Journal of Network Intelligence ©2021 ISSN 2414-8105 (Online)
Taiwan Ubiquitous Information Volume **6**, Number **4**, November **2021**

3D Point Cloud Reconstruction Based on Deformed Network

Wei Liu

Department of Electronic and Information Engineering
Laiwu Vocational and Technical College
No. 1 Shancai Street, Jinan, Shandong, China
sdutlw@126.com

Xiu-Yan Sun

Department of Mechanical and Electrical Engineering
Laiwu Vocational and Technical College
No. 1 Shancai Street, Jinan, Shandong, China
sunxiuyan0634@163.com

Lin-Lin Tang*

Department of Computer Science and Engineering
Harbin Institute of Technology, Shenzhen
Taoyuan Street, Shenzhen, China
Corresponding author: hittang@126.com

Sachin Kumar

Department of Computer Science and Engineering
Ajay Kumar Garg Engineering College
Delhi-Hapur Bypass Road, Ghaziabad, India
imsachingupta@rediffmail.com

Received August 2021; revised October 2021

Abstract. *With development of deep learning, 3D reconstruction has become more and more popular based on it. For complexity and variability of 3D reconstruction object itself, overall reconstruction quality of 3D reconstruction method based on voxels is not high. 3D reconstruction method based on point cloud has better reconstruction effect than the method based on voxel. In this paper, a fully connected point cloud deformation network and a GraphX-based multi-resolution point cloud deformation network is proposed. Experiments show its efficiency and when IoU is used as evaluation metrics, problem of poor quality evaluation in most voxel-based methods can be solved.*

Keywords: 3D Reconstruction, Point Cloud, Deep Learning

1. **Introduction.** Traditional 3D object reconstruction methods based on monocular vision usually use accurate models [1, 2], or use 2D annotations [2] to assist during reconstruction. But these methods are usually limited to some Specific 3D reconstruction scene. Due to complexity and change of real scenes and data, such models that require assumptions are not effective in practical applications.

With development of deep learning and the emergence of large-scale shape sets, such as the ShpaeNet dataset [3], and progress of data-driven methods, researchers have been interested in methods that imitate human visual system. Since 2015, scholars have used deep learning to complete 3D reconstruction based on 2D images. At the same time, representation method of the 3D reconstruction results plays a vital role in choice of architecture of 3D reconstruction network based on deep learning [4], this affects quality and efficiency of 3D reconstruction results. For 3D reconstruction based on monocular vision, since useful information is basically enriched in the surface of the 3D shape or the area near surface, voxel-based method often causes unnecessary waste. Point cloud is a common method to represent 3D surface.

Fan et al. [5] designed a point set generation network (PSGN), which can generate a target 3D point cloud by inputting a single 2D image. As enlightening study, this method proves the power of 3D representation method of point cloud. Jiang et al. [6] also adopted a similar method, they introduced Geometric Adversarial Loss (GAL) to improve it. Unlike reconstruction network architecture that directly generates a 3D point cloud from a single image, Zeng et al. use depth map generated from a single 2D image as an intermediate expression. Then, they generated depth map into a partial point cloud and a complete point cloud in turn.

Previous methods have a common disadvantage that generated point cloud is relatively sparse. Although shape is similar to tag point cloud, it is difficult to demonstrate surface details of 3D shape due to small number of points in point cloud. In addition, to one-step method of predicting dense point clouds from a single 2D image, gradually improving reconstruction resolution of point clouds is also a commonly used method. Yu et al. [7] proposed a network structure that outputs a dense point cloud from a sparse point cloud, namely punet network, which implements point cloud upsampling. Based on this, Mandikal and Radhakrishnan [8] proposed DensePCR network. It is a deep pyramid network that improves resolution of point cloud in stages. Firstly, it uses a simple encoding and decoding network to output a sparse point cloud with 1024 points. Then the sparse point cloud passes through two dense reconstruction networks is done to increase resolution to 16 times and finally form a dense 3D point cloud. Dense reconstruction network firstly aggregates global and local features from point cloud, and increases number of points to 4 times by copying itself. In order to ensure that the copied points will not produce the same results as the origin, different disturbances will be added between the same points.

Dense reconstruction of point clouds based on deep learning is inseparable from feature extraction of point clouds. Due to disorder of point clouds, convolutional neural network acting between neighboring pixels is difficult to directly use on point clouds. In order to use powerful convolution neural network on point cloud, Li et al. [9] proposed χ-convolution. Main idea of χ-convolution is to make good use of local information like a convolutional neural network, but point cloud does not naturally carry position information on arrangement like pixels, and there is no edge between vertices like a grid. So the first step of χ-convolution is to find K neighborhoods of a certain point through K nearest neighbor algorithm, and then process features of these K points. It is similar to resolution of the 2D feature map in convolutional neural network will gradually decrease, and number of channels will increase. And it is similar to resolution of 2D feature map in convolutional neural network will gradually decrease, and channel number will increase. χ-Convolution will aggregate points in neighborhood and reduce the number of points. It also aggregates features of all points and increases feature dimension, so that the neighborhood of the point is more representative.

Although point clouds have good performance on complex 3D shapes, as an unstructured 3D shape description method, it is difficult for us to use point clouds on regular grids to represent convolutional neural networks well. So, a 3D point cloud reconstruction network based on deformed network is proposed here. We conducted related experiments on

ShpaeNet dataset and demonstrated superiority of 3D point cloud reconstruction network based on the deformed network.

2. **Our Proposed Method.** The point cloud-based 3D reconstruction network is shown in Figure 1 Reconstruction network is mainly composed of three modules: a 2D image coding network, a point cloud feature extraction network, and a point cloud deformation network.

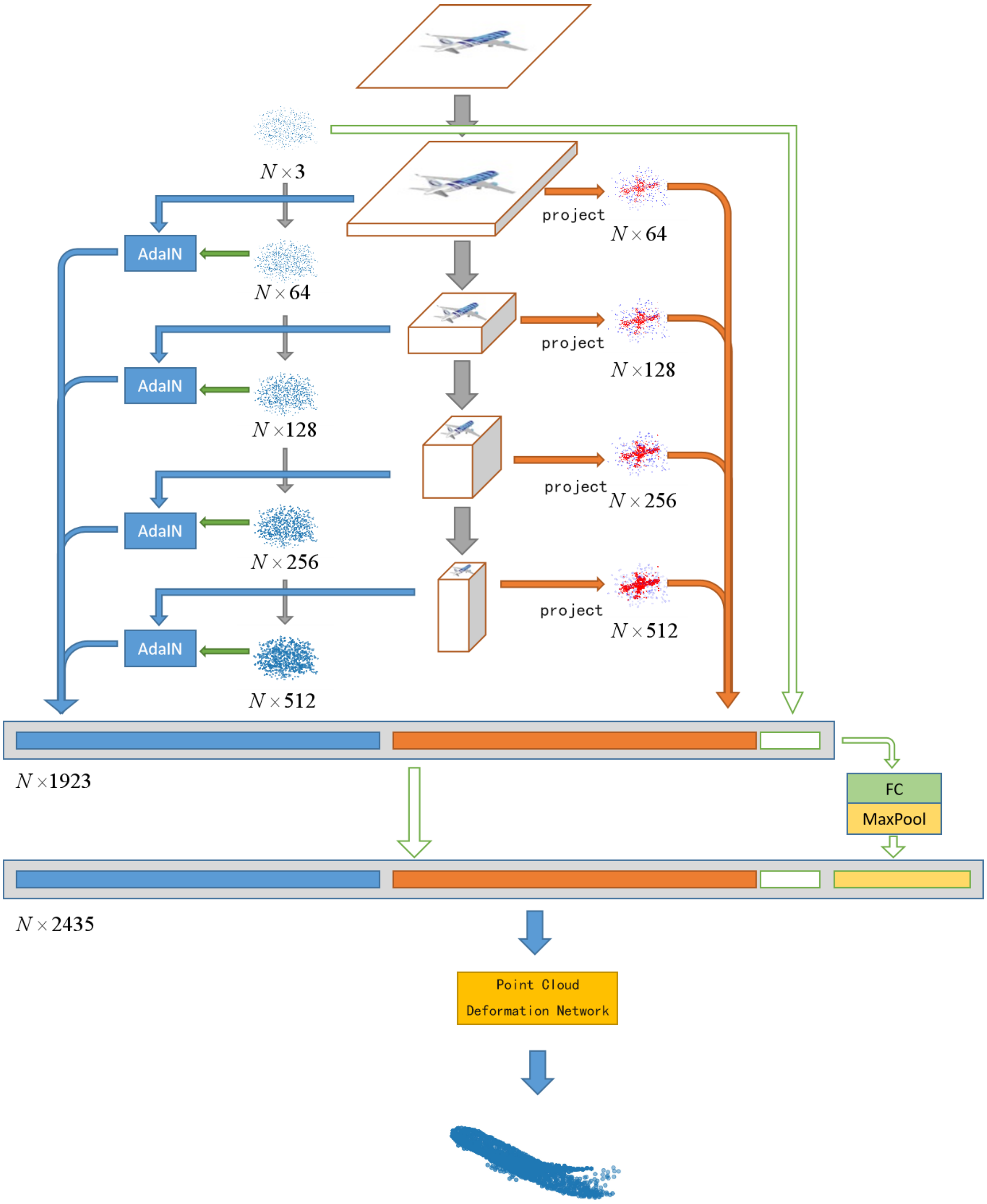

FIGURE 1. Design of 3D Reconstruction Network Based on Point Cloud

For a given 2D image corresponding to a 3D point cloud object, we first use the encoding network of 2D image to encode it, and continuously increase number of channels and reduce resolution through the layered 2D convolution operation. Finally, we extract multi-scale feature map of 2D image. Point cloud feature extraction network uses the important information of 2D coordinates of each point in the initial random point cloud to further perform feature extraction from the multi-scale features extracted from the above-mentioned 2D image encoding network, and extracts local features, style features based on specific points, and global features based on the entire point cloud. In the figure, orange part is the specific point feature, blue part is the style feature, white part is 3D coordinates of the random point cloud, and the yellow part is the global feature.

Then we mix these features with randomly generated initial point cloud, and input them into deformed network, namely the 3D point cloud decoder, to generate the predicted point cloud of the 2D image.

2.1. **Local Features of Point Cloud.** For input random initial point cloud, the only information it has is the 3D coordinates of each point in the point cloud through a random function. With help of camera internal parameters, we can convert 3D coordinates of each point in point cloud into 2D coordinates of feature map output in above-mentioned 2D image coding network, so as to realize the projection of the 3D random point cloud to the 2D feature map. For a real point cloud, if we want to correctly project each point in the point cloud to a 2D image with different angles, we should use camera parameters of 2D image to perform operations such as deformation and rotation on the point cloud. We use inverse process of random point cloud generation to calculate coordinates of each point in point cloud projected to a 2D plane, aMnd then scale these coordinates to 2D feature maps of different sizes. Since 2D coordinates obtained in above calculations are floating-point values and cannot accurately describe the specific pixel coordinates of the feature map, we will use the bilinear interpolation method to calculate feature value with distance as the weight from the four pixels adjacent to the floating point. The coordinates are then used as local feature of each point in point cloud. Through bilinear interpolation, we can add rich specific point features to each point in the point cloud.

2.2. **Stylistic Feature of Point Cloud.** In order to derive global shape information, we obtained a concept from literature on image style transfer [10]. By transferring ”style” of 2D image to point cloud, we can describe global shape information of point cloud to a certain extent, which is called the style feature here. From a global perspective, mean and variance of feature map obtained by multi-level convolution of a 2D image of an object can be used to describe the shape of the object to a certain extent. After retrieving these mean and variance from the multi-scale feature map of 2D input image, eliminate mean and variance in features corresponding to original point cloud, and finally ”embed” mean and variance of 2D image feature map into 3D point, the style transfer is completed. Here, we will use Adaptive Instance Normalization (AdaIN) to transfer the style of the random initial point cloud.

Let $X_i \in R^{c_i \times h_i \times w_i}$ denote feature map numbered obtained by 2D image from 2D image coding network, the number of channels is c_i, the height is h_i, and the width is w_i. Let $Y_i \in R^{N \times c_i}$ represent the dimensional feature obtained by passing the coordinate value of the point cloud through a series of multi-layer perceptrons, and y_j represents the feature vector of a point j in the point cloud at this scale.μ_{X_i} and σ_{X_i} represent mean and variance calculated from the entire , and respectively represent the mean and variance calculated from the entire Y_i. Here we define the 2D to 3D AdaIN formula as shown in formula (1).

$$\text{AdaIN}\left(X_i, y_i\right) = \sigma_{X_i} \frac{y_j - \mu Y_i}{\sigma_{Y_i}} + \mu_{X_i} \tag{1}$$

2.3. **Fully Connected Point Cloud Deformation Network.** As a network structure can guarantee that the result will not become worse, the residual network structure is very popular. For the fully connected point cloud deformation module introduced above, we added a shortcut connection and named it the residual fully connected point cloud deformation module, resFC module for short, to ensure that the fully connected point cloud deformation network can be as fast as possible use point cloud features extracted from 2D images. Its structure is shown in Figure 2.

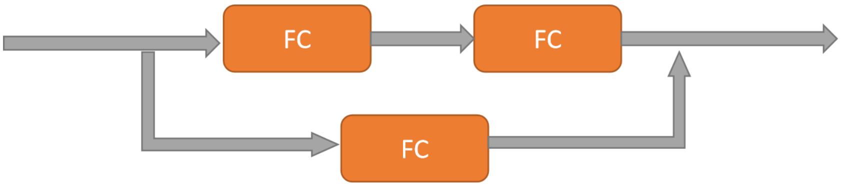

Figure 2. resFC module

2.4. **Multi-resolution Point Cloud Deformation Network.** Graph convolution [11] can take advantage of the local interaction feature between points in point cloud, but graph convolution is applied to the representation of a 3D grid with a topological structure. The point cloud is not like a 3D grid that there is edge information between each point, so graph convolution cannot be used in the point cloud. Literature [12] proposed a point cloud deformation network module called GraphX, as shown in Figure 3. GraphX is a network module similar to MLP-Mixer network [13].

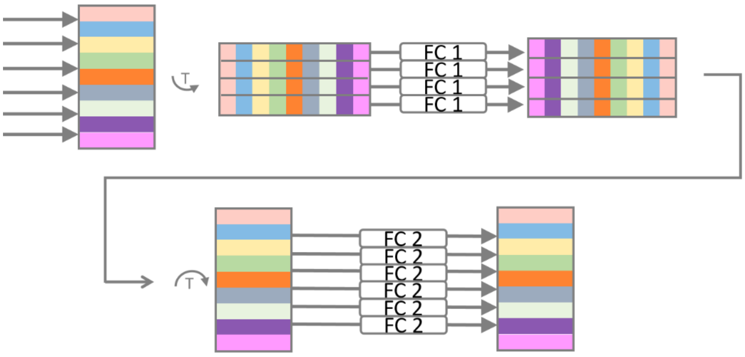

Figure 3. GraphX point cloud deformation module similar to MLP-Mixer structure

For input point cloud features, GraphX module first transposes them, and then uses a fully connected layer for processing. At this time, features in the same dimension between each point in the point cloud will exchange information with each other. By modifying the output dimension of fully connected layer, we can adjust number of points in point cloud. Compared with the upsampling process in Pixel2Mesh [14] that the increase in number of points in the grid is limited by number of edges in grid, and number of edges can only be increased by vertices at a time, using this structure can obviously adjust number of points more flexibly. After that, we transpose point cloud features and perform full connection operation to get the deformed point cloud. χ-convolution and graph convolution act on the neighborhood, while the GraphX deformation operation similar to MLP-Mixer acts on entire point cloud. The mathematical definition of the point cloud deformation module based on GraphX is shown in formula (2):

$$f_k^{(0)} = h\left(n_k\right) = h\left(W^T\left(\sum_{f_i \in F} w_{i,k} f_i + b_k\right) + b\right) \tag{2}$$

$F \in R^d$ represents set of d-dimensional feature vectors of the point cloud, f_i is a feature vector on it, and $f_i^{(0)}$ is output feature vector of the k-th layer. For f_i and $f_i^{(0)}$, $w_{ik}, b_k \in R$ are the mixed weights and mixed deviations obtained by training, $W \in R^{d \times d_0}, B \in R^{d_0}$ are parameters of the fully connected layer immediately after the mixing operation.

In order to ensure that the network can learn more useful feature information, for GraphX-based point cloud deformation module, we also designed its residual version, referred to as resGraphX module for short, and its structure is shown in Figure 4.

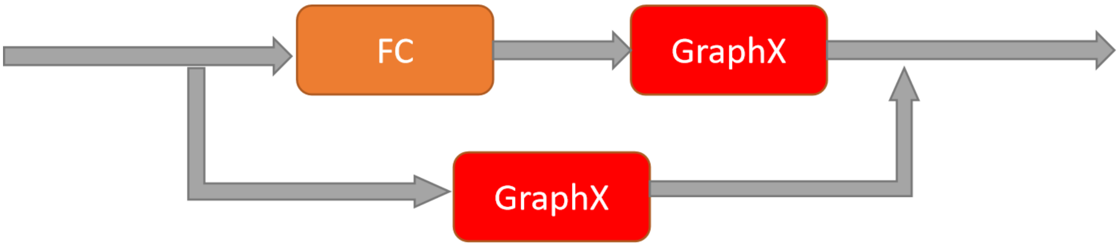

Figure 4. resGraphX module

Based on resGraphX module, we improved GraphX module and proposed a GraphX-based multi-resolution point cloud deformation network(MGXN). Its structure is shown in Figure 5. When the point cloud feature extraction network passes points with features into GraphX-based multi-resolution point cloud deformation network, it will use 3 different resGraphX modules to convert original point cloud into 3 point clouds with different resolutions. Based on these three point clouds, we use resGraphX module to upsample them, and finally get three point clouds with the same number of points. After connecting these three point clouds, and after a series of resGraphX modules, we get the final reconstructed point cloud.

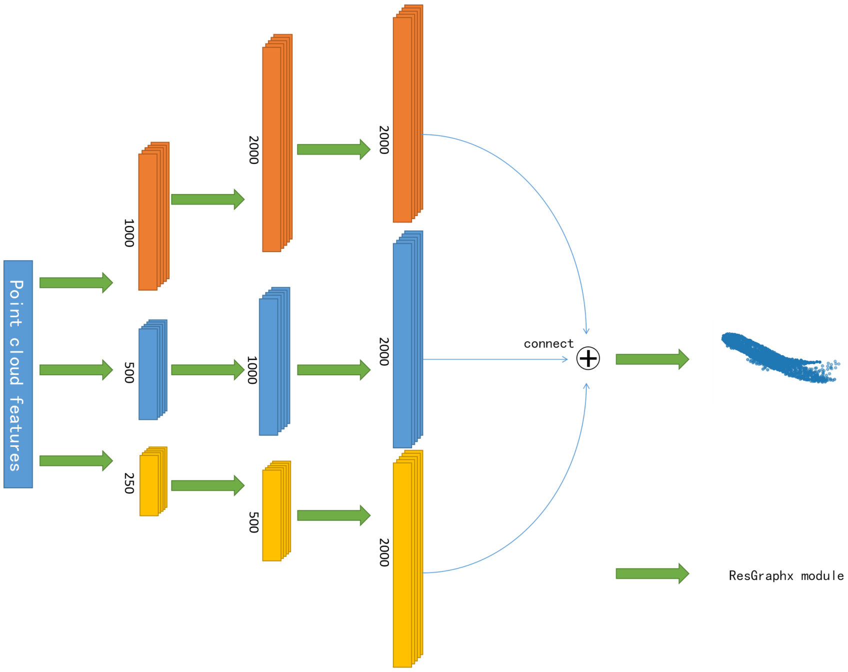

Figure 5. Multi-resolution point cloud deformation network based on GraphX

2.5. **Loss Function.** Since the point cloud is a disordered 3D shape representation, order of the points in the point cloud does not affect representation of 3D shape of the point cloud, so for the 3D point cloud reconstruction network, we need to use a loss function that does not change the relative order of the input points to describe the gap between the predicted point cloud and the real point cloud.

In some algorithms, some scholars also use a loss function similar to L1 loss [15], but the generated 3D shape surface is relatively rough. In the field of 3D reconstruction based on point clouds, we often use earth removal distance (EMD) and chamfer distance (CD) to measure the gap between two point clouds, and use them as a loss in the training process of the 3D reconstruction network function.

Chamfer distance can be used as a measure of the gap between different point clouds. Specific mathematical formula is as follows, let $P \in R^3$ and $Q \in R^3$ denote two different point cloud shapes, then the chamfering distance $d_{CD}(P, Q)$ between the point clouds P and Q is calculated as the formula (3) shown.

$$d_{CD}(P, Q) = \frac{1}{|P|} \sum_{p \in Q} \min_{q \in Q} \|p - q\|_2^2 + \frac{1}{|Q|} \sum_{q \in Q} \min_{p \in P} \|q - p\|_2^2 \tag{3}$$

3. **Experimental Results and Analysis.** In this article, we use a subset of the ShapeNet dataset [16]. 3D-R2N2 provides a rendered 2D image and ground truth point cloud for this ShapeNet subset, which was then processed by Nguyen et al. [12]. The dataset is composed of a total of 43,783 3D point cloud models in 13 categories, and in this article we use the default dataset segmentation method attached to the database.

The calculation formula of IoU is shown in formula (4). IoU represents the ratio of intersection and union between two sets A and B. Extend to binary 3D voxels, it is the ratio of intersection and union between two 3D voxels.

$$\mathrm{IoU}(A, B) = \frac{|A \cap B|}{|A \cup B|} \tag{4}$$

When using the chamfer distance as the evaluation index of the reconstruction quality, we choose 3D-R2N2 [17], PSGN [5], Pixel2Mesh [14] and PCDnet [12] four baselines to compare the experimental results. In the training process of the 3D reconstruction network, Adam optimizer is used, the learning rate is set to 5e-5, and the batchsize to 4. All others use the default settings. The number of points generated in the experiment is 2000. The chamfer distance between the 3D reconstruction result and the tag point cloud is shown in Table 1. Here, the average chamfer distance of each category and the average chamfer distance of all categories is given. Results of PCDnet [12] are obtained from recurring experiments, and the best results have been blacked out.

It can be seen from the Table 1 that 3D reconstruction quality of GraphX-based multi-resolution point cloud deformation network used in this experiment has surpassed the previous best reconstruction method in evaluation index of chamfer distance. In addition, it can be found that the reconstruction quality of the multi-resolution point cloud deformation network based on GraphX completely exceeds fully connected point cloud deformation network [18-20]. The fully connected point cloud deformation network cannot change the number of points in point cloud during point cloud deformation process, while the GraphX-based multi-resolution point cloud deformation network will continuously sample the point cloud during the point cloud deformation process. Even if we increase the number of points in the initial random point cloud to increase features of points that reach point cloud deformation network through point cloud feature extraction network, connect point cloud deformation network still can't have the reconstruction performance beyond the multi-resolution point cloud deformation network based on GraphX.

Table 1. Quantitative comparison of chamfer distance between point cloud based 3D reconstruction network and four baselines on 13 main categories of shapelnet dataset

Category	3D-R2N2 [17]	PSGN [5]	Pixel2Mesh [14]	PCDnet [12]	FCnet	MGXN
airplane	0.895	0.430	0.477	0.123	0.136	**0.119**
bench	1.819	0.629	0.624	0.201	0.234	**0.195**
cabinet	0.735	0.439	0.381	0.264	0.314	**0.263**
car	0.845	0.333	0.268	0.190	0.228	**0.187**
chair	1.432	0.645	0.610	0.316	0.359	**0.309**
monitor	1.707	0.722	0.755	0.251	0.293	**0.248**
lamp	4.009	1.193	1.295	**0.528**	0.572	0.529
speaker	1.507	0.756	0.739	**0.413**	0.484	**0.413**
firearm	0.993	0.423	0.453	0.124	0.136	**0.122**
couch	1.315	0.549	0.490	0.262	0.305	**0.256**
table	1.116	0.517	0.498	0.295	0.336	**0.287**
cellphone	1.137	0.438	0.421	**0.157**	0.190	0.158
watercraft	1.215	0.633	0.670	0.212	0.243	**0.210**
mean	1.445	0.593	0.591	0.257	0.295	**0.254**

When using IoU as the evaluation index of reconstruction quality, we choose PSGN [5], GAL [6], PCDnet [12] and the 3D reconstruction method based on shape layer four baselines to compare experimental results. PCDnet are obtained from recurring experiments. The larger the IoU value, the better, and the best result has been blackened.

Table 2. Quantitative comparison of point cloud based 3D reconstruction network with IOU (%) of four baselines on 13 main categories of shapenet dataset

Category	PSGN [5]	GAL [6]	PCDnet [12]	Shape Layer method	MGXN
airplane	60.1	68.5	73.4	61.8	**73.6**
bench	55.0	70.9	**72.5**	61.7	71.2
cabinet	77.1	77.2	78.1	**80.9**	78.3
car	83.1	73.7	83.3	**83.9**	83.4
chair	54.4	**70.0**	66.3	49.7	65.9
monitor	55.2	**80.4**	73.4	50.7	73.6
lamp	46.2	**67.0**	53.2	55.5	51.2
speaker	73.7	69.8	**70.8**	69.0	70.6
firearm	60.4	71.5	74.9	55.1	**75.2**
couch	70.8	73.9	77.0	67.3	**77.1**
table	60.6	**71.4**	60.4	53.5	60.1
cellphone	74.9	77.3	**85.4**	82.9	84.9
watercraft	61.1	67.5	**75.4**	53.5	75.1
mean	64.0	71.2	**72.6**	64.3	72.3

It can be seen from Table 2 that the 3D reconstruction quality of the point cloud-based monocular vision 3D reconstruction network used in this experiment is very close to the current best result in the IoU evaluation metrics.

For qualitative analysis, several 2D images are selected to display reconstruction results. Reconstruction results are shown in Figure 6. The left side is the target 2D image, the middle is the target tag point cloud, and the right is the reconstruction point cloud.

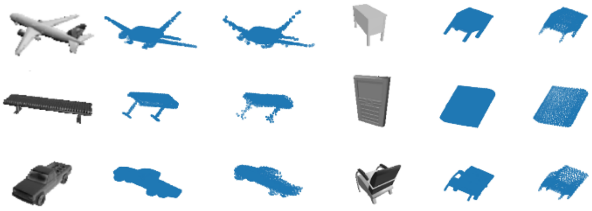

Figure 6. Display of 3D reconstruction results of some objects

4. **Conclusions.** By improving the extraction of specific point features, style features and global features of the point cloud, a 3D point cloud reconstruction network based on a deformed network is proposed. Through experimental evaluation and ablation experiments, we demonstrate the effectiveness of 3D point cloud reconstruction network based on deformed network in monocular 3D reconstruction. When the chamfer distance is used as the evaluation index, this reconstruction network can obtain better results than other baselines.

Acknowledgment. This work was supported by Shenzhen Science and Technology Plan Fundamental Research Funding JCYJ20180306171938767 and Shenzhen Foundational Research Funding JCYJ20180507183527919.

REFERENCES

[1] V. Blanz, T. Vetter, A morphable model for the synthesis of 3D faces, *The 26th Annual Conference on Computer Graphics and Interactive Techniques*, pp. 187-194, 1999.

[2] I. Kemelmacher-Shlizerman, Internet based morphable model, *IEEE International Conference on Computer Vision*, pp. 3256-3263, 2013.

[3] A. X. Chang, T. Funkhouser, L. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su, J. Xiao, Shapenet: An information-rich 3d model repository, *arXiv preprint arXiv:1512.03012*, 2015, https://ui.adsabs.harvard.edu/abs/2015arXiv151203012C.

[4] K. Fu, J. Peng, Q. He, H. Zhang, Single image 3D object reconstruction based on deep learning: A review, *Multimedia Tools and Applications*, vol. 80, no. 1, pp. 463-498, 2021.

[5] H. Fan, H. Su, L. J. Guibas, A point set generation network for 3d object reconstruction from a single image, *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 605-613, 2017.

[6] L. Jiang, S. Shi, X. Qi, J. Jia, Gal: Geometric adversarial loss for single-view 3d-object reconstruction, *European Conference on Computer Vision*, pp. 802-816, 2018.

[7] L. Yu, X. Li, C. W. Fu, D. Cohen-Or, P. A. Heng, Pu-net: Point cloud upsampling network, *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2790-2799, 2018.

[8] P. Mandikal, V. B. Radhakrishnan, Dense 3d point cloud reconstruction using a deep pyramid network, *IEEE Winter Conference on Applications of Computer Vision*, pp. 1052-1060, 2019.

[9] Y. Li, R. Bu, M. Sun, W. Wu, X. Di, B. Chen, Pointcnn: Convolution on x-transformed points. *Advances in Neural Information Processing Systems*, vol. 31, pp. 820-830, 2018.

[10] L. A. Gatys, A. S. Ecker, M. Bethge, Image style transfer using convolutional neural networks, *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2414-2423, 2016.

[11] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, G. Monfardini, The graph neural network model. *IEEE Transactions on Neural Networks*, val. 20, no. 1, pp. 61-80, 2008.

[12] A. D. Nguyen, S. Choi, W. Kim, S. Lee, GraphX-convolution for point cloud deformation in 2D-to-3D conversion, *IEEE/CVF International Conference on Computer Vision*, pp. 8628-8637, 2019.

[13] I. Tolstikhin, N. Houlsby, A. Kolesnikov, L. Beyer, X. Zhai, T. Unterthiner, J. Yung, A. Steiner, D. Keysers, J. Uszkoreit, M. Lucic, A. Dosovitskiy, Mlp-mixer: An all-mlp architecture for vision, *arXiv preprint arXiv:2105.01601*, 2021, https://ui.adsabs.harvard.edu/abs/2021arXiv210501601T.

[14] N. Wang, Y. Zhang, Z. Li, Y. Fu, W. Liu, Y. G. Jiang, Pixel2mesh: Generating 3d mesh models from single rgb images, *Proceedings of the European Conference on Computer Vision*, pp. 52-67, 2018.

[15] V. Golyanik, S. Shimada, K. Varanasi, D. Stricker, Hdm-net: Monocular non-rigid 3d reconstruction with learned deformation model, *International conference on Virtual Reality and Augmented Reality*, pp. 51-72, 2018.

[16] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, J. Xiao, 3d shapenets: A deep representation for volumetric shapes, *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1912-1920, 2015.

[17] C. B. Choy, D. Xu, J. Y. Gwak, K. Chen, S Savarese, 3d-r2n2: A unified approach for single and multi-view 3d object reconstruction, *European Conference on Computer Vision*, pp. 628-644, 2016.

[18] K. Wang, S. P. Xu, C. M. Chen, M. M. Hassan, C. Savaglio, P. Pace, G. Aloi, A Trusted Consensus Scheme for Collaborative Learning in the Edge AI Computing Domain, *IEEE Network*, vol. 35, no. 1, pp. 204-210, 2021.

[19] K. Wang, P. Xu, C. M. Chen, S. Kumari, M. Shojafar, M. Alazab, Neural architecture search for robust networks in 6G-enabled massive IoT domain, *IEEE Internet of Things Journal*, vol. 8, no. 7, pp. 5332-5339, 2020.

[20] S. Kumar, A. Damaraju, A. Kumar, S. Kumari, C. M. Chen, LSTM Network for Transportation Mode Detection, *Journal of Internet Technology*, vol. 22, no. 4, pp. 891-902, 2021.