IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received May 31, 2019, accepted June 16, 2019, date of publication June 19, 2019, date of current version July 9, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2923842

SSL-Net: Point-Cloud Generation Network With Self-Supervised Learning

RAN SUN, YONGBIN GAO, ZHIJUN FANG, (Senior Member, IEEE), ANJIE WANG, AND CENGSI ZHONG

School of Electronic and Electrical Engineering, Shanghai University of Engineering Science, Shanghai 201620, China

Corresponding authors: Yongbin Gao (gaoyongbin@sues.edu.cn) and Zhijun Fang (zjfang@sues.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61831018, Grant 61802253, Grant 61772328, in part by the Collaborative Innovation Center for Economic Crime Investigation and Prevention Technology of Jiangxi Province under Grant JXJZXTCX-027, and in part by the Chenguang Talented Program of Shanghai under Grant 17CG59.

ABSTRACT Inferring the three-dimensional structure of objects from monocular images has far-reaching applications in the field of 3D perception. In this paper, we propose a self-supervised network (SSL-Net) to generate 3D point clouds from a single RGB image, unlike the existing work which requires multiple views of the same object to recover the full 3D geometry. To provide the extra self-supervisory signal, the generated 3D model is simultaneously rendered into an image and compared with the input image. In addition, a pose estimation network is integrated into the 3D point cloud generation network to eliminate the pose ambiguity of the input image, and the estimated pose is also used for rendering the 2D image with the same pose as input image from 3D point clouds. The extensive experiments on both real and synthetic datasets show that our method not only qualitatively generates point clouds with more details but also quantitatively outperforms the state-of-the-art in accuracy.

INDEX TERMS 3D reconstruction, point-cloud, self-supervised learning, 3D shape completion, single view reconstruction.

I. INTRODUCTION

3D shape perception is a fundamental theme both in human and computer vision. The ability to infer 3D structure from monocular images has far-reaching applications in the field of robotics and perception, such as AR/VR [1], scene segmentation [2], object deformation, robotic grabbing and obstacle avoidance [3], [4]. However, the task faces considerable challenges: firstly, an image does not have a one-one correspondence with its 3D structure, leading to inherent ambiguity. Secondly, information is limited in 2D inputs due to lack of efficient image mining methods.

Although human can infer the three-dimensional structure of a scene and the shapes of objects from limited information due to a strong prior knowledge about shapes and geometries of objects or scenes. It is extremely challenging for computers to reconstruct an 3D object or a scene from one or multiple viewpoints. It is an inherently ill-posed problem where the variety of factors, such as shape, color, texture and illumination may lead to a correspondence between the model and multiple different 2D images. To tackle the problem, most existing methods obtain information from an image sequence [5] or multi-view stereo (MVS) [39]. However, in many cases, such as real-time 3D reconstruction and the environment where multiple views of the object cannot be obtained, these multi-view reconstruction will no longer be suitable and unable to meet actual needs. Hence, 3D reconstruction from single-view image is an emerging research since it is more suitable for different scenes, more convenient for retrieving data and more economical and practical. In recent years, convolutional neural network (CNN) has made remarkable progress in the field of 3D vision [7]–[9]. Especially, several large open source 3D model repositories have generated, such as ShapeNet [10] and Pix3D [11], which all contribute to the further study of single-view reconstruction tasks.

A 3D model can be represented in various forms, such as voxel or point cloud. Recent researches on 3D reconstruction based on deep learning can directly map the input image to a geometric model [12]–[14]. 3D CNN is utilized on the voxel grid to reconstruct 3D structures from a single image, in which volumetric representation is typically

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

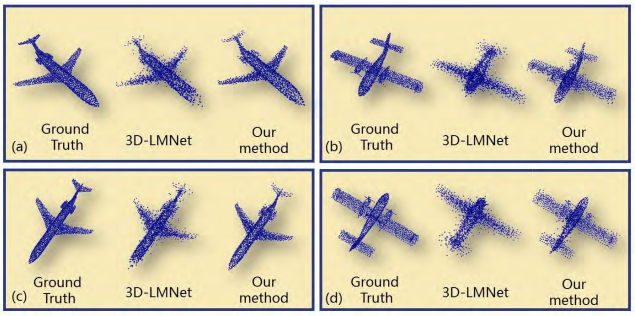

FIGURE 1. Diagram of difficulties in single-view point cloud object reconstruction. On one side, recovering 3D structure from a single-view image is an ill-posed problem. The result of optimizing the model only with the supervision of ground-truth 3D data [19] may lack detailed surfaces. On the other side, it is also found that the visual error of generated point clouds vary with different poses. For instance, (a) and (c) represent the same object from two different perspectives but they look quite different, (b) and (d) are the same. The reconstructed point clouds from 3D-LMNet miss some important detailed surfaces, while our method can solve these difficulties effectively and receive detailed 3D shapes.

required [15]–[17]. However, unlike image generation, in which each pixel is equipped with specific spatial and texture information, voxel representation is inherently more difficult due to the rough shape surface, complicated calculation, and excessive computational consumption. In contrast, 3D point clouds effectively exploit the sparseness of the data and can represent the surface of the shape with more details. Recent work in this field has focused on designing neural network architectures and loss functions to process and predict 3D point clouds more accurately [18]–[20]. Point clouds are prestigious for their scalable data representations, compact encoding of shape information, and optionally embedding textures.

However, there are still unsolved problems in the task of single-view image reconstruction with point clouds. On one side, recovering 3D structure from a single-view image is an ill-posed problem. The result of optimizing the model only with the supervision of ground-truth 3D data [19] is shown as in FIGURE 1, where the reconstructed point clouds lack detailed surfaces. It can be seen that it is not sufficient to induce the network to generate a reasonable and accurate 3D model relying solely on the supervision of three-dimensional information. On the other side, it is also found that the visual error of generated point clouds vary with different poses.

In this paper, we propose a cascaded network based on self-supervised learning which reconstructs 3D point clouds from a single RGB image. Compared with the traditional structure from motion (SFM) method [5] and MVS method [6] for 3D reconstruction, both of which require a mass of images to cover each viewpoint of the object, the proposed network can infer the unseen part of the object from a single image based on the semantic learning, resulting in an intact dense reconstruction of 3D objects. In order to solve the problem of inadequate supervision, we introduce an extra self-supervisory signal, the generated 3D model is simultaneously rendered into an image and compared with the input image. Meanwhile, a pose estimation network is integrated into the 3D point cloud generation network to eliminate the pose ambiguity of the input image, and the estimated pose is also used for rendering the 2D image with the same pose as input image from 3D point clouds. In addition, the method proposed in this paper can also be used as a basic method to generate the preliminary point clouds of the object, and then combined with the existing methods to obtain a more detailed three-dimensional model. The key contributions of our work are summarized as follows:

- We propose a novel self-supervised learning (SSL-Net) pipeline for generating 3D point clouds from a single RGB image, in which the generated 3D point cloud is transformed into a 2D image to compare with the input RGB image. The difference of these two images is used as extra 2D loss for 3D point cloud generation network, which compensates for the 3D loss that lacks of structure information.
- An image pose estimation network is integrated into the 3D point cloud generation network, the predicted pose of the input image can eliminate the impact of pose variations of input image, resulting in a pose-aware 3D reconstruction.

II. RELATED WORK

A. 3D OBJECT RECONSTRUCTION

Most works on 3D reconstruction focus on the multi-view geometry (MVG) [21], such as structure from motion (SfM) [5], [22], [23], simultaneous localization and mapping (SLAM) [6], and depth sensing devices (e.g RealSense cameras) based methods [24]. Although these methods contribute a lot in specific scenes, there are some drawbacks, such as 1) MVG can't rebuild missing parts of the view, therefore enough images from every viewpoint are required to ensure the integrity of reconstruction. 2) Reconstruction of multi-views means an increase in computational complexity, which is not applicable for real-time reconstruction. These drawbacks limit the application of multi-view reconstruction. Therefore, the learning-based approaches for single-view reconstruction are extensively investigated recently.

The learning-based approach requires sufficient training data to learn semantic features. However, there is no large open 3D database in the field that meets the requirements in the early stage. Recently, the emergence of some large 3D model libraries (such as ModelNet [25], ShapeNet [10] and Pixel3D [11]) promotes the research progress of 3D object reconstruction. Meanwhile, a number of effective networks for 3D data analysis have been proposed, such as the network FPNN [26] for voxel shapes, as well as Pointnet [27] and PointCNN [7] for the feature extraction of point clouds.

B. DEEP LEARNING ON SINGLE IMAGE GENERATION

Prior works including 3D auto-encoder [28] and recurrent network [14] learn a latent representation for volumetric data. With the significant progress in 3D deep learning field, most

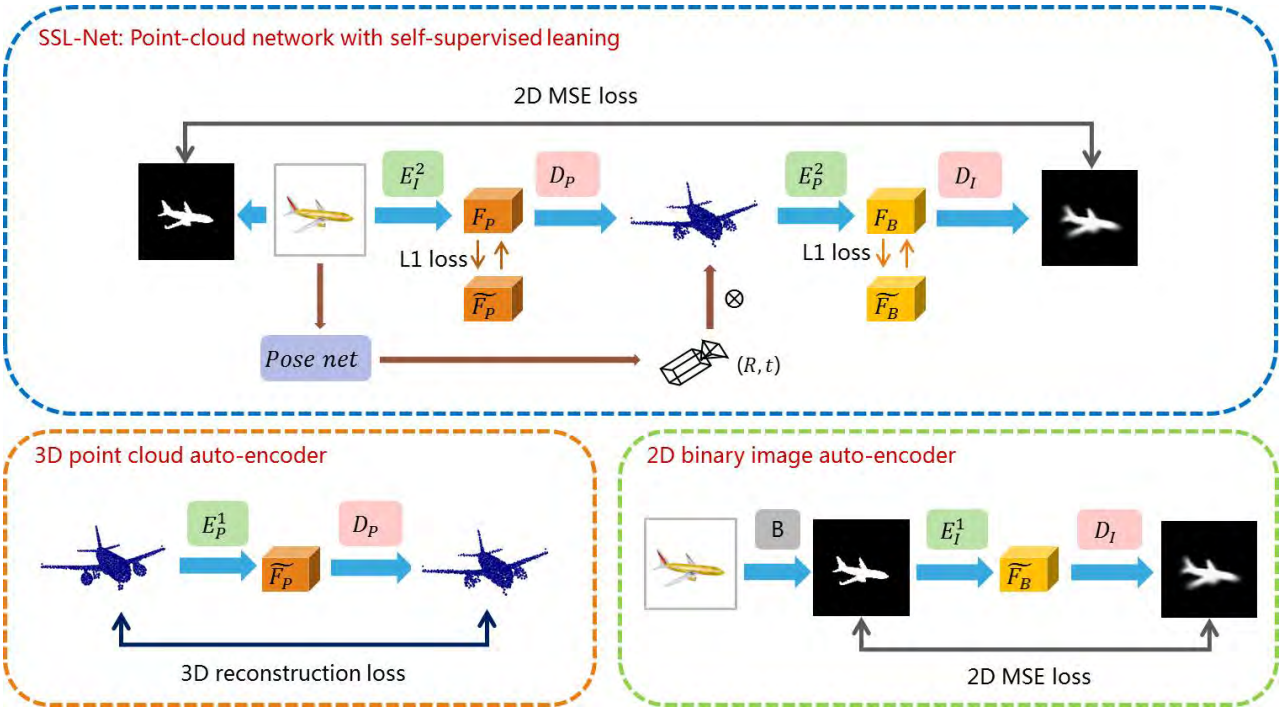

FIGURE 2. SSL-Net pipeline. We propose a 3D point cloud generation network based on self-supervised learning (SSL-Net). The auto-encoder network for pre-train is applied to both point clouds and images to obtain the latent features, which are used to supervise the SSL-Net. We introduce an extra self-supervisory signal, that is, the generated 3D model is simultaneously rendered into an image and compared with the input image. In addition, a pose estimation network is integrated into the 3D point cloud generation network to eliminate the pose ambiguity of the input image, and the estimated pose is also used for rendering the 2D image with the same pose as input image from 3D point clouds..

of the learning-based single-view reconstruction is generated by 3D CNN similar to the method of 2D CNN image generation, and represented by voxel shape [17], [29]. Stutz and Geiger [30] proposed a weak supervisory mechanism fitting 3D shapes completion through a variational auto-encoder (VAE). Hane *et al.* [16] reconstructed a single RGB image with a high-precision voxel grid. Yang *et al.* [15] proposed a novel network based on generative adversarial networks (GAN) recovering the 3D structure directly from a single depth view, which is referred to 3D-RecGAN. However, the voxel grid resolution obtained with these two methods is 256^3, which is extremely demanding on hardware devices. Recently, Tatarchenko *et al.* [31] proposed an octree representation, which enables higher resolution outputs in 3D reconstruction with limited memory. However, 3D voxel is not a mainstream 3D representation in game and movie industries. 3D shape represented by voxel increases the surface accuracy by adding three-dimensional blocks, and abundant information is located on the surface of the 3D object, which expends a lot of computing resources.

In order to balance computational complexity and surface accuracy, mesh [32], [34] and point cloud based methods have recently been proposed. Wangle *et al.* [32] proposed a coarse-to-fine mechanism to directly generate a triangular mesh of a color image based on Graph CNN. Jack *et al.* [33] proposed a free-form deformations method for learning 3D reconstruction from a single image. Haoqiang *et al.* [18] proposed a network (PSG-Net) and loss functions to generate scattered point clouds, and presented a single-view 3D reconstruction approach superior to the voxel representation method [14]. Mandikal *et al.* [19] emphasized on the importance of learning the latent representation of 3D point clouds before mapping the image into 3D space, which improved the reconstruction accuracy of point clouds. Lin *et al.* [20] optimized the network by synthesizing new depth maps for the input images to obtain denser point clouds.

III. APPROACH

In this paper, we propose a self-supervised learning pipeline for 3D point cloud reconstruction from a single RGB image as shown in FIGURE 2. It consists of a point cloud auto-encoder, a binary image auto-encoder and a network generating 3D point clouds from the input RGB images. In order to represent all the characteristics of the input data with fewer parameters, we introduce four latent features ($\widetilde{F_P}$, F_P, $\widetilde{F_B}$ and F_B) and design them into the same dimensions (512-dimensional features) for ease of supervision and the subsequent decoding process. The 3D models are composed of numerous unordered points and RGB images are composed of Tens of thousands of pixels, both of which are represented by a large amount of parameters and bring much trouble to fast feature extraction and reconstruction. According to recent research, we found that the auto-encoder is very suitable for handling the above problem for itself is a common method

of dimensionality reduction. We use the appropriate feature extraction network as the encoder for different input data. The details of the extracted features are as follows:

1) $\widetilde{F_P}$ is the latent extraction feature of the 3D point cloud auto-encoder. As shown in the orange dotted block diagram of FIGURE 2, we use a network structure similar to point-net [27] as the encoder E_P^1 to extracting features. $\widetilde{F_P}$ contains the structure and direction information of the point clouds, which can be used to describe the 3D shape. In addition, $\widetilde{F_P}$ will serve as a supervisor for the next point cloud generation network.

2) F_P is actually the latent feature of the input RGB images using two-dimensional convolution in the point cloud generation network as shown in the blue dotted block diagram of FIGURE 2. To generate the preliminary point clouds, we take the trained D_P obtained by the point cloud auto-encoder as the decoder of F_P. As long as F_P and $\widetilde{F_P}$ are close enough, we treat the output of D_P as the generated preliminary point clouds.

3) $\widetilde{F_B}$ is the latent extraction feature of the binary image auto-encoder. As shown in the green dotted block diagram of FIGURE 2, the input binary image is encoded by E_I^1, and then, the obtained feature $\widetilde{F_B}$ is applied to represent the deep information of the binary image. In addition, $\widetilde{F_B}$ will serve as a supervisor for the next point cloud generation network to restore the binary images.

4) F_B is the latent feature of the generated preliminary point clouds obtained from the decoder D_P as shown in the blue dotted block diagram of FIGURE 2. To restore the binary images of the preliminary point clouds, we take the trained D_I obtained by the binary image auto-encoder as the decoder of F_B. As long as F_B and $\widetilde{F_B}$ are close enough, we keep the encoder E_P^2 no longer changing. At last, the network training of binary image restoration from the generated point clouds is completed.

In the process of generating delicate point clouds, the input RGB image is first used to reconstruct the preliminary 3D point clouds with the encoder E_I^2 and decoder D_P, and then the preliminary point clouds are rendered into binary images with the trained encoder E_P^2 and decoder D_I. The generated binary images are compared with the input binary images, and the 2D MSE loss between these two binary images is used as self-supervised loss, and the 3D chamfer loss is used as a reconstruction loss. The self-supervised learning requires less training data than the vanilla supervised learning, and extracts latent features of the input data, which makes the network have favorable generalization ability. In addition, the proposed pose net used to estimate the pose of the input image can solve the problem of pose ambiguity during the reconstruction process.

In detail, the training process of SSL-Net can be divided into three stages. The first stage is to generate 3D point clouds from single-view RGB images: using the point cloud auto-encoder network (E_P^1, D_P) to learn the latent features $\widetilde{F_P}$ of the input point clouds, and training the image encoder E_I^2 to obtain the latent image features F_P, as well as generating preliminary point clouds from the features $\widetilde{F_P}$ with the trained decoder D_P. In the second stage, the binary images are recovered from the preliminary point clouds: using the image auto-encoder network (E_I^1, D_I) to learn the latent image features $\widetilde{F_B}$, and training the point clouds encoder E_P^2 to get the latent features F_B with the supervision of $\widetilde{F_B}$, as well as recovering the binary images corresponding to the preliminary point clouds. The third stage combines the first two stages with pose net to achieve joint optimization. The details of each module are described below.

A. 3D POINT CLOUD RECONSTRUCTION FROM IMAGE

The 3D point cloud reconstruction is based on a pre-trained auto-encoder, which consists of an encoder and a decoder, the encoder learns the latent features from the 3D point clouds and the decoder reconstruct the 3D point clouds from the latent features. Based on the pre-trained auto-encoder, 2D images are used as input to train a network that output the features approximating the latent features of auto-encoder, and the latent features can be decoded into 3D point clouds further.

1) 3D POINT CLOUD AUTO-ENCODER

First, we train the point cloud auto-encoder network (E_P^1, D_P) to obtain a priori representation of the point clouds with small amount of data, mapping the output geometry to input image through the iterative optimization process. As shown in FIGURE 3, the input is represented as $\widetilde{S_P} \subseteq \mathbb{R}^{B\times N\times 3}$ (B represents batch, N represents the number of points in a target three-dimensional structure), and the output is mapped to $S_P \subseteq \mathbb{R}^{B\times N\times 3}$. The network architecture of encoder E_P^1 is similar to Pointnet [27], in which 5 layers of 1D convolution is used to obtain the features with the dimension of $B\times N\times 512$, followed by max pooling to generate latent features $\widetilde{F_P} \subseteq \mathbb{R}^k$ ($k = 512$) with the dimension of $B \times 512$. The decoder D_P is composed of three fully connected layers, and the output is reshaped to $B \times 2048 \times 3$. In this process, we use chamfer distance [18] as supervisory signal. The chamfer loss function is expressed as:

$$L_{CD}(\widetilde{S_P}, S_P) = \sum_{x\in\widetilde{S_P}} \min_{y\in S_P} \|x-y\|_2^2 + \sum_{y\in S_P} \min_{x\in\widetilde{S_P}} \|x-y\|_2^2. \quad (1)$$

2) POINT CLOUD GENERATION

The auto-encoder enables the network to reconstruct point clouds from latent features $\widetilde{F_P}$ by decode D_P. In order to recover the point clouds from a single image, we only need to train an image encoder E_I^2 to approximate the image feature $\widetilde{F_P}$ to the latent point cloud features F_P. The feature loss L_{F_P} is as follows:

$$L_{F_P}(\widetilde{F_P}, F_P) = \sum\nolimits_{w\in\widetilde{F_P}, z\in F_P} |w-z|, \quad (2)$$

where w and z represent the elements in $\widetilde{F_P}$ and F_P, respectively.

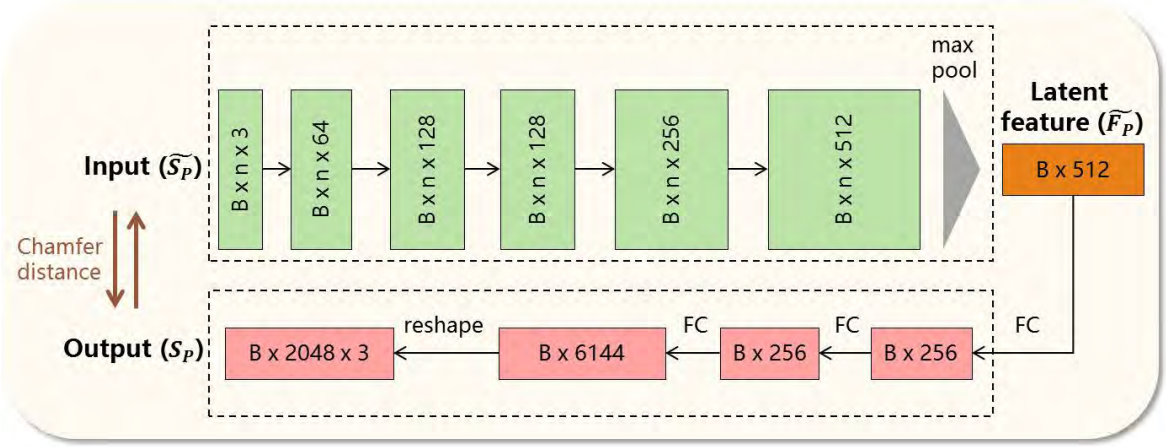

FIGURE 3. 3D point cloud auto-encoder. First, the ground truth point clouds $\widetilde{S_P}$ are sent to the encoder E_P^1, and then the latent features $\widetilde{F_P}$ are extracted from 5 convolution layers. At last, the reconstructed point clouds S_P can be obtained from $\widetilde{F_P}$ through 3 fully connected layers.

TABLE 1. Parameters of binary image auto-encoder network. The encoder is composed of convolution layers except the last layer, and the decoder consists of a structure with deconvolution and convolution alternately.

E_I^1 / E_I^2				D_I			
L.No.	Layer	Filter size/Strides	Output size	L.No.	Layer	Filter size/Strides	Output size
$1-2$	conv	3×3 / 1	$128\times128\times32$	1	deconv	5×5 / 4	$4\times4\times512$
3	conv	3×3 / 2	$64\times64\times64$	2	conv	3×3 / 1	$4\times4\times512$
$4-5$	conv	3×3 / 1	$64\times64\times64$	3	deconv	3×3 / 2	$8\times8\times512$
6	conv	3×3 / 2	$32\times32\times128$	$4-5$	conv	3×3 / 1	$8\times8\times512$
$7-8$	conv	3×3 / 1	$32\times32\times128$	6	deconv	3×3 / 2	$16\times16\times256$
9	conv	3×3 / 2	$16\times16\times256$	$7-8$	conv	3×3 / 1	$16\times16\times256$
$10-11$	conv	3×3 / 1	$16\times16\times256$	9	deconv	3×3 / 2	$32\times32\times128$
12	conv	3×3 / 2	$8\times8\times512$	$10-11$	conv	3×3 / 1	$32\times32\times128$
$13-14$	conv	3×3 / 1	$8\times8\times512$	12	deconv	3×3 / 2	$64\times64\times64$
15	conv	3×3 / 1	$8\times8\times512$	$13-14$	conv	3×3 / 1	$64\times64\times64$
16	conv	5×5 / 2	$4\times4\times512$	15	deconv	3×3 / 2	$128\times128\times32$
17	FC	–	512	16	conv	5×5 / 1	$128\times128\times1$

As shown in the first column of TABLE 1, the encoder E_I^2 is composed of convolutional layers except the last layer. The network extracts deep features from the input image and finally gets a 512-dimensional feature vector through the fully connection layer. Once the extracted image features F_P are fit with point cloud latent features $\widetilde{F_P}$, the 3D point clouds can be reconstructed by decoder D_P of the point cloud auto-encoder.

B. BINARY IMAGE GENERATION FROM 3D POINT CLOUDS

For 3D point clouds lack of structure information, they are stored as array of 3D points without adjustment relationship. 3D point cloud loss is not straightforward to measure the similarity of two point clouds. Thus, we project the 3D point cloud into a binary image to recover the structure information, and the binary image of a point cloud is compared with the input binary image as a supervisory loss. In order to train the network in an end-to-end manner, we propose a binary image generation network similarly with point cloud generation network, in which, the binary image auto-encoder is first trained to learn the latent image features that can be reconstructed to binary image, and an encoder is re-trained to extract features from point clouds that can approximate the latent image features.

1) BINARY IMAGE AUTO-ENCODER

Consistent with the subsection III-A-1), we perform image auto-encoder (E_I^1, D_I) on the binary image of the input RGB image. The network structure is shown in FIGURE 2. Firstly, binarization is performed on the input image to obtain the real binary image $\widetilde{B_I}$. And then $\widetilde{B_I}$ is sent to the image encoder to obtain the latent features $\widetilde{F_B} \subseteq \mathbb{R}^k$ ($k = 512$) of the binary image. The parameters of the encoder E_I^1 are shown in the left column of TABLE 1. The network is composed of convolutional layers excepspatial features are decoded, and the image content is filled with a transposed convolutional layer, so that the image is gradually enriched to recover the original binary image. The parameters of each layer are shown in the right column of TABLE 1.

Mean squared error (MSE) is used as loss function to evaluate the reconstruction error of the binary image. Each pixel value of the real binary image and the decoder output are represented by p and q respectively, where $p \in \widetilde{B_I}$, $q \in B_I{}^*$.

Each image contains M pixel values. Therefore, the loss of the two binary images can be expressed as follows:

$$L_{MSE}(\widetilde{B_I}, B_I{}^*) = \frac{1}{M}\sum\nolimits_{p\in\widetilde{B_I}, q\in B_I{}^*}(p-q)^2. \tag{3}$$

Once the auto-encoders are trained, the next stage includes training the image encoder E_I^2 and point cloud encoder E_P^2 to fit their output features F_P and F_B, respectively.

2) BINARY IMAGE PREDICTION

The binary image auto-encoder is used to to extract latent image features. In order to predict the binary images from point clouds, we need to train an extra encoder to extract features that approximate the latent image features. The network structure of the encoder E_P^2 is the same as E_P^1. Before performing the binary image prediction, we use the learned image pose (R_m, t_m) to translate and rotate the point clouds until they are transformed into the same coordinate system:

$$x_i' = R_m x_i + t_m, \quad i \in [0, N-1], \tag{4}$$

where, N denotes the number of points, and each point x_i is transformed to x_i' by the matrix $[R, t]$.

Given the output image pose, the transformed 3D point cloud shape has a one-by-one correspondence with its binary image. After the operation of the encoder E_P^2, we obtain the point cloud features F_B, which is used to calculate the L1 regularization loss with the binary image features $\widetilde{F_B}$:

$$L_{F_B}(\widetilde{F_B}, F_B) = \sum\nolimits_{u\in\widetilde{F_B}, v\in F_B}|u-v|. \tag{5}$$

The loss function L_B of binary image prediction is expressed as:

$$L_B = \lambda L_{F_B} + L_{MSE}. \tag{6}$$

C. POINT-CLOUD GENERATION NETWORK WITH SELF-SUPERVISED LEANING

The point cloud generation network reconstructs the 3D point clouds from a single image (Section III-A), and the point cloud is transformed into a binary image by a binary image network (Section III-B). These two networks are combined and fine-tuned in an end-to-end manner to provide self-supervised signal. Since the pose information of 3D point clouds is unobtainable, we need to estimate the pose of input image to predict the binary image from the 3D point cloud, and the difference of the two binary images provides the extra self-supervised loss.

We collect the Euler angles and translation vector of each RGB image from ground truth to estimate its viewpoint, and design the pose net to extract the features of the input image as shown in FIGURE 4. The structure of pose net shares the parameters with E_I^2 except the last layer, which uses the fully connected layer to output a six-dimensional vector to represent the image pose information. The six-dimensional vector is multiplied by the reconstructed point cloud according to equation 4. At last, a binary image with the same pose of input RGB image can be generated from the 3D point cloud.

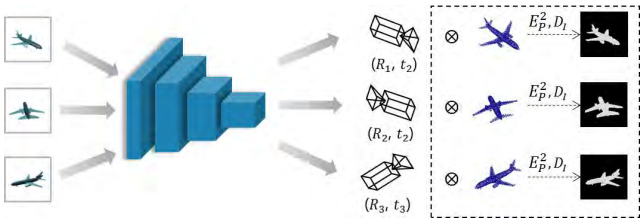

FIGURE 4. Image pose estimation. The feature is extracted from the input, and the pose information of the image is then estimated, thereby obtaining a three-dimensional model of a specific perspective.

The image viewpoint information is composed of six parameters $(\alpha, \beta, \gamma, a, b, c)$, where (α, β, γ) represent three direction angles (yaw, pich and roll), $t = (a, b, c)$ represents the translation vector. The direction angle is converted to a rotation matrix R using equation 7, as shown at the bottom of the next page.

The real binary image and the recovered binary image constitute the optimization loss function:

$$L_{opt}(\widetilde{B_I}, B_I) = \frac{1}{M}\sum\nolimits_{p\in\widetilde{B_I}, q\in B_I}(p-q)^2, \tag{8}$$

where, each image contains M pixel values. Each pixel of the real binary image and the decoder output is represented by $p \in \widetilde{B_I}$ and $q \in B_I{}^*$ respectively.

The generated 3D model is simultaneously rendered into an image and compared with input image. In addition, a pose estimation network is integrated into the 3D point cloud generation network to eliminate the pose ambiguity of the input image, and the estimated pose is also used for rendering the 2D image with the same pose as input image from 3D point clouds.

IV. EXPERIMENTAL RESULTS

In this section, we extensively evaluate the proposed self-supervised network method qualitatively and quantitatively. In addition to comparing with previous 3D shape generation works, we also analyze the importance of each component in our model by ablation study.

A. EXPERIMENTAL SETUP

1) DATA PREPARATION

Shapenet dataset is used to train and evaluate the performance, which is provided by Chang *et al.* [10] and it is a collection of 3D CAD models organized by the

$$R(\alpha,\beta,\gamma) = \begin{bmatrix} cos\alpha cos\beta & cos\alpha sin\beta sin\gamma - sin\alpha cos\gamma & cos\alpha sin\beta cos\gamma - sin\alpha sin\gamma \\ sin\alpha cos\beta & sin\alpha sin\beta sin\gamma - cos\alpha cos\gamma & sin\alpha sin\beta cos\gamma - cos\alpha sin\gamma \\ -sin\beta & cos\beta sin\gamma & cos\beta cos\gamma \end{bmatrix}. \tag{7}$$

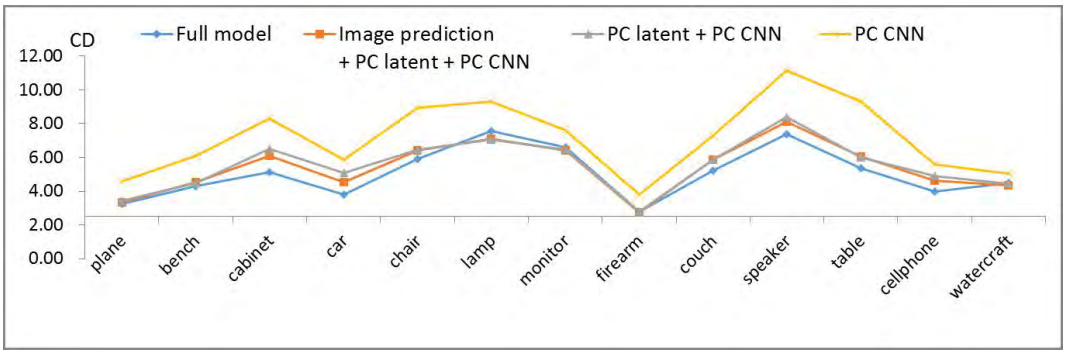

FIGURE 5. Ablation study of chamfer distance. These curves reflect the effect of each module on the full model quantitatively, the full model consists of point cloud latent feature generation, point cloud auto-encoder and binary image prediction. For CD, smaller is better.

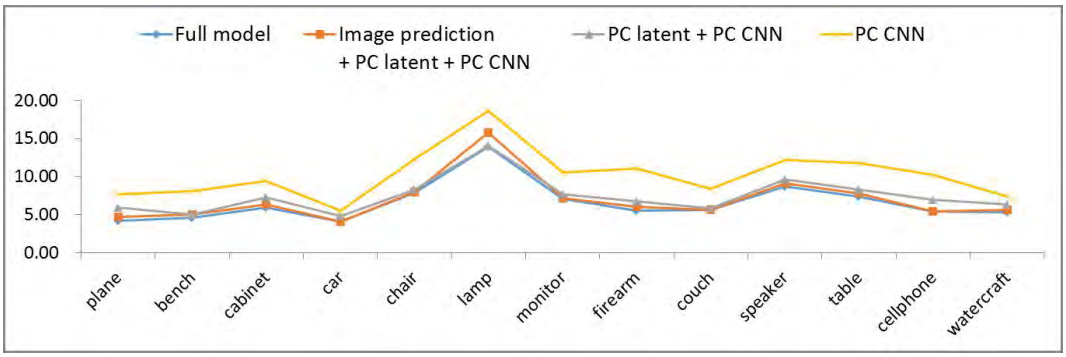

FIGURE 6. Ablation study of EMD. These curves reflect the effect of each module on the full model quantitatively, the full model consists of point cloud latent feature generation, point cloud auto-encoder and binary image prediction. For EMD, smaller is better.

WordNet [34] hierarchy. Shapenet includes 50k models belonging to 13 object categories. Each model is rendered from various camera viewpoints, and the corresponding camera intrinsic and extrinsic matrices are recorded. Each CAD model corresponds to 24 rendered RGB images from different azimuth angles with the resolution of $128 \times 128 \times 3$. For fair comparison, we used the same training/testing split as Choy et al. [14], that is, 80% of the dataset is used for training, and the remaining 20% is used for testing.

2) EVALUATION METRICS

In recent years, chamfer distance (CD) and earth mover's distance (EMD) [3] have become two widely used evaluation methods in the field of 3D reconstruction. Therefore, we report both CD (Equation 1) as well as EMD as performance metrics. The EMD between two point sets $\widetilde{S_P}$ and S_P is given by:

$$L_{EMD}(\widetilde{S_P}, S_P) = \min_{\phi:\widetilde{S_P}\to S_P} \sum_{x\in\widetilde{S_P}} \|x - \phi(x)\|_2, \qquad (9)$$

where $\phi : \widetilde{S_P} \to S_P$ is a bijection. Obviously, for CD and EMD, the smaller is better.

For computing the metrics, consistent with 3D-LMNet, we re-normalize both the ground truth and predicted point clouds within a bounding box of length 1 unit and apply the iterative closest point algorithm (ICP) [35] on them for better alignment, providing the predicted point clouds with the same view pose as the ground truth to reduce calculation errors.

3) BASELINES

We compare our SSL-Net with several state-of-the-art single image 3D reconstruction methods. Since the metrics are defined on point clouds, we can evaluate PSG-Net [18] and 3D-LMNet [19] directly on their outputs. For 3D-R2N2 [14], we can evaluate it by uniformly sampling points from mesh created using the Marching Cube [36] method.

4) SETTINGS

In our method, the input image size is $128\times128\times3$ and binary image size is 128×128. The reconstructed point cloud consists of 2048 points (each point consists of three coordinates x, y and z). The network is implemented in Tensorflow and optimized using the Adam optimizer with weight decay 1e-5. The batch size is 32, the total number of training epoch in the three training stages are 500, 30 and 30 respectively, and the learning rate is set to 1e-5. The total training time is 72 hours on Nvidia GTX 1080. It takes about 107ms to generate a 3D point cloud model in testing.

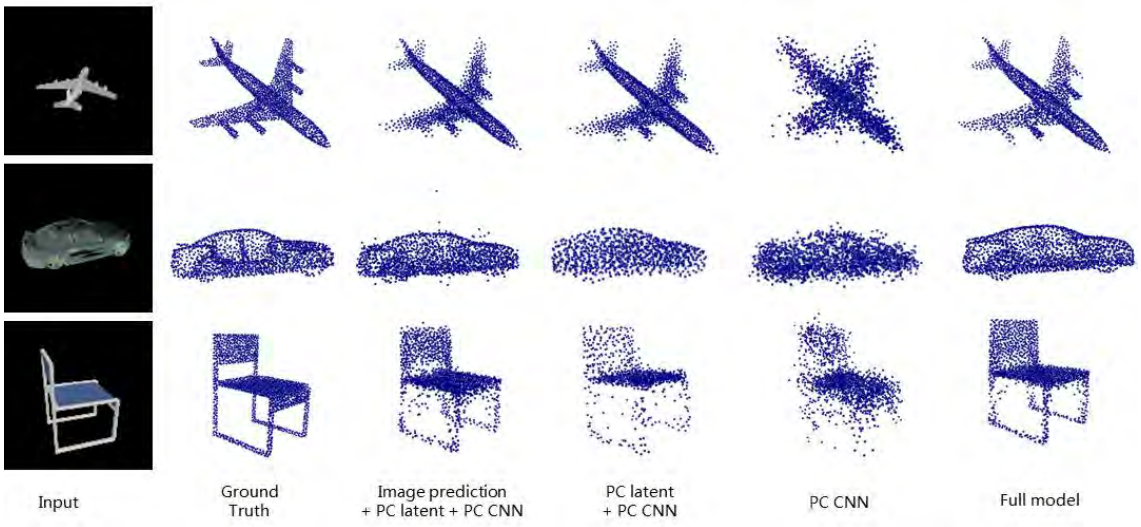

FIGURE 7. Ablation study of the proposed method. These results demonstrate that all the components presented in this work contribute to the final performance.

TABLE 2. Comparisons with state-of-the-art on Shapenet dataset (per category), including 3D-R2N2 [14], PSG-Net [18] and 3D-LMNet [19] (smaller is better), where all numbers are scaled by 100. Best results under each threshold are bolded.

Category	EMD				CD			
	3D-R2N2	PSG-Net	3D-LMNet	Proposed	3D-R2N2	PSG-Net	3D-LMNet	Proposed
plane	6.06	6.38	4.77	**4.73**	8.95	3.74	3.34	**3.25**
bench	11.36	5.88	4.99	**4.91**	18.91	4.63	4.55	**4.30**
cabinet	25.2	**6.04**	6.35	6.21	7.35	6.98	6.09	**5.15**
car	16.7	4.87	4.10	**4.07**	8.45	5.20	4.55	**3.78**
chair	14.66	9.63	8.02	**7.87**	14.32	6.39	6.41	**5.93**
lamp	14.24	16.17	15.80	**13.92**	40.09	**6.33**	7.10	7.55
monitor	16.67	7.59	7.13	**7.10**	17.07	**6.15**	6.40	6.6
firearm	6.88	8.48	6.08	**5.59**	9.93	2.91	**2.75**	2.77
couch	21.14	7.42	**5.65**	5.66	11.35	6.98	5.85	**5.23**
speaker	27.32	**8.70**	9.15	8.72	15.07	8.75	8.10	**7.36**
table	16.41	8.40	**7.82**	8.04	11.16	6.00	6.05	**5.37**
cellphone	9.12	**5.07**	5.43	5.39	11.37	4.56	4.63	**3.96**
watercraft	9.35	6.18	5.68	**5.66**	12.15	4.38	**4.37**	4.48
Mean	15.01	7.75	7.00	**6.76**	14.32	5.62	5.40	**5.06**

B. ABLATION STUDY

In this section, we perform the ablation study to analyze the importance of each component in our model. FIGURE 5 and FIGURE 6 plot the CD and EMD evaluation results by successive removal of one component from our full model, the full model consists of point cloud latent feature generation, point cloud auto-encoder and binary image prediction. We argue that quantitative analysis does not exhaustively reflect the quality of the recovered 3D geometry, so each component is also qualitatively visualized to show the contribution in our system. The results are shown in FIGURE 7. These results demonstrate that all the components presented in this work contribute to the final performance.

1) POSE NET

We first remove the pose net module, and the $[R, t]$ matrix can be only obtained from the known data. It can be seen from the third column of FIGURE 7, the predicted point clouds have more noise points than the full model, and the values of CD and EMD are worse either. Therefore, the ability to distinguish image perspective can reduce the noise in the shape of the object, making the surface more detailed.

2) IMAGE PREDICTION BLOCK

On the basis of the previous step, we then remove the binary image prediction module, which is used to self-supervising with the ground truth. It can be seen that the predicted 3D shapes become less close. The quantitative results (grey lines in FIGURE 5 and FIGURE 6) also demonstrate the effectiveness of the binary image prediction module.

3) PC LATENT BLOCK

Finally, we demonstrate the role of point cloud auto-encoder networks in object reconstruction. The chamfer distance is used as a loss function to generate 3D point clouds without passing through the auto-encoder. As can be seen

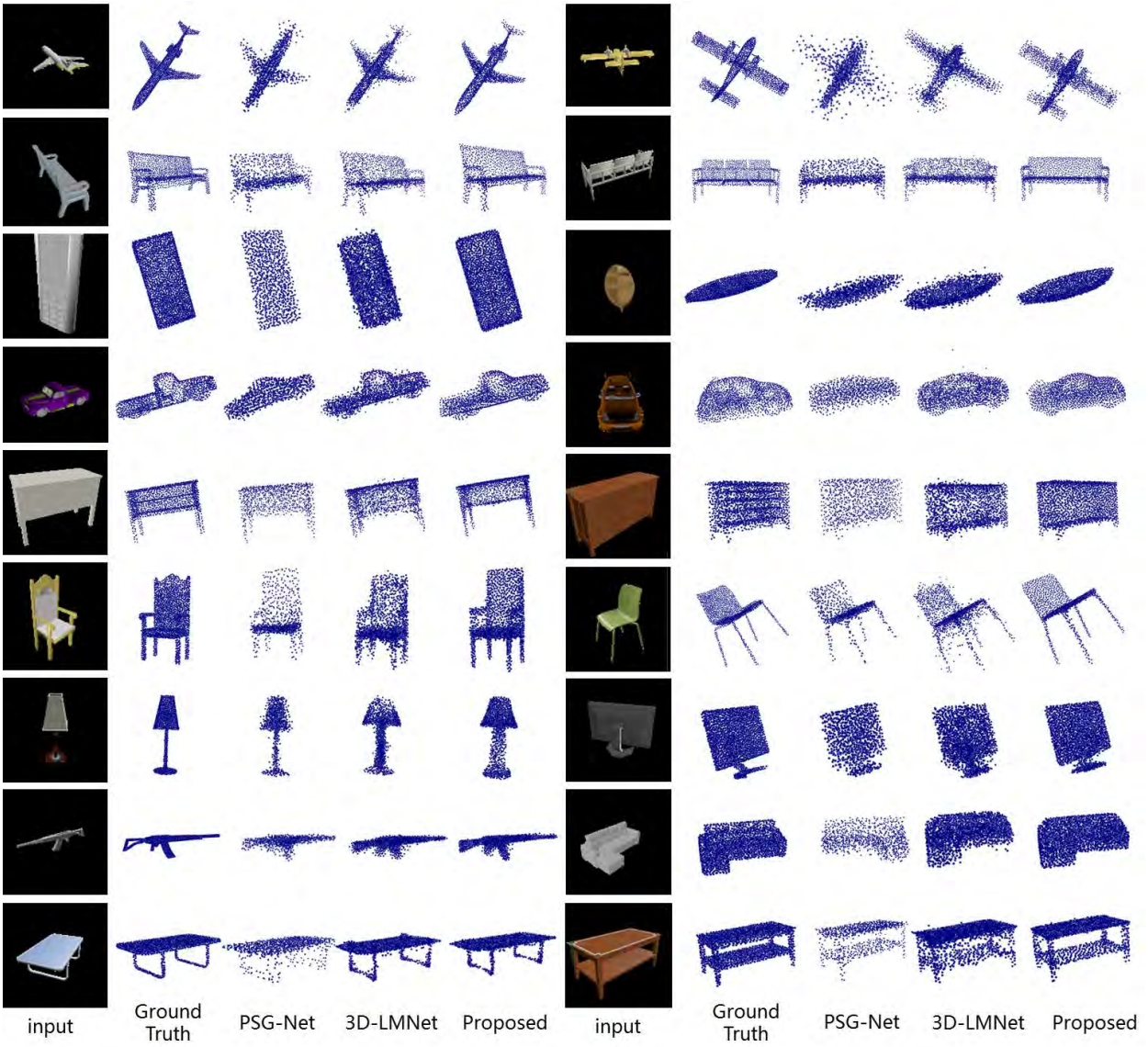

FIGURE 8. Qualitative results compared to state-of-the-art methods on ShapeNet, including the input images, ground truth, PSG-Net [18], 3D-LMNet [19] and our proposed method.

in FIGURE 5 and FIGURE 6 (yellow line) that both quantitative and qualitative results are not well. In this case, only the outline of the model can be reconstructed without rich detail, therefore, the point clouds tend to be very sparse. For example, when generating a chair, the legs of the chair cannot be fully reconstructed.

C. COMPARISONS WITH STATE-OF-THE-ART

TABLE 2 shows the performance of CD and EMD of 13 categories on the Shapenet dataset when comparing with three different state-of-the-art methods for single-view 3D reconstruction. Meanwhile, we also compared the average accuracy with more state-of-the-art methods which all generate the 3D model from a single RGB image. Among them, 3D-R2N2 [14] proposes a voxel grid method, N3MR [38], Pixel2Mesh [32] and AtlasNet [37] reconstruct the 3D mesh, 3D-PT-Generation [20], PSG-Net [18], 3D-LMNet [19] and 3D-PSRNet [40] represent the 3D shape with point clouds. It can be seen that the proposed method achieves the state-of-the-art results in both CD as well as EMD. Our network outperforms the state-of-the-art methods in 9 out of 13 categories in Chamfer and 8 out of 13 categories in the EMD metric, while also obtaining lower overall mean scores. Among the compared methods, the most advanced one is 3D-LMNet, which uses auto-encoder to extract the latent features of point clouds and obtains better results. However, this method lacks of sufficient self-constrained conditions. Therefore, we propose to apply self-supervised learning to achieve more detailed and optimized three-dimensional shapes. To demonstrate this, we visualize the reconstruction results and

FIGURE 9. Visualization of binary image recovery: (a) Input binary image; (b) Input RGB image; (c) Reconstructed point cloud; (d) Recovered binary image. For each model, the input images from three different orientations are chosen to obtain the corresponding binary images.

FIGURE 10. Qualitative results of real-world images. We use the model trained from the ShapeNet dataset to run directly on the real images using the image cropping method and each point cloud model consists of 2048 points.

compare it with the predicted shapes of PSG-Net [18] and 3D-LMNet [19] as shown in FIGURE 8. We can see that the 3D point clouds generated by PSG-Net only uses 1024 points to represent the shape of the object, the resolution of which is too low to describe the complete surface details of the object. 3D-LMNet uses 2048 points to represent the shape

TABLE 3. Comparisons with more state-of-the-art on Shapenet dataset (average CD value of each category), where all numbers are scaled by 100 (smaller is better). Best results under each threshold are bolded.

Method	EMD	CD
N3MR [38]	133.86	26.29
3D-R2N2 [14]	15.01	14.32
3D-PT-Generation [20]	–	6.29
Pixel2Mesh [32]	13.8	5.91
PSG-Net [18]	7.75	5.62
3D-LMNet [19]	7.00	5.40
3D-PSRNet [40]	7.29	5.26
AtlasNet [37]	–	5.11
Proposed	**6.76**	**5.06**

of the object. The reconstruction result is much better than PSG-Net, but it still can't achieve quite detailed surface in some cases. For example, in the first row of FIGURE 8, some noise points around the aircraft fuselage cause the wing partial missing, and the shape of the object is not clear enough. While our method has stronger applicability to images of different blurred perspectives by joint optimization network. The design of multiple loss functions also enables the predicted point clouds to have reasonable constraints and detailed surface while having high degrees of freedom. For fair comparison, we adopt the same evaluation method as 3D-LMNet, whose metric is calculated at 1024 points after ICP alignment [35] with the ground truth.

D. VISUALIZATION OF BINARIZATION OF 3D POINT CLOUDS

The binarization of 3D point clouds enables the network to compared with input image, which provides a complete self-supervised signal. This section visualizes the predicted binary image from the 3D point clouds. For images with different viewpoints of the same object, the network can generate an 3D point cloud with the corresponding perspective by estimating image pose matrix $[R, t]$, and the binarization of the 3D point clouds retain the same pose with the input image.

FIGURE 9 shows some visualization examples of each component of SSL-Net, including the input binary image, input RGB image, generated 3D point clouds, and recovered binary image. For each model, the input images from different poses are chosen to recover their own 3D point clouds, and predict the corresponding binary images. Although the edge of the restored binary image is slightly blurred, it still can accurately distinguish the poses of the same object. The restored binary image is similar to the input binary image. By comparing these two binary images, a self-supervised network is formed, and the 3D point clouds are optimized by the self-supervised signal from 2D binary images.

E. MORE RESULTS AND APPLICATIONS TO REAL WORLD DATA

To evaluate the generalization ability of the proposed method, we qualitatively evaluate our network on real-world images. We use the model trained from the ShapeNet dataset to run directly on the real images using the image cropping method provided by PSG-Net [18], and the results are shown in FIGURE 10. It is illustrated that our model trained on synthetic data has excellent generalization capabilities over the real-world images with various categories.

V. CONCLUSION

In this paper, we propose a cascaded self-supervised network to reconstruct 3D point clouds from a single image, after generating 3D point clouds, the network framework can further restore the binary image of the input as an extra self-supervision. In addition, the image pose estimation enables the network to distinguish image viewpoints even when the angle is blurred, so that it can still be effectively constrained to generate reasonable point clouds. Our network outperforms the current prestigious methods in both chamfer distance and EMD metric. The experimental results of the single-image 3D reconstruction indicate that we can generate more accurate and more detailed 3D shapes than state-of-the-art methods. The visualization of binarization of input images and point clouds depicts that the extra self-supervised loss can improve the accuracy of 3D point clouds. The self-supervised learning pipeline for 3D generation is well worth studying and can achieve great improvements in 3D object reconstruction.

REFERENCES

[1] A. Sharma, O. Grau, and M. Fritz, "VConv-DAE: Deep volumetric shape learning without object labels," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Apr. 2016, pp. 1–10.
[2] D. Angela and N. Matthias, "3DMV: Joint 3D-multi-view prediction for 3D semantic scene segmentation," Mar. 2018, *arxiv:1803.10409*. [Online]. Available: https://arxiv.org/abs/1803.10409
[3] J. Varley, C. DeChant, A. Richardson, J. Ruales, and P. Allen, "Shape completion enabled robotic grasping," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2017, pp. 2442–2447.
[4] Y. Gao and H. J. Lee, "Local tiled deep networks for recognition of vehicle make and model," *Sensors*, vol. 16, no. 2, p. 226, 2016.
[5] J. L. Schönberger and J.-M. Frahm, "Structure-from-motion revisited," in *Proc. IEEE CVPR*, Jun. 2016, pp. 4104–4113.
[6] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, J. J, Leonard, "Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age," *IEEE Trans. Robot.*, vol. 32, no. 6, pp. 1309–1332, Dec. 2016.
[7] Y. Li, R. Bu, M. Sun, W. Wu, X. Di, and B. Chen, "PointCNN: Convolution on $\mathcal{X}$-transformed points," in *Proc. Conf. Workshop Neural Inf. Process. Syst. (NIPS)*, Nov. 2018, pp. 820–830.
[8] R. Mahdi, O. Markus, and L. Vincent, "Feature mapping for learning fast and accurate 3D pose inference from synthetic images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Mar. 2018, pp. 4663–4672.
[9] C. Liu, J. Wu, and Y. Furukawa, "FloorNet: A unified framework for floorplan reconstruction from 3D scans," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Mar. 2018, pp. 201–217.
[10] A. X. Chang, T. Funkhouser, L. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su, J. Xiao, L. Yi, and F. Yu, "ShapeNet: An information-rich 3D model repository," Dec. 2015, *arXiv:1512.03012*. [Online]. Available: https://arxiv.org/abs/1512.03012
[11] 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 Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 2974–2983.
[12] C. Kong, C.-H. Lin, and S. Lucey, "Using locally corresponding cad models for dense 3D reconstructions from a single image," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5603–5611.
[13] A. Kurenkov, J. Ji, A. Garg, V. Mehta, J. Gwak, C. Choy, and S. Savarese, "DeformNet: Free-form deformation network for 3D shape reconstruction from a single image," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2018, pp. 858–866.

[14] 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. (ECCV)*, Sep. 2016, pp. 628–644.

[15] B. Yang, S. Rosa, A. Markham, N. Trigoni, and H. Wen, "Dense 3D object reconstruction from a single depth view," *IEEE Trans. Pattern Anal. Mach. Intell.*, to be published.

[16] C. Hne, S. Tulsiani, and J. Malik, "Hierarchical surface prediction for 3D object reconstruction," in *Proc. Int. Conf. 3 Dimensional Vis. (3DV)*, vol. 1, pp. 412–420, 2017.

[17] 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. Workshop (ICCVW)*, vol. 1, Mar. 2017, pp. 679–688.

[18] F. Haoqiang, S. Hao, and G. Leonidas, "A point set generation network for 3D object reconstruction from a single image," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 1, Jan. 2017, pp. 2463–2471.

[19] P. Mandikal, N. Murthy, M. Agarwal, and R. V. Babu, "3D-LMNet: Latent embedding matching for accurate and diverse 3D point cloud reconstruction from a single image," in *Proc. Brit. Mach. Vis. Conf. (BMVC)*, Jul. 2018, pp. 1–19.

[20] C. H. Lin, C. Kong, and S. Lucey, "Learning efficient point cloud generation for dense 3D object reconstruction," in *Proc. AAAI Conf. Artif. Intell. (AAAI)*, Jun. 2018, pp. 1–10.

[21] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge, U.K.: Cambridge Univ. Press, Jun. 2003, pp. 1333–1341.

[22] V. Sudheendra, R. Susanna, S. Cordelia, S. Rahul, and F. Katerina, "SFM-Net: Learning of structure and motion from video," Apr. 2017, *arXiv:1704.07804*. [Online]. Available: https://arxiv.org/abs/1704.07804

[23] A. Wang, Z. Fang, Y. Gao, X. Jiang, and S. Ma, "Depth estimation of video sequences with perceptual losses," *IEEE Access*, vol. 6, pp. 30536–30546, Jun. 2018.

[24] F. Steinbrcker, C. Kerl, and D. Cremers, "Largescale multi-resolution surface reconstruction from RGBD sequences," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2013, pp. 3264–3271.

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

[26] Y. Li, S. Pirk, H. Su, C. R. Qi, and L. J. Guibas, "FPNN: Field probing neural networks for 3D data," in *Proc. Conf. Workshop Neural Inf. Process. Syst. (NIPS)*, May 2016, pp. 307–315.

[27] R. Q. Charles, H. Su, M. Kaichun, and L. J. Guibas, "PointNet: Deep learning on point sets for 3D classification and segmentation," in *Proc. Int. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2017, pp. 77–85.

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

[29] M. Firman, O. M. Aodha, S. Julier, and G. J. Brostow, "Structured prediction of unobserved voxels from a single depth image," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 5431–5440.

[30] D. Stutz and A. Geiger, "Learning 3D shape completion under weak supervision," *Int. J. Comput. Vis.*, vol. 19, no. 7, pp. 1–20, Oct. 2018.

[31] 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)*, vol. 1, Aug. 2017, pp. 2107–2115.

[32] N. Wang, Y. Zhang, Z. Li, Y. Fu, W. Liu, and Y.-G. Jiang, "Pixel2Mesh: Generating 3D mesh models from single rgb images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Oct. 2018, pp. 55–71.

[33] D. Jack, J. K, Pontes, S. Sridharan, C. Fookes, S. Shirazi, F. Maire, and A. Eriksson, "Learning free-form deformations for 3D object reconstruction," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2018, pp. 1–16.

[34] C. Fellbaum and G. Miller, *WordNet: An Electronic Lexical Database*. Cambridge, MA, USA: MIT Press, 1998.

[35] P. J. Besl and N. D. McKay, "A method for registration of 3-D shapes," *Int. Soc. Opt. Photon.*, vol. 14, no. 2, pp. 239–249, 1992.

[36] W. E. Lorensen and H. E. Cline, "Marching cubes: A high resolution 3D surface construction algorithm," in *Proc. 14th Annu. Conf. Comput. Graph. Interact. Techn.*, 1987, pp. 163–169.

[37] G. Thibault, F. Matthew, G. K. Vladimir, C. R. Bryan, and A. Mathieu, "A Papier-Mâché approach to learning 3D surface generation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Feb. 2018, pp. 216–224.

[38] H. Kato, Y. Ushiku, and T. Harada, "Neural 3D mesh renderer," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Nov. 2017, pp. 3907–3916.

[39] Y. Yao, Z. Luo, S. Li, T. Shen, T. Fang, and L. Quan, "Recurrent MVSNet for high-resolution multi-view stereo depth inference," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Feb. 2019, pp. 1–15.

[40] P. Mandikal, N. K. L, and R. V. Babu, "3D-PSRNet: Part segmented 3D point cloud reconstruction from a single image," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2018, pp. 1–13.

RAN SUN was born in China, in 1994. She is currently pursuing the master's degree with the School of Electronic and Electrical Engineering, Shanghai University of Engineering Science, Shanghai, China. Her research interests include 3D reconstruction, computer vision, and machine learning.

YONGBIN GAO received the Ph.D. degree from Chonbuk National University, South Korea. He is currently a Faculty Member of the School of Electronic and Electrical Engineering, Shanghai University of Engineering Science, Shanghai, China. He has published numerous SCI papers in prestigious journals such as *Information Science and Pattern Recognition Letters*, and he has published in the areas of image processing, patter recognition, and computer vision.

ZHIJUN FANG received the Ph.D. degree from Shanghai Jiao Tong University, Shanghai, China. He is currently a Professor and the Dean of the School of Electronic and Electrical Engineering, Shanghai University of Engineering Science. His current research interests include image processing, video coding, and pattern recognition. He has received the GanPo 555 Talents Program Award and the One-Hundred, the One-Thousand, and the Ten-Thousand Talent Project Award of Jiangxi Province. He was the General Chair of the Joint Conference on Harmonious Human Machine Environment, in 2013, and the General Co-Chair of the International Symposium on Information Technology Convergence, from 2014 to 2017.

ANJIE WANG was born in China, in 1993. He is currently pursuing the master's degree with the School of Electronic and Electrical Engineering, Shanghai University of Engineering Science, Shanghai, China. He is currently a Visiting Scholar with the Institute of Digital Media, Peking University, Beijing, China. His research interests include SLAM, computer vision, and machine learning.

CENGSI ZHONG was born in China, in 1995. She is currently pursuing the master's degree with the School of Electronic and Electrical Engineering, Shanghai University of Engineering Science, Shanghai, China. Her research interests include action detection, computer vision, and machine learning.