IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received March 23, 2020, accepted May 1, 2020, date of publication May 6, 2020, date of current version May 18, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2992554

3D-ReConstnet: A Single-View 3D-Object Point Cloud Reconstruction Network

BIN LI, YONGHAN ZHANG, BO ZHAO, AND HONGYAO SHAO
School of Computer Science, Northeast Electric Power University, Jilin City 132012, China

Corresponding author: Bin Li (libinjlu5765114@163.com)

This work was supported in part by the Science and Technology Development Plan Project of Jilin Province under Grant 20180520017JH, and in part by the Science and Technology Project of the Jilin Provincial Education Department of China under Grant JJKH20170107KJ.

ABSTRACT Object 3D reconstruction from a single-view image is an ill-posed problem. Inferring the self-occluded part of an object makes 3D reconstruction a challenging and ambiguous task. In this paper, we propose a novel neural network for generating a 3D-object point cloud model from a single-view image. The proposed network named 3D-ReConstnet, an end to end reconstruction network. The 3D-ReConstnet uses the residual network to extract the features of a 2D input image and gets a feature vector. To deal with the uncertainty of the self-occluded part of an object, the 3D-ReConstnet uses the Gaussian probability distribution learned from the feature vector to predict the point cloud. The 3D-ReConstnet can generate the determined 3D output for a 2D image with sufficient information, and 3D-ReConstnet can also generate semantically different 3D reconstructions for the self-occluded or ambiguous part of an object. We evaluated the proposed 3D-ReConstnet on ShapeNet and Pix3D dataset, and obtained satisfactory improved results.

INDEX TERMS 3D reconstruction, point cloud, uncertainty in reconstruction, 3D neural network.

I. INTRODUCTION

Reconstructing the shape of 3D objects from a single-view is the fundamental task of robot navigation and grasping, CAD, virtual reality and so on. Therefore, data-driven 3D object reconstruction has attracted more and more attention.

At present, there are two kinds of 3D object representations: voxel and point cloud. The voxel-based neural networks [1]–[3] can reconstruct 3D objects by generating voxelized three-dimensional occupancy grids. However, voxel representation suffers from two problems: sparse information and high computational complexity, especially in high resolution 3D object processing. In order to make up for the deficiency of voxel expression, Fan *et al.* [4] proposed point cloud-based 3D object reconstruction which is a deep learning method to study point cloud generation. The 3D point cloud of an object is composed of three-dimensional points uniformly sampled from the surface of the object. Point cloud model has scalability and flexibility, so we use point cloud as our 3D representation.

The difficulties of 3D point cloud reconstruction are: 1. When a 2D input contains enough information, the reconstruction network needs to infer an accurate 3D reconstruction; 2. When a 2D input is ambiguous or uncertain, the reconstruction network needs to reconstruct multiple plausible reconstructions 3D output for the 2D input. As shown in Figure 1 (a), the view of the chair in the two 2D images provides enough information for reconstruction.

FIGURE 1. Single-view reconstruction for an unambiguous 2D input.

FIGURE 2. Multiple plausible reconstructions for an ambiguous 2D input.

The associate editor coordinating the review of this manuscript and approving it for publication was Genoveffa Tortora.

Figure 2 (a) is the self-occluded view of the chair in Figure 1 (a). It is unreasonable to predict a single deterministic output for an ambiguous input. In this work, we propose a neural network called 3D-ReConstnet for single-view 3D point cloud reconstruction. The 3D-ReConstnet uses residual network to extract a feature vector from 2D input, and uses probability distribution learned from the feature vector to predict 3D point cloud. The 3D-ReConstnet can generate the determined 3D output as shown in Fig. 1 (b) for a 2D image with sufficient information (such as Figure 1 (a)), but in the case of uncertainty or ambiguity in the input 2D image (such as Figure 2 (a)), 3D-ReConstnet can generate multiple plausible reconstructions as shown in Figure 2 (b)∼(d).

In summary, our contributions in this work are as follows:

1) We propose an end-to-end 3D point cloud reconstruction network: 3D-ReConstnet. The end-to-end network structure enables 3D-ReConstnet to infer 3D point cloud directly from 2D image features, avoiding the feature propagation across the network like those multi-stage network [18], and avoiding the loss of features.

2) For an ambiguous 2D input, our 3D-ReConstnet can generate multiple plausible 3D reconstructions from a single input image.

3) We evaluated 3D-ReConstnet on ShapeNet and Pix3D datasets. The experimental results show that 3D-ReConstnet outperforms the state-of-art reconstruction methods in the task of single view 3D reconstruction.

The rest of this paper is organized as follows: Section II introduces the related work. In Section III, we introduce the 3D-ReConstnet in detail. In Section IV, we evaluate the 3D-ReConstnet on ShapeNet and Pix3D dataset. Section V concludes this paper.

II. RELATED WORKS

A. SINGLE-VIEW 3D RECONSTRUCTION

The traditional 3D reconstruction method [5]–[7] needs multiple view correspondence. As a result, single-view 3D reconstruction has more advantages than traditional methods. Single-view point cloud reconstruction can be roughly divided into voxel-based 3D reconstruction and point cloud-based 3D reconstruction.

Voxel-based 3D reconstruction. As described below, a number of works have based on voxel representations. Choy *et al.* [1] trained a recurrent neural network to learn the mapping from 2D image to 3D output from a large number of synthetic data. In [8], a 3D local shape generation method is proposed. This method infers a low resolution but complete output by using a 3D encoder, and associates the output with the 3D graphics in the shape database to obtain 3D voxel reconstruction. Tulsiani *et al.* [9] proposed an unsupervised 3D voxel reconstruction neural network trained by multi-view observations of unknown poses. Shubham *et al.* [10] explored the way to reconstruct 3D outputs by using different 2D view projections, such as depth maps, color images, image semantics and so on. Although several studies [11], [12] are devoted to solve the two defects of voxel: sparse information and high computational complexity, and have achieved some good results, the defects of voxel are still obvious compared with point cloud.

Point cloud-based 3D reconstruction. Fan *et al.* [4] first proposed a 3D reconstruction method based on point cloud. In this method, Chamfer distance (CD) and Earth Mover's distance (EMD) were chosen as loss functions to train an autoencoder point cloud generation network, and multiple plausible reconstructions can be generated for ambiguous input by variational autoencoder [14], [15]. In [16], a segmented and point cloud reconstruction network: 3D-PSRNet was proposed. In the training process, 3D-PSRNet propagates the segmented or reconstruction information to another task, and uses the CD and location aware segmentation loss as the loss function. The main contribution of the work in [17] is that the author proposed geometric adversarial loss with two components: geometric loss and conditional adversarial loss. Geometric loss is responsible for ensuring that the shape of 3D reconstruction is close to ground-truth, while conditional adversarial loss generates a semantically-meaningful point cloud. Mandikal *et al.* [18] proposed a two-stage point cloud reconstruction network: 3D-LMNet. First, 3D-LMNet uses chamfer loss to train a point cloud auto-encoder. Then, 3D-LMNet uses diversity loss and latent matching loss to map the vector of auto-encoder to a probability distribution to solve the problem of uncertain 2D input. In [19], a deep pyramid network for generating dense 3D point clouds was proposed. The pyramid network is trained by CD and EMD loss to predict a low-resolution point cloud. Then, the low-resolution point cloud becomes a high-resolution point cloud through dense reconstruction network. Chen *et al.* [20], proposed a Point Auto-Encoder, which is implemented based on the novel semi-convolutional and semi-fully-connected layers proposed that can handle the problem of mapping from single global feature vector to massive numbers of 3D points. All the related work of point cloud-based 3D reconstruction is devoted to two problems in 3D point cloud reconstruction: 1. Design a better point cloud reconstruction neural network. 2. Choose a more suitable loss function. Only by putting forward better solutions to the above two problems, can we reconstruct more accurate 3D output for 2D input with sufficient information and reasonable output with multiple possibilities for uncertain 2D input.

III. APPROACH

A. ARCHITECTURE OF 3D-ReConstnet

The architecture of the 3D-ReConstnet is shown in Figure 3. Our 3D-ReConstnet is an end-to-end neural network. The end-to-end network architecture enables the semantic features of 2D images to be transferred only within the network, rather than across the network like 3D-LMNet [18], thus reducing the loss of features. The 3D-ReConstnet has three main tasks: 2D input feature extraction, sampling a probabilistic vector, point cloud generation. The depth neural

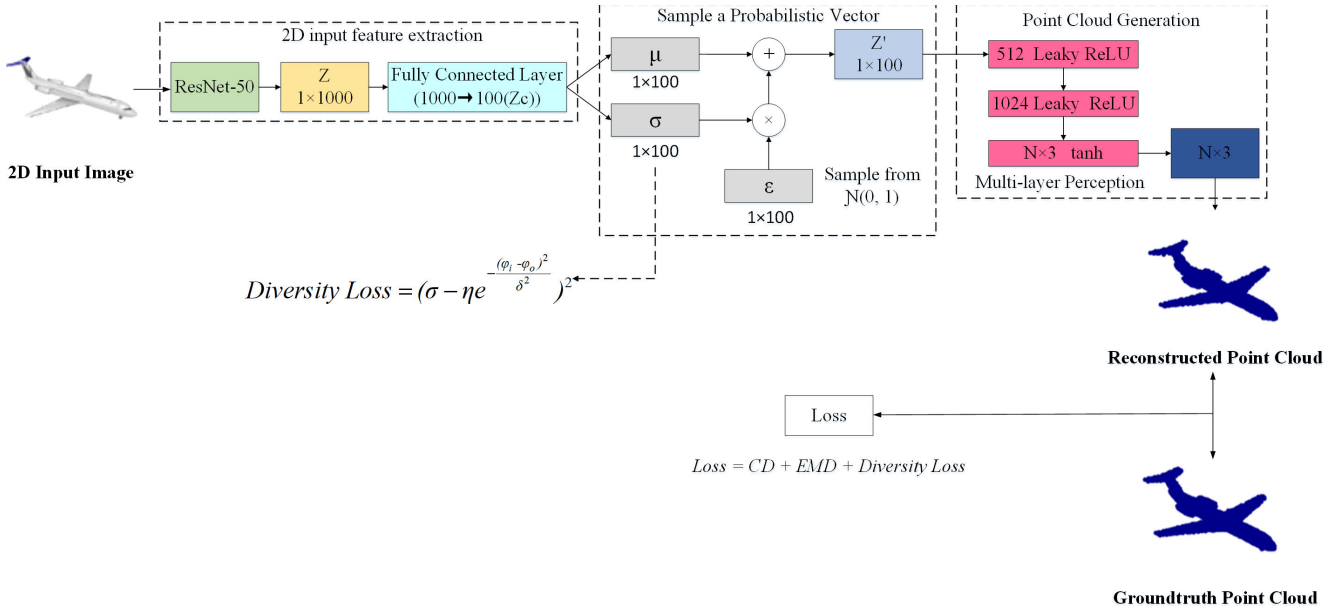

FIGURE 3. The architecture of 3D-ReConstnet.

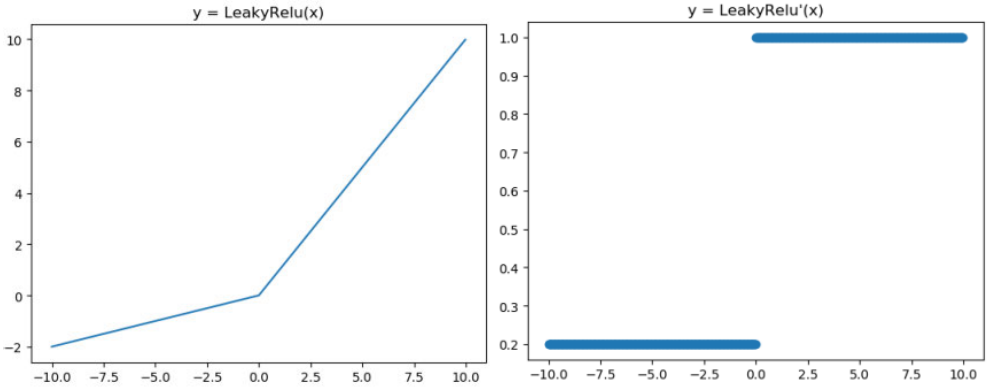

FIGURE 4. Leakey ReLU and derivative of Leakey ReLU.

network ResNet-50 is used to extract the features of 2D images. The addition of residual makes the deep neural network ResNet-50 easy to train, and it can extract sufficient semantic features of 2D images without vanishing gradient. The 3D-ReConstnet first uses the residual network ResNet-50 proposed in [21] to extract the features of 2D input image and gets a feature vector Z. After that, the full connection layer compresses the dimension of vector Z from 1000 to 100, and obtains vector Z_c.

We learn a probabilistic distribution from the vector Z_c in order to generate multiple possibility 3D shapes for uncertain 2D input. We map the vector Z_c of a specific 2D input to a Gaussian vector Z', i.e. $Z' \sim \mathcal{N}(\mu, \sigma^2)$. We use the ''reparameterization trick'' of Variational Auto-Encoders [14], [15] to deal with the randomness in the network. The network in the middle dotted box in Figure 3 is responsible for predicting the mean μ and standard deviation σ of Z_c, sampling $\varepsilon \sim \mathcal{N}(0, 1)$, and finally obtaining the Gaussian probabilistic vector as $Z' = \mu + \varepsilon\sigma$. The mean μ of Z' is unconstrained, and the standard deviation σ is constrained by ε, so that the uncertain 2D input image can be reconstructed meaningfully and diversely.

We use a multi-layer perceptron(mlp) with two hidden layers and one output layer to transform the probabilistic vector as Z' into point cloud data. The activation function of the two hidden layers is Leaky ReLU [22], and that of the output layer is tanh [23]. The output channels of the two hidden layers are 512 and 1024 respectively. The output channels of the output layer are N×3, where N is the number of points in the point cloud.

Figure 4 (a) and (b) are schematic diagrams of the Leaky ReLU and derivative of Leaky ReLU, respectively. The value

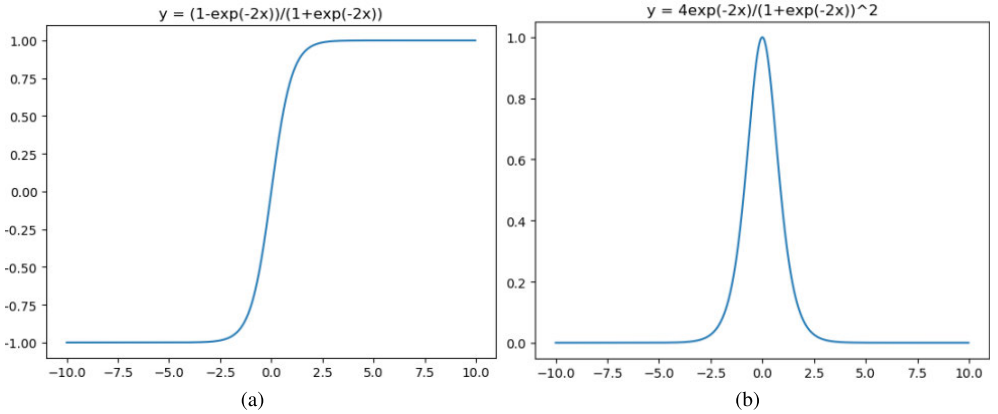

FIGURE 5. Tanh and derivative of tanh.

of the ReLU function on the negative axis is zero. However, the Leaky ReLU function has non-zero values on the negative axis (Figure 4 (a)), and Leaky ReLU also has non-zero derivative values on the negative axis (Figure 4 (b)). Therefore, using Leaky ReLU as the activation function, the negative information in the neural network will not be lost. The range of the normalized point cloud data coordinates are between $[-1,1]$, which indicates that there are many points with negative coordinates. In the process of forward propagation of network information, the coordinate information contained in the negative value will be transferred to the next layer through Leaky ReLU. In the process of network information back propagation, because the derivative value of Leaky ReLU is not zero, the gradient corresponding to the negative value will provide more help for the weight update of the network.

Using tanh as the activation function of the last layer of multi-layer perceptron can quickly fit the generated point cloud data between $[-1,1]$. Figure 5 (a) and (b) are schematic diagrams of tanh and derivative of tanh, respectively. As shown in Figure 5 (a), the tanh activation function can restrict the coordinate value range of the generated 3D point cloud data to $[-1,1]$. The ground truth read by the reconstruction network is normalized, that is, the coordinates of the ground truth are exactly between $[-1,1]$. In the early stage of network training, the activation function tanh can reduce the gap between the generated point cloud data and the ground truth as much as possible, so as to accelerate the fitting speed. However, we use Leaky ReLU as the activation function in the first two layers of MLP instead of tanh. This is because the gradient of neural network may disappear in the process of training, and improper activation function is one of the reasons for vanishing gradient. From Figure 5 (b), it can be seen that the derivative of tanh is 1 when the horizontal axis is 0, and the corresponding derivative values of other positions are less than 1, even in the positive and negative infinite fields, the derivative tends to 0, that is, the derivative of tanh activation function is less than 1 in most cases. When tanh is used as the activation function, the result of chain derivation may approach to 0 as the gradient accumulates, and eventually the vanishing gradient. In order to reduce this risk, we only use tanh as the activation function in the last layer of the mlp.

B. LOSS FUNCTION

The loss function of the 3D-ReConstnet is defined as:

$$Loss = CD + EMD + DiversityLoss \tag{1}$$

where the diversity loss is defined by [18]:

$$DiversityLoss = \left(\sigma - \eta e^{-\frac{(\varphi_i - \varphi_o)^2}{\delta^2}}\right)^2 \tag{2}$$

where φ_o is the azimuth angle of maximum occlusion view, and φ_i is the azimuth angle of the 2D input image. The goal of network training is to minimize the loss. The diversity loss only acts on standard deviation σ of the probabilistic vector Z' sampling network. The smaller the difference between φ_o and φ_i, that is, the larger the occlusion of the 2D input, the greater the value of σ. The larger the value of σ, the more likely the 3D-ReConstnet is to generate multiple plausible reconstructions.

Since point cloud is an unordered representation, we need to use a loss function independent of the relative order of the input points to train the point cloud generation network. Fan *et al.* [4] proposed using Chamfer distance (CD) and Earth Mover's distance (EMD) [25] to train point cloud generation network. This method was widely used in later works [16]–[19].

Let $X_{gt} \in \mathbb{R}^{N\times 3}$ represent ground-truth and $X_{pred} \in \mathbb{R}^{N\times 3}$ represent the generated point cloud, where N represents the number of points in the point cloud. The chamfer distance between X_{gt} and X_{pred} is defined as:

$$d_{Chamfer}(X_{gt}, X_{pred}) = \sum_{X_g \in X_{gt}} \min_{X_p \in X_{pred}} ||X_g - X_p||_2^2 + \sum_{X_p \in X_{pred}} \min_{X_g \in X_{gt}} ||X_g - X_p||_2^2 \tag{3}$$

The chamfer distance measures the square distance between each point in set X_{gt} and its closest point in set X_{pred}.

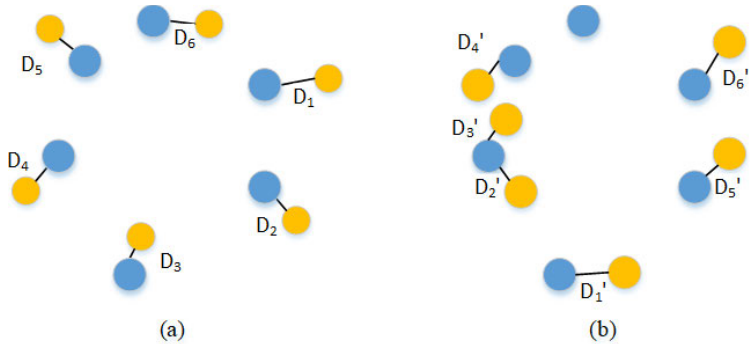

FIGURE 6. Different reconstruction with similar chamfer distance.

The EMD loss between X_{gt} and X_{pred} is defined as:

$$d_{EMD}(X_{gt}, X_{pred}) = \min_{\phi: X_{gt} \to X_{pred}} \sum_{x \in X_{gt}} ||x - \phi(x)||_2 \quad (4)$$

where ϕ is a bijection. The EMD performs a point-to-point mapping between set X_{gt} and set X_{pred}.

In [4], Fan *et al.* shows the Mean-shape behavior of CD and EMD through pictures. After that, in the process of training 3D reconstruction network with CD and EMD, other research work found their characteristics as follows:

1) The chamfer distance is related to the contour of the reconstructed point cloud. The 3D reconstruction network trained by chamfer distance is easier to catch the rough contour of 3D object [4], [26]. However, the reconstruction network trained by CD is easy to generate a splash shape, which blurs the geometry of the reconstructed shape, and chamfer loss may confuse different reconstruction with similar chamfer distance. Figure 6 illustrates the cause of this confusion. Let the blue dot represent the ground-truth and the yellow dot represent the predicted point cloud. Suppose that the yellow dots in Figure 6 (a) and (b) represent two different reconstruction results. $D_1 \sim D_6$ represent the distance between 6 ground-truth points and 6 points obtained from the first reconstruction. $D'_1 \sim D'_6$ represent the distance between 6 ground-truth points and 6 points obtained from the second reconstruction. If the sum of $D_1 \sim D_6$ is equal to the sum of $D'_1 \sim D'_6$, the CD will determine that the two reconstruction results are equal. But this is not the case. In the EMD loss function, ϕ represents the bijection relationship between the truth value and the predicted point cloud, so EMD loss has no defect of reconstruction confusion.

2) The EMD is related to the visual quality of the reconstructed point cloud [4], [26]. The lower the EMD loss, the better the visual quality of 3D reconstruction [26], [27]. However, the reconstructed network trained by EMD is not good at grasping the whole contour of the reconstructed object. We can see that the CD and the EMD loss have their own advantages, so we take the combination of them as the loss function of 3D-ReConstnet. The role of CD is to train the network to form the contour of the reconstructed object, and the role of EMD is to train the network to modify the appearance of the reconstructed object.

IV. EXPERIMENTS

We evaluated the proposed 3D-ReConstnet on the ShapeNet [28] dataset and the Pix3D [29] dataset, respectively. ShapeNet dataset consists of 43809 CAD models in 13 categories. Pix3D dataset consists of 7595 real images and their corresponding metadata (masks, ground truth CAD models and pose). In order to compare with these related works [4], [18], we use the same partition of training set and test set as [1]: the ratio of training set to test set is 4 to 1. We use the training set divided from the ShapeNet to train the 3D-ReConstnet, and carry out 3D reconstruction experiments on the test set divided from the ShapeNet and Pix3D data set respectively.

Implementation Details: 3D-ReConstnet is trained using the Adam optimizer, with batch size of 32 and learning rate 0.00005 for 50 epochs. We crop the size of a 2D input image to 128 × 128 and use it as the input of 3D-ReConstnet. The parameters of ResNet-50 [21] used to extract the features of 2D pictures are shown in Table 1. We just want to extract the features of 2D images, so we don't use softmax at the end of ResNet-50 like [21].

Evaluation Methodology: We use the Chamfer Distance (Chamfer) and Earth Mover's Distance (EMD) calculated on 1024 random sampling points to evaluate the reconstruction quality in all our experiments. We selected three images from each object category in ShapeNet and Pix3D datasets and showed their qualitative 3D reconstruction results in Figure 7-8. Figure 9 show the qualitative 3D reconstruction results of ambiguous 2D input.

A. 3D RECONSTRUCTION ON ShapeNet DATASET

Figure 7 show the qualitative 3D reconstruction results of three images of each object category in ShapeNet. The predicted resolution of 3D point cloud reconstruction in Figure 7 is 2048. Table 2 shows the CD and EMD values of point cloud reconstructed by 3D-ReConstnet(ours), PSGN [4] and 3D-LMNet [18] on ShapeNet dataset. The smaller the values of CD and EMD, the better the reconstruction quality. The values of CD and EMD of 3D-ReConstnet are lower than those of PSGN and 3D-LMNet, while also having lowest mean scores of CD and EMD.

FIGURE 7. Reconstructions on ShapeNet dataset.

TABLE 1. Parameters of ResNet-50.

layer name	output size	parameters
conv1	64×64	7×7 kernel, 64 channels, stride 2
conv2_x	32×32	3×3 max pool, stride 2
		[1×1 kernel, 64 channels 3×3 kernel, 64 channels 1×1 kernel, 256 channels] ×3
conv3_x	16×16	[1×1 kernel, 128 channels 3×3 kernel, 128 channels 1×1 kernel, 512 channels] ×4
conv4_x	8×8	[1×1 kernel, 256 channels 3×3 kernel, 256 channels 1×1 kernel, 1024 channels] ×6
conv5_x	4×4	[1×1 kernel, 512 channels 3×3 kernel, 512 channels 1×1 kernel, 2048 channels] ×3
output	1×1	4×4 average pool, 1000-dimension fully connected layer

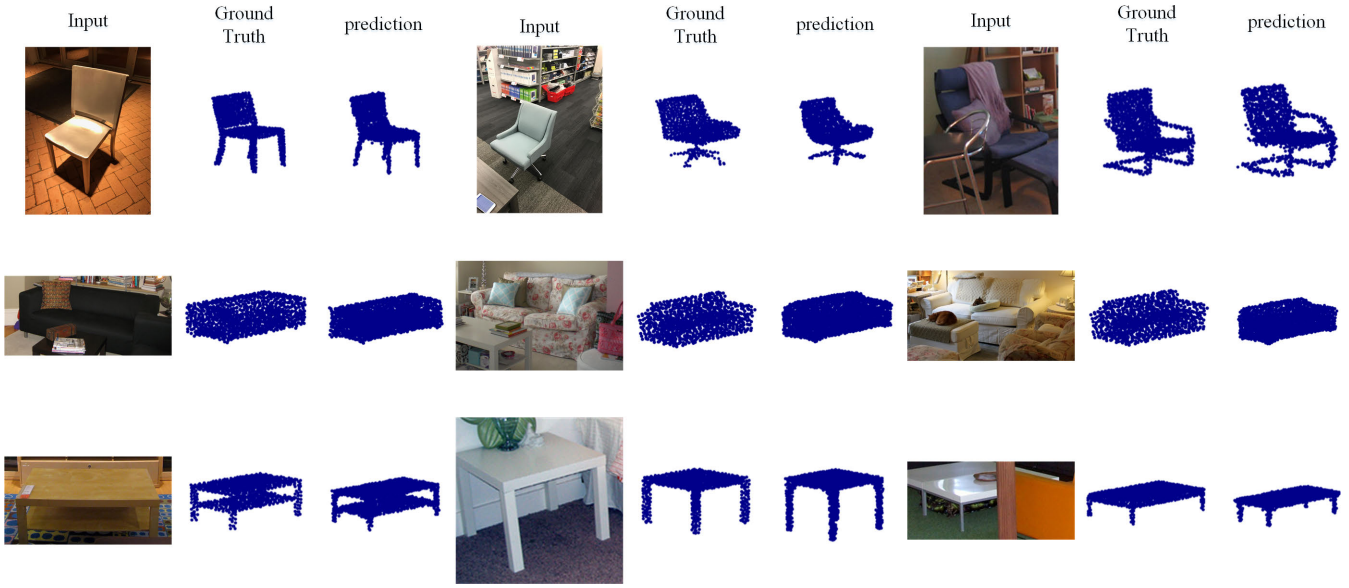

FIGURE 8. Reconstructions on Pix3D dataset.

TABLE 2. Single view reconstruction results on ShapeNet dataset.

Category	CD×10^{-2}			EMD×10^{-2}		
	PSGN	3D-LMNet	Ours	PSGN	3D-LMNet	Ours
airplane	3.74	3.34	**2.42**	6.38	4.77	**2.80**
bench	4.63	4.55	**3.57**	5.88	4.99	**3.22**
cabinet	6.98	6.09	**4.66**	6.04	6.35	**3.84**
car	5.20	4.55	**3.59**	4.87	4.10	**2.87**
chair	6.39	6.41	**4.41**	9.63	8.02	**4.24**
lamp	6.33	7.10	**5.03**	16.17	15.80	**6.40**
monitor	6.15	6.40	**4.61**	7.59	7.13	**4.38**
rifle	2.91	2.75	**2.51**	8.48	6.08	**3.63**
sofa	6.98	5.85	**4.58**	7.42	5.65	**3.83**
speaker	8.75	8.10	**5.94**	8.70	9.15	**5.26**
table	6.00	6.05	**4.41**	8.40	7.82	**4.26**
telephone	4.56	4.63	**3.59**	5.07	5.43	**3.06**
vessel	4.38	4.37	**3.81**	6.18	5.68	**3.99**
mean	5.62	5.40	**4.09**	7.75	7.00	**3.98**

B. 3D RECONSTRUCTION ON Pix3d DATASET

Figure 8 show the qualitative 3D reconstruction results of three images of each object category in Pix3D. The predicted resolution of 3D point cloud reconstruction in Figure 8 is 2048. Table 3 shows the CD and EMD values of point cloud reconstructed by 3D-ReConstnet(ours), PSGN [4] and 3D-LMNet [18] on Pix3d dataset. The smaller the values of CD and EMD, the better the reconstruction quality.

TABLE 3. Single view reconstruction results on Pix3D dataset.

Category	CD×10^{-2}			EMD×10^{-2}		
	PSGN	3D-LMNet	Ours	PSGN	3D-LMNet	Ours
chair	8.05	7.35	**5.59**	12.55	9.14	**5.99**
sofa	8.45	8.18	**6.14**	9.16	7.22	**5.02**
table	10.82	11.2	**7.04**	15.16	12.73	**7.60**
mean	9.11	8.91	**6.26**	12.29	9.70	**6.20**

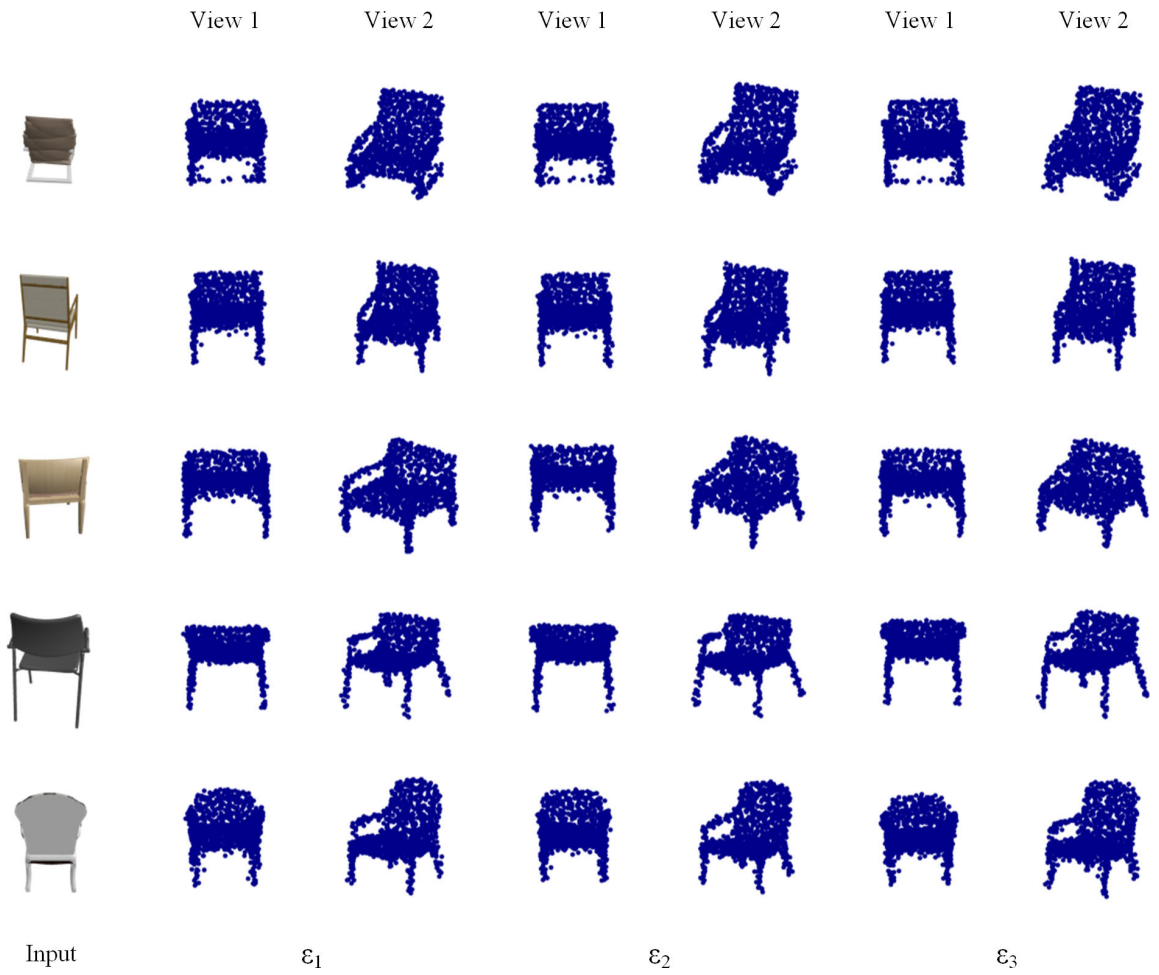

FIGURE 9. Qualitative 3D reconstruction results for ambiguous 2D input.

The values of CD and EMD of 3D-ReConstnet are lower than those of PSGN and 3D-LMNet, while also having lowest mean scores of CD and EMD.

C. GENERATING MULTIPLE PLAUSIBLE OUTPUTS

In this experiment, we select the 2D image with the parameter $\varphi_o = 180°$ in Formula 2 from the chair category of ShapeNet, that is, the back-view image with the maximum occlusion. For each chair image in the back-view, we generated three 3D reconstruction outputs using 3D-ReConstnet. In Figure 9 we show the back and side views of each 3D reconstruction with different ε. The predicted resolution of 3D point cloud reconstruction in Figure 9 is 1024. We show the consistency between reconstruction results and 2D input through back-view, and show the diversity of reconstruction results through side-view. As shown in Figure 9, 3D-ReConstnet can generate semantically different reconstructions which are consistent with the ambiguous input image with the largest occlusion. From Figure 9, we can see that the handle and leg structures of these different reconstruction results are different.

V. CONCLUSION

In this paper, we propose an end-to-end single view 3D reconstruction network: 3D-ReConstnet. The 3D-ReConstnet maps the feature learned from a 2D image to a normally distributed vector to deal with the uncertainty of the self-occluded part of an object. The proposed 3D-ReConstnet can generate the determined 3D output for a 2D image with sufficient information while generate semantically different 3D reconstructions for an ambiguous 2D input. We evaluated 3D-ReConstnet on ShapeNet and Pix3D datasets.

The experimental results show that 3D-ReConstnet outperforms the state-of-art reconstruction methods in the task of single view 3D reconstruction.

REFERENCES

[1] 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 *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 628–644.
[2] J. Gwak, C. B. Choy, M. Chandraker, A. Garg, and S. Savarese, "Weakly supervised 3D reconstruction with adversarial constraint," in *Proc. Int. Conf. 3D Vis. (3DV)*, Oct. 2017, pp. 263–272.
[3] 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.
[4] H. Fan, H. Su, and L. Guibas, "A point set generation network for 3D object reconstruction from a single image," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 605–613.
[5] S. Liu and D. B. Cooper, "Ray Markov random fields for image-based 3D modeling: Model and efficient inference," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 1530–1537.
[6] Haming, Klaus, and Gabriele Peters, "The structure-from-motion reconstruction pipeline-a survey with focus on short image sequences," *Kybernetika*, vol. 46, no. 5, pp. 926–937, 2010.
[7] J. Fuentes-Pacheco, J. Ruiz-Ascencio, and J. M. Rendón-Mancha, "Visual simultaneous localization and mapping: A survey," *Artif. Intell. Rev.*, vol. 43, no. 1, pp. 55–81, Jan. 2015.
[8] A. Dai, C. R. Qi, and M. NieBner, "Shape completion using 3D-Encoder-Predictor CNNs and shape synthesis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5868–5877.
[9] S. Tulsiani, A. A. Efros, and J. Malik, "Multi-view consistency as supervisory signal for learning shape and pose prediction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2897–2905.
[10] S. Tulsiani, T. Zhou, A. A. Efros, and J. Malik, "Multi-view supervision for single-view reconstruction via differentiable ray consistency," in *Proc. IEEE Conf. Comput. Vis. pattern Recognit.*, Jul. 2017, pp. 2626–2634.
[11] G. Riegler, A. O. Ulusoy, and A. Geiger, "OctNet: Learning deep 3D representations at high resolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3577–3586.
[12] C. Hane, S. Tulsiani, and J. Malik, "Hierarchical surface prediction for 3D object reconstruction," in *Proc. Int. Conf. 3D Vis. (3DV)*, Oct. 2017, pp. 412–420.
[13] M. Tatarchenko, A. Dosovitskiy, and T. Brox, "Octree generating networks: Efficient convolutional architectures for high-resolution 3D outputs," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2088–2096.
[14] C. Doersch, "Tutorial on variational autoencoders," 2016, *arXiv:1606.05908*. [Online]. Available: http://arxiv.org/abs/1606.05908
[15] D. P Kingma and M. Welling, "Auto-encoding variational bayes," 2013, *arXiv:1312.6114*. [Online]. Available: http://arxiv.org/abs/1312.6114
[16] P. Mandikal, K. L. Navaneet, and R. V. Babu, "3D-PSRNet: Part segmented 3D point cloud reconstruction from a single image," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 662–674.
[17] L. Jiang, S. Shi, X. Qi, and J. Jia, "Gal: Geometric adversarial loss for single-view 3d-object reconstruction," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2018, pp. 802–816.
[18] P. Mandikal, K. L. Navaneet, M. Agarwal, and R. V. Babu, "3D-LMNet: Latent embedding matching for accurate and diverse 3D point cloud reconstruction from a single image," 2018, *arXiv:1807.07796*. [Online]. Available: http://arxiv.org/abs/1807.07796
[19] P. Mandikal and V. B. Radhakrishnan, "Dense 3D point cloud reconstruction using a deep pyramid network," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2019, pp. 1052–1060.
[20] W. Cheng and S. Lee, "Point auto-encoder and its application to 2D–3D transformation," in *Proc. 14th ISVC*, 2019, pp. 66–78.
[21] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
[22] X. Zhang, Y. Zou, and W. Shi, "Dilated convolution neural network with LeakyReLU for environmental sound classification," in *Proc. 22nd Int. Conf. Digit. Signal Process. (DSP)*, Aug. 2017, pp. 1–5.
[23] B. L. Kalman and S. C. Kwasny, "Why tanh: Choosing a sigmoidal function," in *Proc. Int. Joint Conf. Neural Netw. IJCNN*, Jun. 1992, pp. 578–581.
[24] V. Nair and G. E. Hinton, "Rectified linear units improve restricted Boltzmann machines," in *Proc. 27th Int. Conf. Mach. Learn. (ICML)*, 2010, pp. 807–814.
[25] Y. Rubner, C. Tomasi, and L. J. Guibas, "The earth mover's distance as a metric for image retrieval," *Int. J. Comput. Vis.*, vol. 40, no. 2, pp. 99–121, Nov. 2000.
[26] P. Achlioptas, O. Diamanti, I. Mitliagkas, and L. Guibas, "Learning representations and generative models for 3D point clouds," 2017, *arXiv:1707.02392*. [Online]. Available: http://arxiv.org/abs/1707.02392
[27] L. Yu, X. Li, C.-W. Fu, D. Cohen-Or, and P.-A. Heng, "PU-Net: Point cloud upsampling network," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2790–2799.
[28] 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*. [Online]. Available: http://arxiv.org/abs/1512.03012
[29] X. Sun, J. Wu, X. Zhang, Z. Zhang, C. Zhang, T. Xue, J. B. Tenenbaum, and W. T. Freeman, "Pix3D: Dataset and methods for single-image 3D shape modeling," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2974–2983.

BIN LI was born in Changchun, Jilin, China, in 1982. He received the M.S. and Ph.D. degrees from the School of Computer Science and Technology, Jilin University, China, in 2011 and 2015, respectively. He is currently an Associate Professor with the School of Computer Science, Northeast Electric Power University. His research interests include image processing, computer vision, and pattern recognition.

YONGHAN ZHANG received the bachelor's degree from the Faculty of Software Engineering, Harbin University of Science and Technology, in 2017. He is currently a Graduate Student with the School of Computer Science, Northeast Electric Power University. His research interests include computer vision, image processing, and deep learning.

BO ZHAO received the bachelor's degree from Northeast Electric Power University, in 2017, where he is currently pursuing the master's degree with the School of Computer Science. His research interests include computer vision and deep learning.

HONGYAO SHAO received the bachelor's degree from the Faculty of Computer Science and Technology, Xinyang Normal University, in 2018. She is currently a Graduate Student with the School of Computer Science with Northeast Electric Power University. Her research interests include computer vision, 3D point cloud, and deep learning.