IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received March 26, 2019, accepted April 22, 2019, date of publication April 30, 2019, date of current version May 14, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2914150

RealPoint3D: An Efficient Generation Network for 3D Object Reconstruction From a Single Image

YANG ZHANG, ZHEN LIU, TIANPENG LIU, BO PENG, AND XIANG LI
College of Electronic Science, National University of Defense Technology, Changsha 410073, China

Corresponding author: Zhen Liu (zhen_liu@nudt.edu.cn)

ABSTRACT The generation of 3D models from a single image has recently received much attention, based on which point cloud generation methods have been developed. However, most current 3D reconstruction methods only work for relatively pure backgrounds, which limit their applications on real images. Meanwhile, more fine-grained details are required to provide finer models. This paper proposes an end-to-end efficient generation network, which is composed of an encoder, a 2D–3D fusion module, and a decoder. First, a single-object image and a nearest-shape retrieval from ShapeNet are fed into the network; then, the two encoders are integrated adaptively according to their information integrity, followed by the decoder to obtain fine-grained point clouds. The point cloud from the nearest shape effectively instructs the generation of finer point clouds. To have a consistent spatial distribution from multi-view observations, our algorithm adopts projection loss as an additional supervisor. The experiments on complex and pure background images show that our method attains state-of-the-art accuracy compared with volumetric and point set generation methods, particularly toward fine-grained details, and it works well for both complex backgrounds and multiple view angles.

INDEX TERMS 3D reconstruction, nearest shape retrieval, point cloud generation, single image, projection.

I. INTRODUCTION

With various approaches to obtaining point clouds, 3D reconstruction has witnessed great achievements, particularly in a wide range of applications of deep learning. Considering the ambiguous correspondences between pixels and 3D space points, the projection from 2D to 3D remains notably difficult and intuitive. A single-view image of an object gives incomplete information of a 3D object, for example, the back information of RGBD images is usually missing. One route is to use an existing large dataset to reconstruct 3D objects. However, when taking the various presentations of 3D vision, for example, meshes and volumetric or other regular structures, into account, the reconstructions fail to show more subtle details and extensive sampling resolutions with incremental requirements.

Three-dimensional point clouds, whose characteristics are simple and accessible, become a type of original data format without specific manipulation. In this paper, we propose a new generation network, RealPoint3D, to establish an end-to-end solution of 3D object reconstruction from a single image based on a large-scale existing dataset [1], [2]. Volumetric methods [3]–[6] account for the majority of methods and succeed in addressing the application of 3D scene segmentation, 3D object classification and detection [7]–[9]. Among them, the OGNs (octree generating networks) [5] achieve the best behavior because most existing 3D methods are based on 2D images, which particularly maintain high computational efficiency, low storage requirements and high resolution. However, some revolutionary works related to point cloud processing using deep learning methods appeared in the past 1-2 years. The most significant ones are PointNet [10] and its follower PointNet++ [11]. Point cloud generation methods have been developed based on a single image, such as PSGN (point set generation network) [12].

The current point cloud generation methods have some main limitations: pure backgrounds, the fixed viewpoint and specific distance, which result in poor performance of the model at different views with distortion and loss of details, and the need for a relatively pure background, which causes difficulties in the actual image reconstruction. We are devoted to designing a new architecture and an effective learning paradigm, as shown in Figure 1. The pipeline is goal-oriented: with a given 2D image, we directly project

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

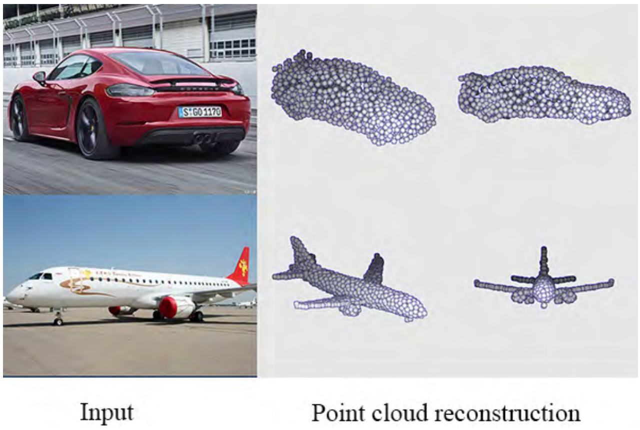

FIGURE 1. **A 3D point cloud reconstruction from a real single image with a complex background. It is visualized from two viewpoints. No segmentation mask is required.**

the pixel information of the object into the 3D space as a point cloud, and calculate the Chamfer distance and projection differences of the generative model and the real model. Specifically, a most similar point cloud from ShapeNet is adopted as an additional input together with the image, which has been proven to be effective and practical for generating finer models.

It is challenging to eliminate the obscurity of diverse views in the projection. Our solution is to render one perspective image for each object and feed the image and similar point cloud into the network in the training phase. At testing stage, a rendered image from different views or a real photo together with a retrieved model is attainable to produce the entire model. Another imminent matter is the complex background in real images. In our experiments, most generative works are strongly affected by the background. We explore the nearest shape templates from a 3D shape database, which can be used to determine the general outline to address the fear of severe distortion. Unlike other RGBD reconstruction tasks, our method can recover the back information from the viewers. We summarize our efforts as follows:

We design a novel network, which is capable to integrate 2d and 3d features adaptively based on a feature fusion module, and our method could reconstruct a 3D object from a real image with a complex background.

A projection supervision scheme is proposed for observing consistent spatial distributions.

Our approach achieves state-of-the-art performance in comparison to volumetric and point cloud generation methods such as OGN and PSGN.

II. RELATED WORK

A. THREE-DIMENSIONAL RECONSTRUCTION FROM SINGLE IMAGES

Lately, the use of generative methods to reconstruct 3D object from a single image has been high profile. However, the problem of 3D structure recovery from a single projection is ill-posed [12], [13]. To address this problem, many early attempts, such as the massive SFM and SLAM [14], [15] methods, were made, all of which required strong presumptions and abundant expertise. ShapeFromX [16]–[20], in which X can be the texture, specularity, shadow, etc., also requires priors on natural images. Boosted by the large-scale dataset of 3D CAD models such as ShapeNet [1], generative methods based on deep learning are emerging. They can be roughly divided into voxel-based methods and point-cloud-based methods. Most generative methods for 3D reconstruction are based on voxel reconstruction.

The 3D-GAN [3] embedded generative task in generative adversarial nets outperforms other unsupervised learning methods by a large margin. Reference [21] explored autoencoder-based networks to learn the latent feature. Reference [22]–[24] added a projection layer to learn the projection from 3D to 2D. In addition, the 3D recurrent neural network (3D-R2N2) [4] and octree generating networks (OGNs) [5] are the most impressive methods. 3D-R2N2 uses long short-term memory (LSTM) to infer 3D models, taking in several images from different perspectives. Derived from the octree representation, an OGN relieves the storage and calculation burden, making it the first generative method applied to large scene reconstruction. The common limitation of such methods is that they are based on the regular structure mimicking 2D convolution operation, which results in inevitable computational waste and loss of original characteristics. In contrast, our method is directly based on raw point clouds to generate and reconstruct the 3D model from a single image.

The work most similar to ours is PSGN [12], which can also directly generate point clouds based on a single image. First, the authors use the Chamfer distance (CD) to calculate the distance in reconstruction, and CD is also used in this paper. The requirement of a pure background, an estimated viewpoint and a specific distance are the restrictive factors of PSGN, whereas our solution eliminates them to some extent.

B. FEATURE EXTRACTION FROM A 3D MODEL

A 3D presentation of both regular and irregular types is usually in a massive format. A main line of research is to extract features from a collection of 2D images using traditional CNNs [25]. Voxels are regular space grids that can be manipulated by, for example, the 2D convolutional operation. Currently, the mainstream approach is to use voxels to slice the space and then apply the 3D convolutional operation. Reference [6], [26] are the pioneers, but their work is confined to a relatively small resolution, leading to a sparse volume. Some works [27] use specific networks on meshes but face difficulties on manifold meshes and for non-isometric shapes.

In contrast, raw 3D data are irregular in that they display discrete points. Few works focus on raw point clouds, which preserve the original information. Reference [28] adopted the attention mechanism and sort point numbers, but this approach lacks the geometry information of the sets.

PointNet [10] is an innovative architecture that can directly distill features from raw point cloud data, and it can be resorted for the classification and segmentation tasks. PointNet++ [11] uses and adjusts on multiple layers or resolutions to extend the receptive fields of PointNet and become elevated. For irregular domains, there is another line of massive works aiming to represent or describe them based on meshes, termed geometric deep learning [29], such as spectral graph CNN [30], Geodesic CNN (GCNN) [31], Laplacian operation [32] and the followers [33], [34].

III. PROBLEM STATEMENT

Our purpose is to generate the complete 3D model of an object from a single image while remaining immune to the disturbance of background and viewpoint. A point cloud model is a collection of scattered points, represented by $P = \{(x_i, y_i, z_i)\}_{i=1}^{N}$, where N is the number of points. We set $N = 1024$ for convenience of comparison with PSGN and other methods. Unlike the voxel-based methods, the original 3D data can be put into our network with no manual handling while benefiting from a superior ability to recover a 3D shape. All points are on the surface of an object.

Our model can be considered a conditional function mapping from a 2D image and the nearest shape to its complete model. The mapping process can be denoted as:

$$P = G\{(I, T; \Phi)\}, \tag{1}$$

where Φ is the network parameter, I is the 2D image, and T is the nearest shape. During the training and testing phase, we search templates for each object that instruct the generation.

IV. APPROACH

A. OVERVIEW

In this section, we will introduce the proposed end-to-end network, namely RealPoint3D, which can efficiently generate a 3D reconstruction model from a single image. Unlike recent single-view 3D reconstruction works, our task of reconstructing the object in complex context is challenging. The proposed algorithm involves several steps: firstly, a nearest 3D shape of the given image is obtained from ShapeNet. Then, the network takes the retrieved point clouds and the image as input and learns to generate the point clouds of the object. In the remaining section, we will introduce this approach in detail, which includes the nearest shape retrieval, novel network RealPoint3D, spatial distance measurement, spatial distribution measurement and implantation details.

B. NEAREST SHAPE RETRIEVAL

For a real image, it is difficult to distinguish the object outline because of the complex background, thus inducing missing details as the 3D point cloud is generated. Inspired by this issue, our solution takes a distinct step from other generative methods. The training dataset that we use is organized as the image and point cloud pairs, thus doing of the combination of 2D and 3D information can alleviate the complex background effect. We search for similar point clouds according to the similarity of the images, measured by the feature maps obtained from the pretrained VGG network. Like other image retrieval practices, the process is instructed by the distance of one or multiple feature maps. Specifically, we use the penultimate feature map and cosine distance in our network. The cosine distance is defined as follows:

$$sim(x, y) = cos\theta = \frac{\vec{x} \cdot \vec{y}}{\|x\| \cdot \|y\|}. \tag{2}$$

where x and y are two images to be compared. The Figures 3, 7 show the retrieval results.

C. REALPOINT3D

To solve the challenge of reconstructing a 3D object with complex backgrounds, we design a novel network to predict the object point clouds. To combine the 2D and 3D information, RealPoint3D has two inputs: the image and the retrieved 3D shape, which closely matches the object in the image. As shown in Figure 2, our network is divided into several parts. In the encoding part, we use 2D CNNs to extract the 2D feature from the image, and we use PointNet++ to obtain the 3D hierarchical feature from the retrieved point clouds. Then, we combine the some 2D and 3D features by an attention-based fusion module. Thus, we obtain more adaptive features that include both image features and spatial features, including 2D RGB, edges and 3D points distribution. Next, the fused feature is fed into the decoding part, where we use the convolutional layers and deconvolutional layers to predict the 3D point clouds of the object. In addition, we learn from the U-Net structure to provide details in the output. The output is a $N \times 3$ matrix, and each row contains the coordinates of one point.

The encoder is a composition of a 2D encoding part and a 3D encoding part. The 2D encoding part has convolutional and ReLU layers, and a single image as an input is stretched as a 1024-dimensional vector. In the 3D encoding part, we take the retrieved shape as another input, which is fed into PointNet++ layers composed by a number of set abstraction layers, including the sampling layer, grouping layer and PointNet layer. PointNet is suitable for unordered point sets, consists of MLP and pooling layers and is designed for classification and semantic segmentation of point clouds. However, PointNet lacks the ability to capture local structures. To overcome this drawback, PointNet++ that consists of alternative hierarchical PointNet components was proposed, which can efficiently and robustly learn deep point set features. In our network, we adopt four set abstraction layers and use the multi-scale grouping strategy to obtain a global feature latent space of the retrieved shape. As a result, we can also obtain a 2048-dimensional global point cloud feature through the 3D encoding part. Then, we merge all features together by feeding them through an attention module, which consists of three fully connected layers to learn relative weights to fuse the two parts. After obtaining the bottleneck

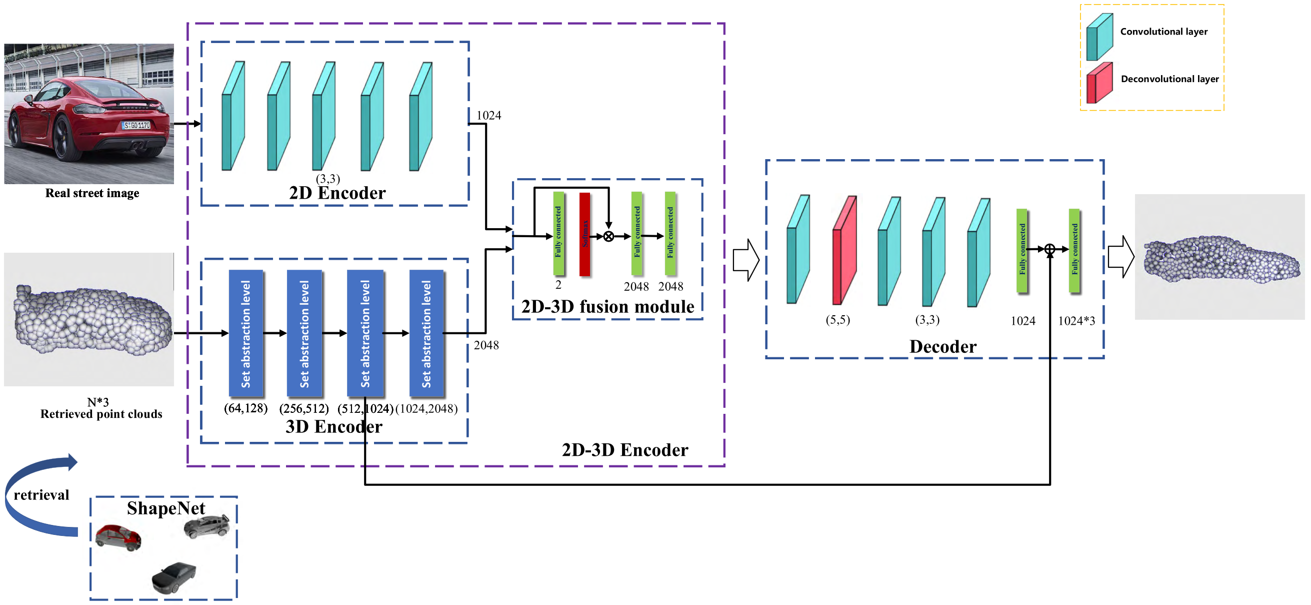

FIGURE 2. Network architecture.

representations, we reshape the flattened feature to the size of $16 \times 16 \times 8$, which is similar to the image size. The decoder is made of convolutional, deconvolutional and fully connected layers. Inspired by U-Net [7], the information of the encoding part is added to the decoder so that we can recover the details in the output. For example, the features of 2D RGB and edges are preserved in low layers, proving important for 3D reconstruction, also, the features from PointNet++ have an abstraction representation of 3D spatial distribution. The last layer is a fully connected layer; then, we reshape it to the size of $N \times 3$. We skip the connections between the last two layers in the encoding part and the deconvolutional layers in the decoding part.

The most different part from other single-view reconstruction works is the combination of 2D and 3D features, to have a clear comparison, we design a simplified version of the network, which is identical to the above structure without the 3D encoding part. Experiments have proven the effectiveness of the 3D part especially in real images. The comparative experiments of the two structures are given in Section V-C.

D. SPATIAL DISTANCE MEASUREMENT

To enable end-to-end training on RealPoint3D, a highly efficient and differentiable loss function must be designed. However, accurately measuring the topological similarity of a 3D shape is notably difficult. In addition, unlike the recent voxelized algorithm 3D-R2N2, in which a voxel including points returns 1, RealPoint3D outputs point clouds directly so that it cannot use the Intersection over Union (IoU) as the loss function. For comparing the similarities between two shapes, the Hausdorff distance metric is widely chosen to measure their discrepancies, but it is not robust to outliers because a single farthest distance can completely determine it. Inspired by PSGN, we explore the Chamfer distance (CD) as the distance function between $S_1, S_2 \subseteq R^3$:

$$d_{CD} = \sum_{p \in S_1} \min_{q \in S_2} \parallel p - q \parallel_2^2 + \sum_{p \in S_2} \min_{q \in S_1} \parallel p - q \parallel_2^2. \qquad (3)$$

CD is easily computable for two point sets under the effective implementation, which is equal to the mean over all nearest neighbor distances, and it induces a nice shape space geometrically. Since CD is more robust to outliers, it is a better choice as the loss function.

E. SPATIAL DISTRIBUTION MEASUREMENT

To guarantee similar spatial distribution of two shapes, a nature way is to observe them from arbitrary views and calculate the similarity, and it is obvious that two identical objects have the same projection. In this paper, we adopt the projection similarity to measure the consistence of spatial distribution between two shapes. Spatial points have their unique 2D locations on an plane, and they differ according to specific parameters, such as the camera intrinsic and external matrix. We can do the projection as in the 3D traditional computer vision: given a point x_i in a 3D space, the corresponding location p_i on a 2D plane is as follows:

$$p_i = R^{-1} \cdot (K^{-1} x_i - t) \qquad (4)$$

where R is the rotation matrix, t is the transform vector and K is the camera intrinsic matrix.

At each iteration, the generated point clouds and ground truth are rotated according to the same random transformation. Then, they are projected onto a 128 × 128 pixel image. For every pixel, the projection pixel and its surrounding three pixels are labeled as foreground. See Figure 6 and Table 1 as the results of projection.

Compared with voxel-based projection, the projection of point clouds delineates fine-grained parts. Also, there is a recent work [35], which generates the multi-view projection directly and is designed for dense point cloud generation. On the contrary, RealPoint3D focuses on real images and adopts the projection of the generated point clouds as an additional supervisor. The projection loss is the per-pixel discrepancy between the two projected images:

$$L_p = \sum_i \| p_i - q_i \|_2^2. \tag{5}$$

where p_i and q_i are pixels from two projection respectively. The total objective function in our model is:

$$L_{total} = d_{CD} + L_p. \tag{6}$$

F. IMPLEMENTATION DETAILS

We train the proposed network in TensorFlow and use Adam as the optimizer. To improve the performance, we choose the batch size of 32 and 200000 gradient steps. The learning rate automatically decays based on the number of iterations. The input image size is 128 × 128, and the last fully connected layer produces 1024 points. In the encoding part, the kernel size of convolutional layers is 3 × 3. In the decoding part, we set the kernel size of convolutional and deconvolutional layers as 5×5. In addition, the multi-scale grouping strategy is applied to the PointNet++ layers; 0.2 and 0.4 are used as the local regions of the ball radius. ReLU is used as all activation functions.

V. EXPERIMENT

We conducted several experiments to demonstrate the effectiveness of our method on real images and images rendered from ShapeNet, which is a large CAD dataset. First, we conducted the experiments on several types of objects, each with rendered images and sampling point clouds on the surface as described in Section V-A. Then, we trained and tested our network on five categories and compared with other state-of-the-art methods (Section V-B). Third, to demonstrate the function of the fusion with 3D features and the projection loss, we compared with a simplified version of the network (Section V-C). Finally, we verified that the purpose of generating objects with multiple viewpoints and complex backgrounds in real images was achieved (Section V-D).

A. DATASET

Composed of a huge number of objects, the ShapeNet dataset is an ongoing large-scale 3D model source that is widely used in 3D research fields, including 3D model retrieval and reconstruction. Our experiment is based on one of its subsets: ShapeNetCore55, which covers 55 common object categories with approximately 51,300 unique 3D models. As we know, it is notably difficult to generate 3D models from real images: if there is no real image dataset, there is no real image baseline. To solve the issue, we rendered CAD models with complex backgrounds to mimic the real world: each model has one fixed viewpoint for training and fixed and random viewpoints for testing, so we can more fairly evaluate the network's generalization to the real world. Simultaneously, to obtain point clouds, we sampled the surface as the ground truth. All point clouds are normalized. We split the dataset into training sets (4/5 of the entire dataset) and testing sets (remaining 1/5).

To evaluate the performance of our architecture on real-world problems, first, we tested our pre-trained model on ObjectNet3D, which is a large-scale dataset for 3D object recognition with 100 categories, 90127 images, 201888 objects in these images and 44147 3D shapes. For our task, we collected the images of the corresponding categories and models. In addition, we prepared some real images of four categories from cameras or the Internet: car, airplane, bench and chair.

B. SINGLE IMAGE RECONSTRUCTION COMPARISON

Two mainstream approaches are compared in the section: point set generation methods and volumetric reconstruction methods. We compared our method RealPoint3D with PSGN, which is reported as the state-of-the-art 3D object generation network based on the point set generation path. We evaluated five common categories of fixed viewpoint images: chair, bench, car, airplane and sofa. To make the measurement more reliable, we trained and tested our model with relatively sparse point clouds of 1024 points, which is sufficient to display the shape of an object. Then, to have a fairer comparison, we re-trained PSGN following its experiment setups with the dataset based on images with complex backgrounds. Then, RealPoint3D was compared with OGN, which is reported as the state-of-the-art volumetric method. OGN surpassed traditional volumetric methods relying on the changing from the original organized voxel grids to the compact octree structure, which significantly enhanced the computational efficiency and reduced the storage. Five categories were trained and tested using the pairs of a image and a retrieved point cloud.

As shown in Table. 1, we quantitatively analyzed the CD scores for each of the five categories in the testing set. Our approach outperformed PSGN in every category, particularly the bench, but the result was close for cars. The possible reason is that the retrieval accuracy of the bench is high, so the similar point clouds can teach to generate more accurate 3D models. Meanwhile, the retrieval accuracy of cars is low mainly because there are various car shapes, so the retrieval is difficult; a relatively wrong retrieval shape may even mislead the generation. Nonetheless, our generation is more accurate. Even though the projection is ambiguous, additional 3D information that is not very similar still outlines the general shape of the particular category. These five categories have higher shape variations and better details than other classes, which strongly indicates that our approach performs better, particularly for objects with fine details. In addition, RealPoint3D has insufficient advantages in the car and sofa categories possibly because these two categories are rotund shapes and have fewer shape details, so PSGN

TABLE 1. CD scores for different methods on complex background images. 'Retrieval' is the loss with the nearest shape. We achieve lower CD in all categories (a smaller number represents better performance).

Category	PSGN	Retrieval	simplified RealPoint3D	RealPoint3D	RealPoint3D-projection
Sofa	0.00220	0.00683	0.00246	0.00195	**0.00182**
Airplane	0.00100	0.00367	0.00138	0.00079	**0.00077**
Bench	0.00251	0.00211	0.00355	**0.00211**	0.00218
Car	0.00128	0.00196	0.00131	0.00126	**0.00125**
Chair	0.00238	0.00691	0.00253	0.00213	**0.00197**

TABLE 2. IoU scores for different methods on complex background images. 'Retrieval' is the loss with the nearest shape.

Category	OGN	Retrieval	RealPoint3D
Sofa	0.11204	0.11691	**0.22132**
Airplane	0.14727	0.36122	**0.53280**
Bench	0.04608	0.16942	**0.35925**
Car	**0.44141**	0.23483	0.33544
Chair	0.13935	0.12643	**0.26897**

TABLE 3. IoU scores for different methods on pure background images. PSGN and OGN have no reported sofa IoU scores.

Category	PSGN	OGN	RealPoint3D
Sofa	-	-	**0.627**
Airplane	0.601	0.587	**0.667**
Bench	0.550	0.481	**0.578**
Car	**0.831**	0.816	0.775
Chair	0.544	0.483	**0.577**

still performs well in these cases. Among them, RealPoint3D with projection loss achieved best performance, proving the promotion with the projection loss. The superiority is obvious on chair due to its various outlines can have more supervision on the generation model.

Considering the large difference between OGN and PSGN, the IoU (intersection over union) was used for OGN evaluation, and the CD distance was used for PSGN, in line with the original papers. Considering the complex background setting, the two networks were re-trained. The results with OGN are shown in Table. 2, measured by the IoU.

For a fairer comparison, RealPoint3D was compared with PSGN and OGN on pure background images as in their initial works, and the results are shown in Table 2. We used the IoU as the evaluation criterion, to be more convincing, the IoU of PSGN and OGN was selected from the previous works. The same five categories were tested, but the IoU values of sofa for PSGN and OGN are missing because there is no sofa in the original reports. RealPoint3D has the highest IoU scores except for cars. The effective generative capability was proven in the other four categories, i.e., RealPoint3D has higher accuracy than OGN and PSGN. Tables 2 and 3 show that the same category has different IoU scores for complex and pure backgrounds, and the former is commonly much lower than the latter. Sofas and chairs with complex backgrounds have the lowest IoU scores, whereas airplanes have the highest scores. Backgrounds have strong effects on the generation, particularly for objects with relatively complicated structures such as sofas and chair; in this situation, RealPoint3D can demonstrate its strong power because of the instruction of the nearest 3D model. In contrast, airplanes and cars have similar shapes but no elaborate structures, so their IoUs slightly differ.

To clearly sketch the details, we show a visual comparison with PSGN in Figure 3. PSGN loses the fine details such as the armrest and wings of the airplane, whereas RealPoint3D provides more fine-grained details. In general, even when the image has a notably complex background and the object is ambiguous, RealPoint3D is robust and can generate a good point cloud. This is attributed to the 3D shape retrieval, which provides additional spatial information as the input. The 3D encoder branch offers the class information of the object, and the image features finally indicate that the output more closely matches the real shape. In Section V-C, we prove the importance of PointNet++ layers by comparing the differences of the two networks.

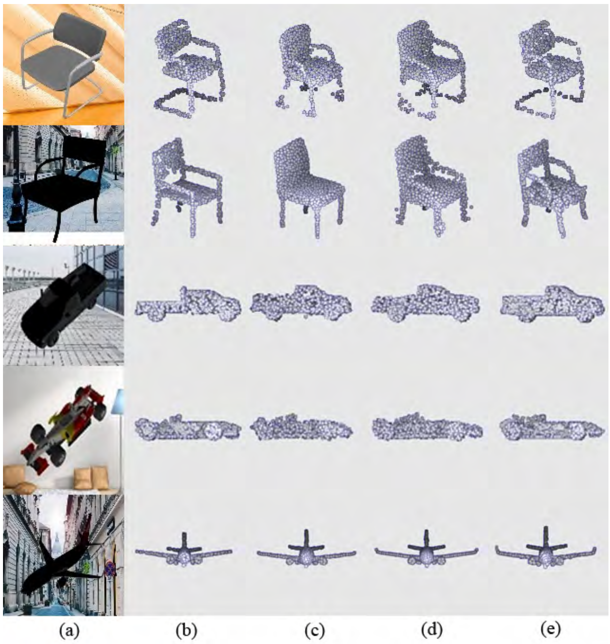

FIGURE 3. Comparison with the state-of-the-art work. (a) Input image with a complex background; (b) retrieval shape; (c) PSGN output; (d) RealPoint3D output; (e) ground truth shape.

C. NETWORK STRUCTURE COMPARISON

We aim to design a network that can generate point clouds of objects while remaining robust to real images with multiple

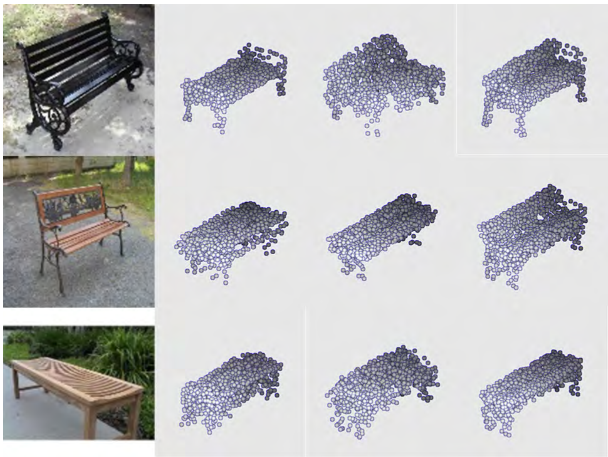

FIGURE 4. Each method occupies one column, and the first column is the 2D image; the three methods are PSGN, simplified RealPoint3D and RealPoint3D.

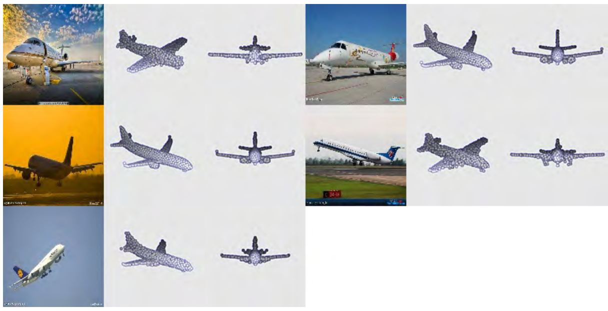

FIGURE 5. Results of RealPoint3D on real images from two viewpoints.

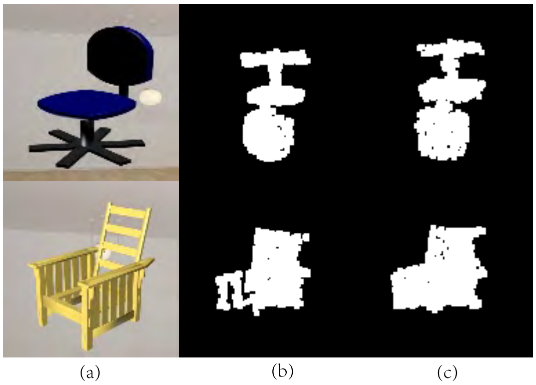

FIGURE 6. Two samples of projection. (a) Rendered images. (b) Projection of ground truth. (c) Projection of generated shapes.

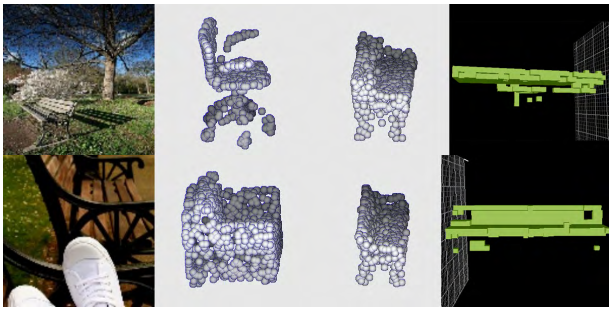

FIGURE 7. Bench images from ObjectNet3D. The orders from left to right: Retrieval, RealPoint3D and OGN.

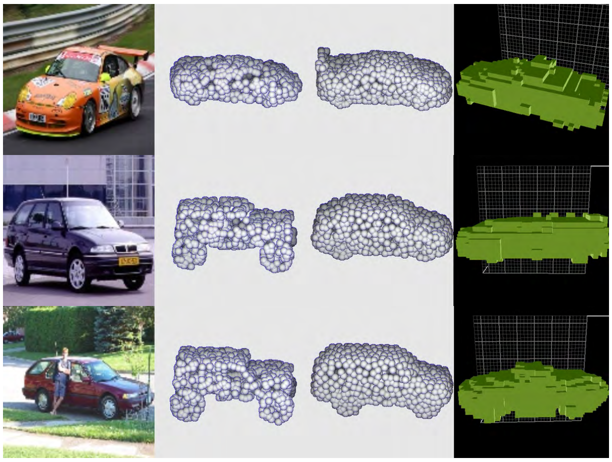

FIGURE 8. Car images from ObjectNet3D. The orders from left to right: Retrieval, RealPoint3D and OGN.

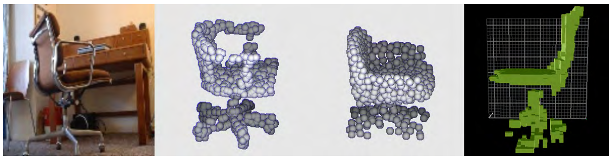

FIGURE 9. Car images from ObjectNet3D. The orders from left to right: Retrieval, RealPoint3D and OGN.

views and complex backgrounds. In this section, we will verify the effective function of the fusion with 3D features and the projection loss.

We perform a comparative experiment on the effectiveness of 3D features, encoded by PointNet++, as shown in Table 1. The branch of PointNet++, the 2D-3D feature fusion module and the deconvolutional layer are cut off to fit the dimension, and the result is illustrated in Figures 4, 10, 11 and 12. Unsurprisingly, the simplified RealPoint3D structure performs worse because it lacks the 3D information, whereas RealPoint3D uses the 3D encoder for the generation.

The function of projection is to delineate the outline of an object. Compared with volumetric methods [36], which cannot delineate some fine-grained parts, our method can make that delineation, thus promoting training quality, seen in Table. 1. Shown in Figure 6 are two samples of the projection results with some fine-grained parts.

We also performed many experiments on the two structures on rendered and real images with different viewpoints, displayed in Figure 13. The simplified RealPoint3D network misses the details and even generates some incorrect small parts of the objects, while RealPoint3D preserves most of them.

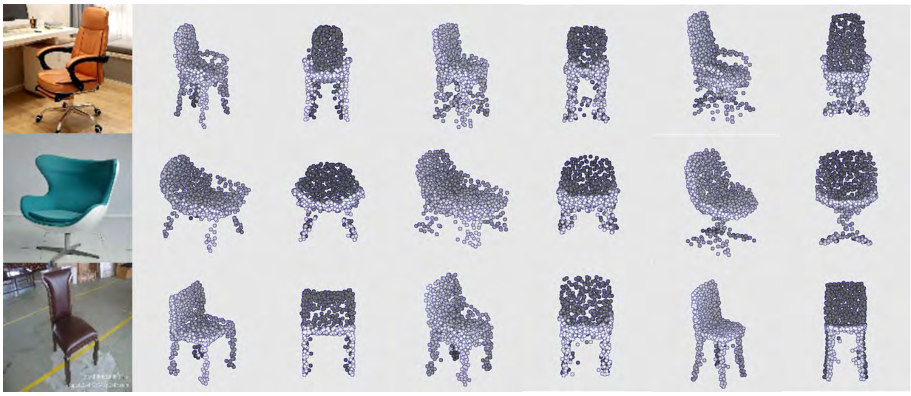

FIGURE 10. Each method occupies two columns, and the first column is the 2D image. The three methods are PSGN, simplified RealPoint3D and RealPoint3D.

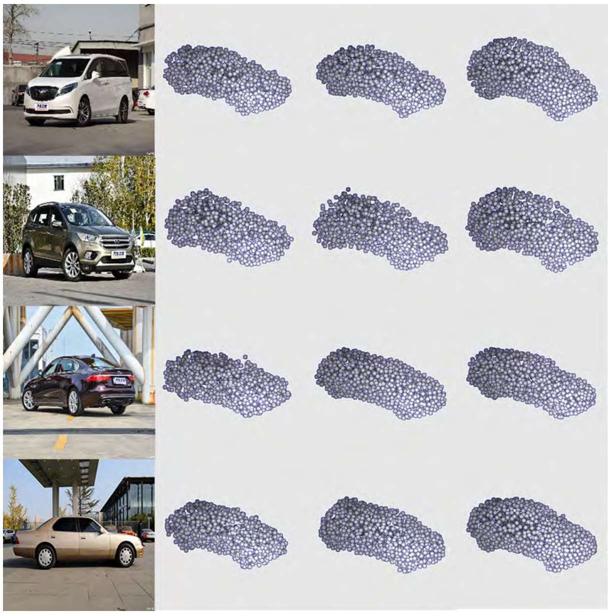

FIGURE 11. Category of cars. Each method occupies one column, the first column is the 2D image, and the three methods are PSGN, simplified RealPoint3D and RealPoint3D.

D. REAL IMAGE RECONSTRUCTION BASED ON REAL IMAGES

Real images from ObjectNet3D and the Internet or cameras are thoroughly tested, and the results are shown below. Because of the lack of corresponding models, the rendered images of 3D models are displayed.

The main drawback of previous generative works is their limited ability to reconstruct from real images with diverse viewpoints and complex backgrounds. To explain the strong competence of our method in addressing this issue, we tested three types of objects in comparison with OGN (Figures 7, 8). As observed, because of the derivation from point cloud generation methods, RealPoint3D demonstrates strong fine-grained generative capability. For example, in the construction of benches in Figure 7, RealPoint3D can recover the legs, but OGN severely misses them. For cars, the wheels are almost flattened and only the coarse shapes remain in OGN, but RealPoint3D has better representation than OGN even for slight flows on the car surfaces. There is a situation with wrong retrieval in Figure 7, but the intact model is still recovered. This result indicates that the images and nearest models act interactively, finally achieving generation of the exact models.

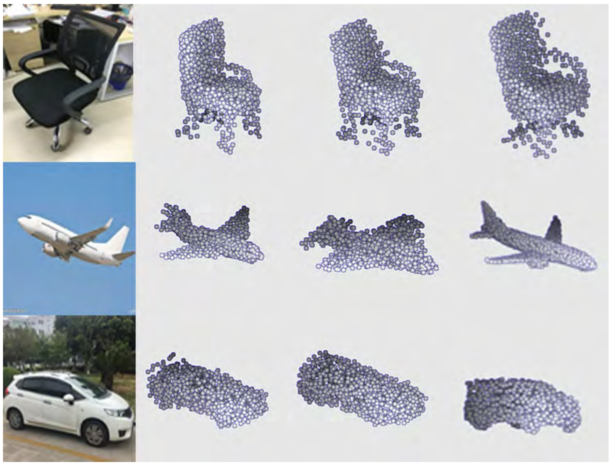

FIGURE 12. Each method occupies one column, the first column is the 2D image, and the three methods are PSGN, simplified RealPoint3D and RealPoint3D.

First, we made comparisons on cars. Our approach has similar performance on simple structures as PSGN (Figures 11 and 12). Cars are approximately cuboids or cylinders. As shown in Figure 11, the result from PSGN is not as smooth as ours, particularly at the top and bottom, with some points flying out from the structures. However, even in different categories of cars (SUV, MVP, limousine, etc.) with different heights and widths, RealPoint3D remains more fine-grained, smooth and solid as usual.

The chair is the most difficult category to reconstruct because of the significantly more fine-grained details (Figures 10, 12 and 13). For swivel chairs, PSGN shows a

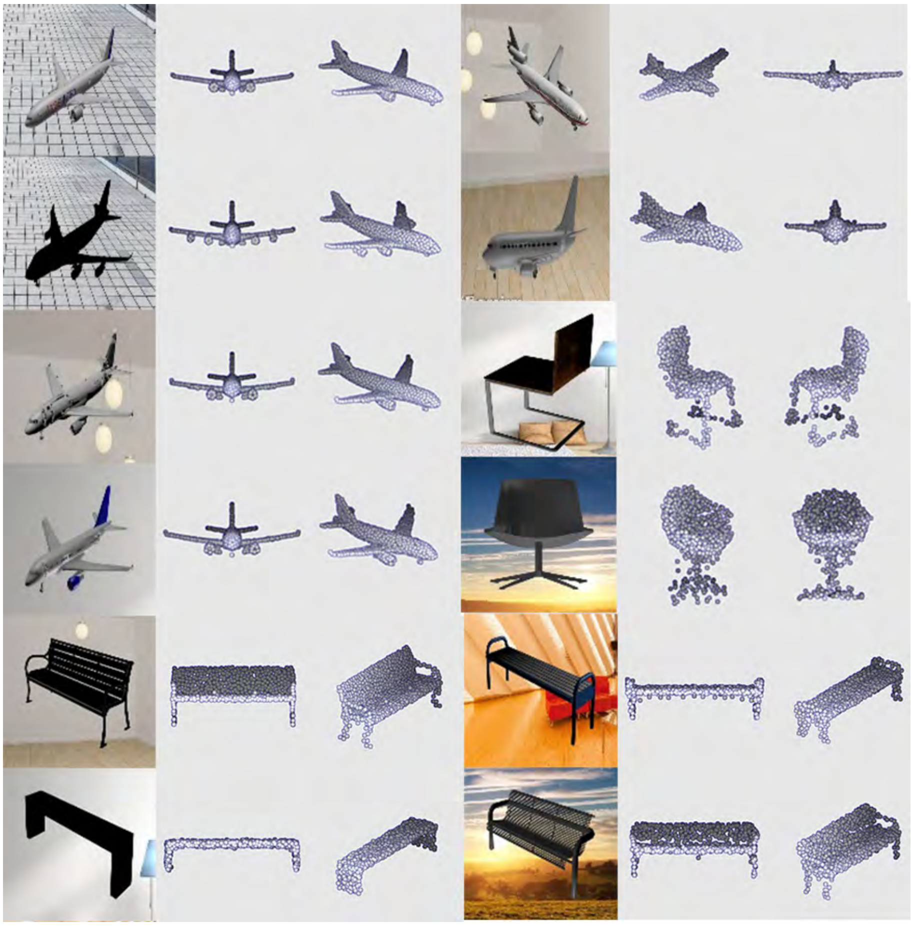

FIGURE 13. Results of RealPoint3D on rendering images from two viewpoints.

large cluster of unordered points on the legs, whereas the reconstruction works well for RealPoint3D. In the middle row, PSGN mistakenly generates four legs, and RealPoint3D forms one leg with four feet, which is consistent with the image. For airplanes (Figures 5, 12 and 13), recognizing the engines, small structures, components on the wings and tails is notably difficult. These parts are missing in the PSGN results. However, these details are well preserved by our network, which clearly shows these indistinguishable parts. In the generation of benches (Figures 4 and 13), our model can maintain a complete reconstruction of backs, legs, and even small connections of different parts, most of which are missing in PSGN results. Figure 4 compares several methods for the bench category. We can easily find that our method still has a good performance for the fine details. The PSGN method is not smooth and has many discrete points. In addition, PSGN lost the armrest of the bench. The simplified Realpoint3D method can only obtain an approximate outline and misses details. Then, we compared RealPoint3D to the RealPoint3D simplified structure on four categories. The simplified RealPoint3D structure performs worse than RealPoint3D because of the lack of 3D features, and it is also worse than PSGN. Because of the fusion of 3D features in the RealPoint3D structure, the network works better than PSGN in fine-detail generation.

E. TIME COMPLEXITY

In our current implementation, we set 500 epochs in the training stage. We trained the data on 5 P100 GPUs in parallel, which cost approximately 10 hours. In the test stage, the processing of 100 images consumes approximately 10 s on a laptop with GPUs. The time efficiency is similar to that of PSGN but is significantly more efficient than OGN. Comparing the testing time, OGN took approximately 160 s per 100 outputs.

VI. CONCLUSION

We designed a novel generation network that is more suitable for 3D fine-grained reconstruction from a single image. The most different part with previous generative methods is the adaptive fusion of nearest 3D features and the projection loss, which can prompt the results to obtain more details with complex backgrounds and multiple viewpoints. We achieved state-of-the-art performance in reconstruction from pure background images and real images in comparison with other generative methods.

REFERENCES

[1] A. X. Chang *et al.*, ''Shapenet: An information-rich 3d model repository,'' *Computer Science*.

[2] Y. Xiang *et al.*, ''Objectnet3D: A large scale database for 3D object recognition,'' in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 160–176.

[3] J. Wu, C. Zhang, T. Xue, B. Freeman, J. Tenenbaum, ''Learning a probabilistic latent space of object shapes via 3D generative-adversarial modeling,'' in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 82–90.

[4] C. B. Choy, D. Xu, J. Gwak, K. Chen, 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.

[5] M. Tatarchenko, A. Dosovitskiy, T. Brox. (2017). ''Octree generating networks: Efficient convolutional architectures for high-resolution 3D outputs.'' [Online]. Available: https://arxiv.org/abs/1703.09438

[6] D. Maturana, S. Scherer, ''Voxnet: A 3D convolutional neural network for real-time object recognition,'' in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep./Oct. 2015, pp. 922–928.

[7] Ö. Çiçek, A. Abdulkadir, S. S. Lienkamp, T. Brox, and O. Ronneberger, ''3D U-net: Learning dense volumetric segmentation from sparse annotation,'' in *Medical Image Computing and Computer-Assisted Intervention*, 2016, pp. 424–432.

[8] H. Chen, Q. Dou, L. Yu, P.-A. Heng. (2016). ''Voxresnet: Deep voxelwise residual networks for volumetric brain segmentation.'' [Online]. Available: https://arxiv.org/abs/1608.05895

[9] D. J. R. Meagher, ''Octree encoding: A new technique for the representation, manipulation and display of arbitrary 3-D objects by computer,'' in *Electrical and Systems Engineering Department Rensseiaer Polytechnic Institute Image Processing Laboratory*, 1980.

[10] C. R. Qi, H. Su, K. Mo, L. J. Guibas, ''Pointnet: Deep learning on point sets for 3D classification and segmentation,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 652–660.

[11] C. R. Qi, L. Yi, H. Su, L. J. Guibas, ''Pointnet++: Deep hierarchical feature learning on point sets in a metric space,'' in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5099–5108.

[12] H. Fan, H. Su, L. J. 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.

[13] Q. Huang, H. Wang, V. Koltun, ''Single-view reconstruction via joint analysis of image and shape collections,'' *ACM Trans. Graph.*, vol. 34, no. 4, p. 87, 2015.

[14] J. Fuentes-Pacheco, J. Ruiz-Ascencio, and J. M. Rendón-Mancha *Visual Simultaneous Localization and Mapping: A Survey*. Norwell, MA, USA: Kluwer Academic Publishers, 2015.

[15] K. Häming and G. Peters, ''The structure-from-motion reconstruction pipeline—A survey with focus on short image sequences,'' *Kybernetika*, vol. 46, no. 5, pp. 926–937, 2010.

[16] J. T. Barron, J. Malik, ''Shape, illumination, and reflectance from shading,'' *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 8, pp. 1670–1687, Aug. 2015.

[17] G. Healey, T. O. Binford, ''Local shape from specularity,'' *Comput. Vis., Graph., Image Process.*, vol. 42, no. 1, pp. 62–86, Apr. 1988.

[18] J. Malik, R. Rosenholtz, ''Computing local surface orientation and shape from texture for curved surfaces,'' *Int. J. Comput. Vis.*, vol. 23, no. 2, pp. 149–168, Jun. 1997.

[19] S. Savarese, M. Andreetto, H. Rushmeier, F. Bernardini, P. Perona, ''3D reconstruction by shadow carving: Theory and practical evaluation,'' *Int. J. Comput. Vis.*, vol. 71, no. 3, pp. 305–336, Mar. 2007.

[20] R. Zhang, P.-S. Tsai, J. E. Cryer, and M. Shah, ''Shape-from-shading: A survey,'' *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 8, pp. 690–706, Aug. 1999.

[21] R. Girdhar, D. F. Fouhey, M. Rodriguez, A. Gupta, ''Learning a predictable and generative vector representation for objects,'' in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 484–499.

[22] X. Yan, J. Yang, E. Yumer, Y. Guo, H. Lee, ''Perspective transformer nets: Learning single-view 3D object reconstruction without 3D supervision,'' in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 1696–1704.

[23] 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. Adv. Neural Inf. Process. Syst.*, 2016, pp. 4997–5005.

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

[25] H. Su, S. Maji, E. Kalogerakis, E. Learned-Miller, ''Multi-view convolutional neural networks for 3D shape recognition,'' in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 945–953.

[26] C. R. Qi, H. Su, M. Niessner, A. Dai, M. Yan, L. J. Guibas, ''Volumetric and multi-view CNNs for object classification on 3D data,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 5648–5656.

[27] J. Bruna, W. Zaremba, A. Szlam, Y. Lecun, ''Spectral networks and locally connected networks on graphs,'' in *Proc. Int. Conf. Learn. Represent.*.

[28] O. Vinyals, S. Bengio, M. Kudlur, ''Order matters: Sequence to sequence for sets,'' in *Proc. Int. Conf. Learn. Represent.*

[29] M. M. Bronstein, J. Bruna, Y. Lecun, A. Szlam, P. Vandergheynst, ''Geometric deep learning: Going beyond Euclidean data,'' in *Proc. IEEE Signal Process. Mag.*, vol. 34, no. 4, pp. 18–42, Jul. 2016.

[30] L. Yi, H. Su, X. Guo, L. J. Guibas, ''Syncspeccnn: Synchronized spectral CNN for 3D shape segmentation,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2282–2290.

[31] J. Masci, D. Boscaini, M. M. Bronstein, P. Vandergheynst, ''Geodesic convolutional neural networks on riemannian manifolds,'' in *Proc. IEEE Int. Conf. Comput. Vis. Workshop*, Dec. 2015, pp. 37–45.

[32] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, P. Vandergheynst, ''The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains,'' *IEEE Signal Process. Mag.*, vol. 30, no. 3, pp. 83–98, May 2012.

[33] M. Defferrard, X. Bresson, P. Vandergheynst, ''Convolutional neural networks on graphs with fast localized spectral filtering,'' in *Proc. 30th Int. Conf. Neural Inf. Process. Syst.*, Dec. 2016, pp. 3844–3852.

[34] T. N. Kipf, M. Welling, ''Semi-supervised classification with graph convolutional networks,'' Tech. Rep.

[35] C.-H. Lin, C. Kong, S. Lucey, ''Learning efficient point cloud generation for dense 3d object reconstruction,'' in *Proc. AAAI Conf. Artif. Intell. (AAAI)*, 2018, pp. 1–8.

[36] M. Tatarchenko, A. Dosovitskiy, T. Brox, ''*Multi-view 3D Models from Single Images with a Convolutional Network*,'' Springer, 2016.

YANG ZHANG received the M.S. degree in information and communications from the National University of Defense Technology (NUDT), Changsha, China, in 2016, where he is pursuing the Ph.D. degree in information and communication engineering with the College of Electronic Science. His current research interests include remote sensing image, point clouds processing, and 3D reconstruction.

ZHEN LIU received the B.S. degree from Zhejiang University, Hangzhou, China, in 2006, and the Ph.D. degree from the National University of Defense Technology (NUDT), Changsha, China, in 2013, where he is currently a Lecturer with the College of Electronic Science. His research interests include radar signal design, radar imaging, radar countermeasures, and compressed sensing theory.

TIANPENG LIU received the B.Eng. and M.Eng. degrees from the National University of Defense Technology (NUDT), Changsha, China, in 2008 and 2011, respectively, where he is pursuing the Ph.D. degree. His primary research interests include radar signal processing, radar imaging, and cross-eye jamming.

BO PENG received the B.S., M.S.E.E., and Ph.D. degrees from the National University of Defense Technology (NUDT), Changsha, China, in 2008 and 2010, respectively, where he is currently a Lecturer with the School of Electronic Science and Engineering. His research interests include nonstationary signal processing, micro Doppler analysis, detection, estimation and imaging of nonstatic targets for radar systems, and radar target recognition.

XIANG LI received the B.S. degree from Xidian University, Xi'an, China, in 1989, and the Ph.D. degree from the National University of Defense Technology (NUDT), Changsha, China, in 1998, where he is currently a Professor with the School of Electronic Science and Engineering. His current research interests include radar signal processing, radar imaging, and automation target recognition.