numbers represent the good results.

**TABLE 1. Validation of CA modules.**

| Methods  | bench  | cabinet | bench  | sofa  | bench  | mIOU  |
|----------|--------|---------|--------|-------|--------|-------|
| Mesh_SC  | 0.5313 | 0.7166  | 0.5659 | 0.7100| 0.5183 | 0.6084|
| Mesh.SK  | 0.5361 | 0.7124  | 0.5673 | 0.7083| 0.5245 | 0.6098|
| Mesh.CA  | 0.5377 | 0.7130  | 0.5734 | 0.7156| 0.5187 | 0.6117|

It is found from the table that Mesh_CA achieves better results. This indicates that it is more useful to focus on the position of the object than on the size of the object for furniture 3D reconstruction.

Some images are selected from the ShapeNet test set randomly to reconstruct, the number of views used is one. The visualization reconstruction results are shown in Figure 4. The first column is the input single image; the second column is the reconstruction result of Mesh_SC; similarly, the third column and the fourth column are the reconstruction results of Mesh.SK and Mesh_CA respectively.

**FIGURE 4. Reconstruction results of three groups.**

As in Figure 4, the chair surface reconstructed by Mesh_CA is more flat. Besides, the concave and convex structure of the bottom of the table can also be well captured.

This illustrates that the way of extracting features from horizontal and vertical directions separately and then aggregating the features in both directions enables the network to better understand the structure of the target object and better reconstruction the furniture object. Therefore, in the work of this paper, the coordinate attention (CA) module is chosen to be introduced in the encoder to improve the performance of the model.

**D. RESULTS OF FURNITURE OBJECT RECONSTRUCTION**

The proposed model Mesh_CA is compared with the recently proposed SoftRas and some current methods, as shown in Table 2.

**TABLE 2. Experimental results on shapenet test set.**

| Methods          | bench  | cabinet | chair  | sofa  | table  | mIOU  |
|------------------|--------|---------|--------|-------|--------|-------|
| Retrieval[44]    | 0.4875 | 0.5713  | 0.3512 | 0.5314| 0.3997 | 0.4502|
| NMR[12]          | 0.4998 | 0.7143  | 0.4990 | 0.6735| 0.4829 | 0.5739|
| SoftRas[13]      | 0.5080 | 0.7116  | 0.5270 | 0.6878| 0.4487 | 0.5766|
| Mesh_CA(ours)    | 0.5385 | 0.7117  | 0.5716 | 0.7175| 0.5206 | 0.6120|

In Table 2:

- **Retrieval[44]:** Yan designed a encoder-decoder network framework, which allows 3D reconstruction of the input image in voxels without 3D information as supervision by introducing a 2D contour loss function based on perspective transformation as supervision.
- **NMR[12]:** Kato used a hand-crafted function to approximate the inverse gradient, while using a standard graphics renderer directly during forward propagation. Contour loss and smoothing loss were used in their work for supervision.
- **SoftRas [13]:** In the work of Liu et al., they studied two versions of the network model, one trained with contour loss and the other with contour and shading supervision. In this paper, we compared with their first work, which is trained with contour information as supervision.

As can be seen from Table 2, compared with SoftRas and NMR, the overall mIOU of the model proposed in this paper is increased by about 6.1% and 6.6% respectively. In addition to the cabinet reconstruction, the performance of the model proposed in this paper has been greatly improved when reconstructing other types of models.

Some furniture images in the ShapeNet test set are randomly selected for reconstruction and compared with the SoftRas model, and the results are shown in Figure 5. The first row is the input single image, the second row is the reconstruction results of SoftRas, and the third row is the reconstruction results of Mesh_CA in this paper.

As shown Figure 5, the model proposed in this paper can predict the 3D structure of the object well (e.g., three legs for the table in the third column); the visual effect of the reconstruction is also improved by using the Laplace prediction loss and the smoothing loss function. The reconstruction results of the last two columns for the chair as well as the cabinet are also more flat. However, when reconstructing the bench (the second column), the hollow structure of the backrest of the bench is not reconstructed.

**E. MODEL GENERALIZATION PERFORMANCE ANALYSIS**

Same as Yang et al. [23], the generalization performance of the model was investigated in order to further verify the effectiveness of the Mesh_CA. Specifically, in the first group of experiments, the model was trained on chair; in the second...
This confirms Mesh_CA has the ability to learn the general situations (i.e. only one category is used as the training set). It shows the proposed method on unknown object classes. It reveals the proposed method in the third group of experiments, the model was trained on the test set for testing. The reconstructed visual results are shown in Table 3. The bold numbers represent the categories in the third group of experiments, the model was trained on bench; and in the third group of experiments, the model was trained on table. Then, the three groups of experiments were tested on five types of furniture objects. The experimental results are shown in Table 3. The bold numbers represent the categories with the best results in different groups.

### TABLE 3. Experimental results on shapenet test set.

| Category | SoftRas | Mesh_CA |
|----------|---------|---------|
|          | First   | Second  | Three  | First   | Second  | Three  |
| bench    | 0.3789  | 0.4943  | 0.3887 | 0.3943  | 0.5052  | 0.4052 |
| cabinet  | 0.4336  | 0.4349  | 0.6017 | 0.4594  | 0.4333  | 0.6117 |
| chair    | 0.5605  | 0.2326  | 0.2943 | 0.5672  | 0.2485  | 0.3145 |
| sofa     | 0.5701  | 0.5655  | 0.5221 | 0.5865  | 0.5723  | 0.5347 |
| table    | 0.2681  | 0.2980  | 0.4786 | 0.2888  | 0.3053  | 0.5064 |
| mIOU     | 0.4422  | 0.4051  | 0.4569 | 0.4592  | 0.4129  | 0.4745 |

In order to better analyze the above three groups of experiments, several furniture pictures are randomly selected from the test set for testing. The reconstructed visual results are shown in Figure 6.

**FIGURE 5. Experimental results on shapenet test set.**

**FIGURE 6. Research on generalization of model.**

By studying the reconstruction performance of the model on unknown object classes. It shows the proposed method can still predict credible 3D shapes even in very challenging situations (i.e. only one category is used as the training set). This confirms Mesh_CA has the ability to learn the general 3D potential features of the object, rather than simply fitting a function for the training set.

### V. CONCLUSION

This paper presents a 3D reconstruction method of furniture object based on differentiable renderer, which named Mesh_CA. Mesh_CA uses only the contour information of the object as the supervision, can realize the 3D reconstruction of furniture object. Training and testing Mesh_CA on the five types of furniture in shapenet and comparing with the current mainstream methods NMR and SoftRas, results show that the mIOU is about 6.6% and 6.1% higher respectively. Finally, generalized experiments show Mesh_CA has good generalization performance. However, the model can only deform the initial template mesh to generate a specific topology, which means it can only reconstruct specific class of objects. In addition, the generalization performance of the model needs to be further improved if it is applied to a real scene. Therefore, the future research directions of this paper include: improving the generalization performance of the model and generating 3D objects with arbitrary topology.

### ACKNOWLEDGMENT

(Lin Jiang and Meng Tong contributed equally to this work.)

### REFERENCES

[1] N. Baka, B. L. Kaptein, M. de Bruijne, T. van Walsum, and J. E. Giphart, “2D-3D shape reconstruction of the distal femur from stereo X-ray imaging using statistical shape models,” Med. Image Anal., vol. 15, no. 6, pp. 840–850, 2011.

[2] I. Kemelmacher-Shlizerman and R. Basri, “3D face reconstruction from a single image using a single reference face shape,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 33, no. 2, pp. 394–405, Feb. 2011.

[3] H. Wang, D. B. Stout, and A. F. Chatziioannou, “Mouse atlas registration with non-tomographic imaging modalities—A pilot study based on simulation,” Mol. Imag. Biol., vol. 14, no. 4, pp. 408–419, Aug. 2012.

[4] C. Ma, L. Chen, and J. Yong, “AU R-CNN: Encoding expert prior knowledge into R-CNN for action unit detection,” Neurocomputing, vol. 355, pp. 35–47, Aug. 2019.

[5] E. H. Huang, R. Socher, C. D. Manning, and A. Y. Ng, “Improving word representations via global context and multiple word prototypes,” in Proc. 50th Annu. Meeting Assoc. Comput. Linguistics, Jeju Island, South Korea, 2012, pp. 873–882.

[6] T. Mikolov, K. Chen, G. Corrado, and J. Dean, “Efficient estimation of word representations in vector space,” 2013, arXiv:1301.3781.

[7] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” Commun. ACM, vol. 60, no. 6, pp. 84–90, Jun. 2012.

[8] Q. V. Le, “Building high-level features using large scale unsupervised learning,” in Proc. IEEE Int. Conf. Acoust., Speech Signal Process., May 2013, pp. 8595–8598.

[9] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao, “3D ShapeNets: A deep representation for volumetric shapes,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2015, pp. 1912–1920.

[10] A. X. Chang, T. Funkhouser, L. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su, J. Xiao, L. Yi, and F. Yu, “ShapeNet: An information-rich 3D model repository,” 2015, arXiv:1512.03012.

[11] J. J. Lim, H. Pirsiavash, and A. Torralba, “Parsing IKEA objects: Fine pose estimation,” in Proc. IEEE Int. Conf. Comput. Vis., Dec. 2013, pp. 2992–2999.

[12] H. Kato, Y. Ushiku, and T. Harada, “Neural 3D mesh renderer,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 3907–3916.

[13] S. Liu, T. Li, W. Chen, and H. Li, “A general differentiable mesh renderer for image-based 3D reasoning,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 44, no. 1, pp. 50–62, Jan. 2022.
[16] X. Li, Y. Zhang, and P. Feng, “Colorful 3D reconstruction from a single image based on deep learning,” in Proc. 3rd Int. Conf. Algorithms, Comput. Artif. Intell., Dec. 2020, pp. 1–7.

[17] D. J. Rezende, S. M. A. Eslami, S. Mohamed, P. Battaglia, M. Jaderberg, and N. Heess, “Unsupervised learning of 3D structure from images,” in Proc. 30th Conf. Neural Inf. Process. Syst. (NIPS), New York, NY, USA, 2016, pp. 4996–5004.

[18] R. J. Williams, “Simple statistical gradient-following algorithms for connectionist reinforcement learning,” Mach. Learn., vol. 8, nos. 3–4, pp. 229–256, May 1992.

[19] P. Henderson and V. Ferrari, “Learning to generate and reconstruct 3D meshes with only 2D supervision,” 2018, arXiv:1807.09259.

[20] D. Badrinarayanan, K. Cho, and Y. Bengio, “Neural machine translation by jointly learning to align and translate,” 2014, arXiv:1409.0473.

[21] Y. Miao, W. Cheng, S. He, and H. Jiang, “Research on visual question answering based on GAT relational reasoning,” Neural Process. Lett., vol. 54, no. 2, pp. 1435–1448, Apr. 2022.

[22] M. M. Loper and M. J. Black, “OpenDR: An approximate differentiable renderer,” in Proc. Eur. Conf. Comput. Vis. Cham, Switzerland: Springer, 2014, pp. 154–169.

[23] B. Yang, H. Wen, S. Wang, R. Clark, A. Markham, and N. Trigoni, “3D object reconstruction from a single depth view with adversarial learning,” in Proc. IEEE Int. Conf. Comput. Vis. Workshops (ICCVW), Oct. 2017, pp. 679–688.

---

HUI JIANG received the master’s degree from Xi’an Jiaotong University, in 2022. She is currently a Research Assistant with the Department of Information Science, School of Printing, Packaging and Digital Media, Xi’an University of Technology. At present, she has published an SCI (Corresponding Author). Her research interests include 3D reconstruction, information process, machine vision, and intelligent signal processing.

LIN JIANG is currently pursuing the Graduate degree in signal and information processing with the School of Printing, Packaging and Digital Media, Xi’an University of Technology. Her research interests include target detection and machine vision.

MENG TONG is currently pursuing the Graduate degree in signal and information processing with the School of Printing, Packaging and Digital Media, Xi’an University of Technology. Her research interests include word processing, object detection, and deep learning.

---

YALIN MIAO received the Ph.D. degree from Xi’an Jiaotong University, in 2006. He is currently an Associate Professor in computer science with the Xi’an University of Technology, China. He has authored more than 20 research papers in various journals and conferences and holds seven patents. His research interests include machine vision, pattern recognition, digital media, and intelligent signal processing.