IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received December 8, 2020, accepted December 20, 2020, date of publication December 23, 2020,
date of current version January 4, 2021.

Digital Object Identifier 10.1109/ACCESS.2020.3046951

Pixel2point: 3D Object Reconstruction From a Single Image Using CNN and Initial Sphere

AHMED J. AFIFI[1], JANNES MAGNUSSON[2], TOUFIQUE A. SOOMRO[3], (Member, IEEE), AND OLAF HELLWICH[1], (Senior Member, IEEE)

[1]Computer Vision and Remote Sensing, Technische Universität Berlin, 10587 Berlin, Germany
[2]Institute for Image Science and Computational Modeling in Cardiovascular Medicine, Charité–Universitätsmedizin Berlin, 13353 Berlin, Germany
[3]Department of Electronic Engineering, Quaid-e-Awam University of Engineering, Science, and Technology, Nawabshah 67480, Pakistan

Corresponding author: Ahmed J. Afifi (ahmed.afifi@campus.tu-berlin.de)

This work was supported by the German Research Foundation and the Open Access Publication Fund of TU Berlin.

ABSTRACT 3D reconstruction from a single image has many useful applications. However, it is a challenging and ill-posed problem as various candidates can be a solution for the reconstruction. In this paper, we propose a simple, yet powerful, CNN model that generates a point cloud of an object from a single image. 3D data can be represented in different ways. Point clouds have proven to be a common and simple representation. The proposed model was trained end-to-end on synthetic data with 3D supervision. It takes a single image of an object and generates a point cloud with a fixed number of points. An initial point cloud of a sphere shape is used to improve the generated point cloud. The proposed model was tested on synthetic and real data. Qualitative evaluations demonstrate that the proposed model is able to generate point clouds that are very close to the ground-truth. Also, the initial point cloud has improved the final results as it distributes the points on the object surface evenly. Furthermore, the proposed method outperforms the state-of-the-art in solving this problem quantitatively and qualitatively on synthetic and real images. The proposed model illustrates an outstanding generalization to the new and unseen images and scenes.

INDEX TERMS Single-view reconstruction, deep learning, point cloud, CNN.

I. INTRODUCTION

Single-view reconstruction is a long-standing ill-posed problem and fundamental to many applications such as object recognition and scene understanding. Single-view 3D reconstruction means using a single image of an object and utilizing it to infer the 3D structure of the object so that it can be viewed from all directions. For multi-view scenarios, a large variety of methods has been proposed which are able to present high-quality reconstruction results [10]. The challenge appears when a single input image is just available for the reconstruction process. Many approaches were proposed with restrictions and special assumptions on the input image to predict 3D geometry [19]. Single-view 3D reconstruction is a hard problem and it mainly depends on the available information and the imposed assumptions on the target object. This information or cues provide prior knowledge that helps in generating 3D shapes with plausible precision [19].

Before the deep learning era, many approaches have been proposed to solve single-view reconstruction depending on the object nature. Some of them were applied to real-world images without any knowledge of the image formation, and the output of these approaches is plausible. One class of these methods focus on curved objects and try to produce smooth objects. These methods define an energy function to minimize the object surface with respect to some constraints such as a fixed area or volume [18], [20], [28]. Other methods focus on piecewise planer objects and utilize semantic knowledge of object locations such as the sky and the ground locations in the image [7].

With the astonishing results obtained by applying deep learning on different computer vision problems, many 3D-based models have made great progress in solving different tasks using 3D data directly such as classification, object parts segmentation, and 3D shape completion. Also, the availability of large-scale datasets [5] encourages researchers to formulate and tackle the single-view reconstruction problem. Volumetric methods were first used to infer the 3D structure of an object from a single view [6]. However, volumetric representation suffers from information sparsity and the heavy computations during the training process. Also, this representation is ineffective in high-resolution outputs. To overcome

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

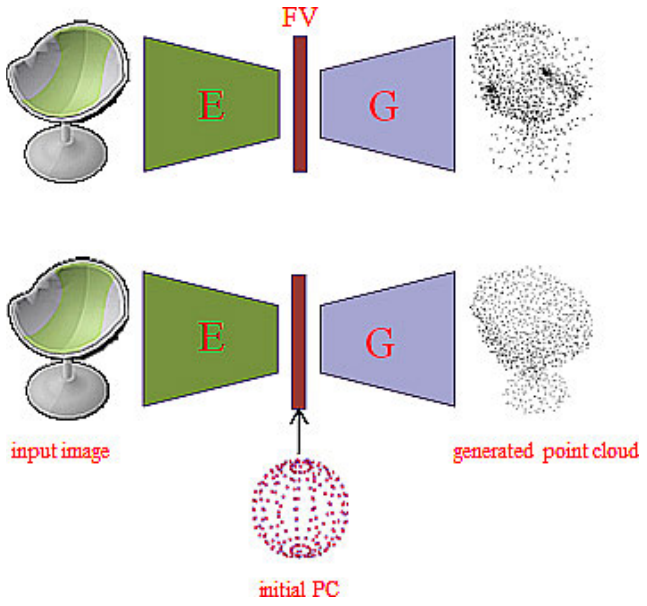


FIGURE 1. A general sketch of the proposed CNN model with different setups. Top: the proposed CNN without an initial point cloud. Bottom: the proposed CNN with the initial point cloud. E: Encoder, G: Generator, PC: Point Cloud, FV: Feature Vector.

this issue, recent works have used point clouds [8] as they are samples on the surface of the objects and effectively capture more object details.

In single-view reconstruction, the reprojection from 2D to 3D is ambiguous due to the loss of the depth information. To this end, we propose a CNN model that solves the task of single-view reconstruction. The model has an encoder-generator shape where the encoder extracts useful features from the input image and the generator infer the point clouds of the object shown in the 2D image. To generate more accurate point clouds, an initial point cloud is used to improve the reconstruction quality. We find that starting from an initial point cloud enforces the points to distribute equally on the shape surface and preserve the object parts. We summarize our contributions as follows: (1) we design a CNN model that can infer the 3D geometry of an object from a single image. The 3D object is represented as a point cloud. (2) Instead of directly inferring the point cloud, we propose to utilize an initial point cloud of a sphere shape to generate the final object point cloud. The experimental results (Sec. V-C) that using an initial point cloud helps in generating better and more accurate reconstruction (Fiqure 1). (3) We evaluate the proposed model on synthetic and real data quantitatively and qualitatively. Our model outperforms the state-of-the-art methods and shows significant results for the task of single-view reconstruction.

II. RELATED WORK

Inferring the 3D structure of an object from a single image is an ill-posed problem, but many attempts have been done such as SFM and SLAM [3], [9]. Moreover, ShapeFrom*X*, where *X* can be shadow, texture, etc. requires prior knowledge on the nature of the input image [2].

When applying deep learning models to generate 3D shapes or to solve other tasks such as segmentation, recognition, or object classification, the object representation plays an important role in designing the network. The most 3D data representations that are used in deep learning are volumetric data, meshes, and point clouds.

To extend the 2D convolutions to 3D, the volumetric representation has mostly been used. **Volumetric data** can be represented as a regular grid in the 3D space [27]. Voxels are used to visualize 3D data and show the distribution of the 3D object in the 3D space. Each voxel in the 3D space that describes the object can be classified into a visible, occluded, or self-occluded voxel according to the viewpoint. It is simple in implementation and compatible with the 3D convolutional neural network. 3D-GAN [26] proposed a generative adversarial network (GAN) to generate 3D objects from a probabilistic space using volumetric CNN. They mapped a low-dimensional probabilistic space to the 3D object space and by this, they outperform other unsupervised learning methods. Moreover, a 3D recurrent neural network (RNN) has been suggested to estimate the 3D shape of an object. 3D-R2N2 [6] proposed to use long short-term memory (LSTM) to infer the 3D geometry using many images of the target object from different perspectives. Recently, 3D-FHNet, which is a 3D fusion Hierarchical reconstruction method, was proposed that can perform 3D object reconstruction of any number of views [14]. The critical limitation of using the volumetric representation in the above-mentioned methods is the computational and the memory cost and the restriction on the output resolution. Also, fine-grained shape parts get lost because the voxel is represented as either occupied or unoccupied.

To avoid the limitation of the volumetric representation, mesh representation is more attractive for real applications as the shape details can be modeled accurately. **3D Meshes** are commonly used to represent 3D shapes. The structure of a 3D mesh comprises a set of polygons which are called faces [4]. These polygons are described using a set of vertices that describe how the mesh coordinates exist in the 3D space. Besides the 3D coordinates of the vertices, there is a connectivity list that specifies how the vertices are connected to each other. Applying deep learning models directly to generate meshes is a challenge as they are not regularly structured. A parameterization-based 3D reconstruction is proposed in [22] that generates geometry images which encode x; y; z surface coordinates. Three separated encoder-decoder networks were used to generate the geometry images. The networks take an RGB image or a depth image as an input and learn the x; y; and z geometry images respectively. Other methods proposed to estimate a deformation field from an input image and apply it to a template 3D shape to generate the reconstructed 3D model. Kuryenkov *et al.* [12] proposed DeformNet that takes an image and the nearest 3D shape to that image from a dataset as an input. Then, the template shape is deformed to match the input image using the Free Form Deformation layer (FFD). In [25], Pixel2Mesh is an

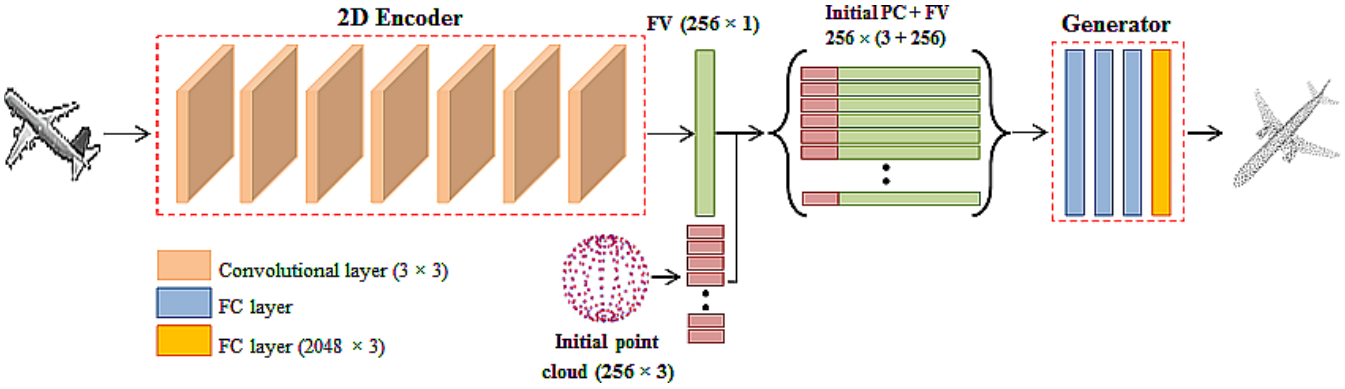


FIGURE 2. The proposed CNN Architecture with the initial Point Cloud.

end-to-end deep learning model that was proposed to generate a triangulated 3D mesh from a single image. The proposed network represented the 3D mesh in graph-based CNN (GCNN). It deforms an initial ellipsoid to leverage the perceptual features extracted from the input image. They adopted a coarse-to-find strategy that makes the deformation process stable. A limitation of using meshes for reconstruction is that the generated output is limited mostly to the initial mesh or the selected template as an initial shape to be deformed.

To overcome the above-mentioned limitations, point clouds are used to represent the 3D data. **3D Point Cloud** is a set of unordered 3D points that approximate the geometry of 3D objects [8]. Points can be represented either as a matrix of size $N \times 3$, a 3-channel grid of size $H \times W \times 3$ where each pixel encodes the (x,y,z) coordinates and $H \times W$ equals to the number of points, or depth maps from different known viewpoints. Point Set Generation Network (PSGN) [8] was the first proposed model to generate a point cloud of an object from a single image and outperforming the volumetric approaches. In RealPoint3D [29], the proposed network has two encoders; the first one extracts 2D features from the input image, the second encoder extracts 3D features from the nearest similar shape to the input image retrieved from the ShapeNet dataset. The extracted features from both encoders are integrated and forwarded to a decoder to generate fine-grained point clouds. The point cloud from the retrieved shape influenced the inferring process and generated finer point clouds. 3D-LMNET [16] trained a 3D point cloud auto-encoder and then learned the mapping from the 2D images to the learned embedded features. Another direction to generate the point cloud is to generate depth images of different perspectives and fuse them to generate the final point cloud. In [13], a generative modeling framework used 2D convolutional operation to predict multiple pre-defined depth images and use them to generate a dense 3D model. In [15], a two-stage training dense point cloud generation network was proposed. In the first stage, the network takes a single RGB image and generates a sparse point cloud. In the second stage, a generator network densifies the sparse point cloud and generate a dense point cloud. After training the two stages, the model becomes an end-to-end network that generates a dense point cloud from a single RGB image.

Our proposed model is different from the mentioned work. It has a simple design and utilizes an initial point cloud to predict the final point cloud accurately. The model has a single input and generates the point cloud directly without retrieving and utilizing a similar 3D model to the input image as proposed in [29]. Also, it doesn't use other 2D supervision such as silhouettes to infer the 3D object structure.

III. METHODOLOGY

Our main goal is to infer a complete 3D shape of an object from a single RGB image. We select point clouds to represent the generated output (Eq. 1). We set the number of the points generated from the CNN to $N = 2048$. From our experiments, this number of points is sufficient to cover the whole surface of the object and preserves the major structures.

$$S = \{(x_i, y_i, z_i)\}_{i=1}^{N} \tag{1}$$

A. 3D CNN MODEL

The proposed network is illustrated in Figure 2. It consists of two parts; the encoder part and the generator part. The encoder part is a set of consecutive 2D convolutional layers followed by ReLU as a non-linear activation function. These layers are used to extract the object features from the 2D input images. To predict the 3D point cloud of the object, an initial point cloud of a sphere shape is used. The initial point cloud is concatenated with the extracted features from the encoder. Then, it is fed into the generator part to get the final point cloud of the object, where fully connected layers (FC) are used to generate a $N \times 3$ matrix, where each row contains the coordinates of one point. Each network part is described in detail below.

Encoder Net. The role of the encoder part is to extract the distinction features from the input image that can correctly describe the object with details. It consists of consecutive layers of 2D convolutional layers and ReLU layers. The convolutional layers are seven layers. The first three convolutional layers are of sizes 32, 64, and 128, respectively. The remaining layers have a size of 256. All convolutional layers

have a kernel size of 3×3 and a stride of 2. The stride of 2 in the convolutional layers helps in decreasing the spatial size of the features as pooling layers do. Comparing to the pooling layers, the strided convolutional layers are trainable and can extract useful features. The size of the input image is 128×128. The extracted feature from the encoder has a size of $1 \times 1 \times 256$ which will be reshaped and concatenated with the initial point cloud.

Generator Net. The generator part is a simple network consisting of four fully connected layers (FC). The extracted feature vector from the encoder is reshaped to 1×256 and then concatenated with the initial point cloud. The initial point cloud has a sphere shape consisting of 256 equally spaced points. The reshaped feature is concatenated with each point of the initial point cloud, and the new feature has a size of $256 \times (3+256)$. Figure 2 shows the reshape and concatenation process. The new feature is fed into the generator. After three FC layers followed by ReLU, the generator ends with a fully connected layer that predicts the final point cloud with a shape of 2048×3.

The proposed network is different from other single-view reconstruction models as the proposed model utilizes an initial point cloud with a sphere shape for better inference of the final point cloud. In the results section, we will discuss and show the importance of using this setup and how it improves the final results.

B. LOSS FUNCTION

Selecting a suitable loss function to train the CNN model is a critical step. The nature of the problem, the dataset representation, and the output values are the points that should be considered when designing the loss function. The loss function measures the error between the inferred output and the corresponding ground-truth. According to the error, the model weights are optimized and updated. In our case, the loss function will measure the distance between the generated point cloud and the ground-truth shape. It should fulfill the following conditions; (1) the selected loss function should be efficient to compute and differentiable so that it can be used for the back-propagation step, and (2) it should be robust against the outliers [8].

So, the required loss function L between two 3D shapes, $S^{pred}, S^{gt} \subseteq \mathbb{R}^3$, is defined as:

$$L(\{S^{pred}\}, \{S^{gt}\}) = \sum d(S^{pred}, S^{gt}) \tag{2}$$

where S^{pred} and S^{gt} are the predicted 3D shape and the correspondence ground-truth shape, respectively.

Since the point cloud is an orderless representation, the loss function should be invariant to the ordering of the points. To this end, we propose to use and compare two different loss functions: Chamfer Distance (CD) [24] and Earth Mover's Distance (EMD) [21].

Chamfer Distance (CD). The Chamfer Distance between $S_1, S_2 \subseteq \mathbb{R}^3$ is defined as:

$$d_{CD} = \sum_{x \in S_1} \min_{y \in S_2} \|x - y\|_2^2 + \sum_{y \in S_2} \min_{x \in S_1} \|x - y\|_2^2 \tag{3}$$

In the first term of Eq. 3, for each point in the predicted point cloud, CD finds first the nearest neighbor in the ground-truth point cloud and sums the squared distance up. The second term of Eq. 3 does the same but from the ground-truth point cloud to the predicted point cloud. CD is piecewise smooth and continuous, and the search process is independent for each point. So, this function is parallelizable and produces high-quality results. The lower the value, the better and more accurate the generated shape. The drawback of CD is that there is no clear mechanism to enforce the uniformity of the generated point cloud because the optimization process leads to a minima where a subset of points account for the whole shape and cluster the remaining points.

Earth Mover's Distance (EMD). The EMD between S_1, $S_2 \subseteq \mathbb{R}^3$ is defined as:

$$d_{EMD} = \min_{\phi: S_1 \to S_2} \sum_{x \in S_1} \|x - \phi(x)\|_2 \tag{4}$$

where $\phi : S_1 \to S_2$ is a bijection and the size of S_1 and S_2 is equal, $s = |S_1| = |S_2|$.

In EMD, ϕ maps each point from S_1 to a one unique point in S_2. It enforces a point-to-point assignment between the two point clouds. EMD is differentiable and parallelizable, but computationally expensive (with respect to the time and the memory for high-resolution point clouds).

IV. EVALUATION

In this section, we outline the implementation details of the proposed architecture and the datasets used for training. We also discuss the testing datasets that will be used to evaluate and compare the proposed method against the state-of-the-art.

A. IMPLEMENTATION DETAILS

We implemented and trained the proposed model in TensorFlow [1]. The input image size is 128×128. For each object category, we trained a separate model. The encoder generates a latent feature of dimension 256. The generator network outputs a point cloud of size 2048×3. Adam optimizer [11] was used to optimize the network parameters with a learning rate of $5e^{-5}$ and a minibatch of size 32. We trained the model until the validation accuracy stopped increasing.

B. DATASET PREPARATION

1) ShapeNet

Reference [5] is a large-scale synthetic 3D dataset that is widely used in 3D research such as 3D model retrieval and reconstruction. **ShapeNetCore** is a subset of the ShapeNet dataset that we used in our experiment. It is manually cleaned and aligned. It has more than $50K$ unique 3D models which cover 55 common object categories. We focus on 13 categories and use the $80\% - 20\%$ train-test split provided by [5]. The input images provided by [6] are used during training, where each model is rendered from 24 different azimuth angles.

The 2D input images used for training and testing are provided by [6]. Each model in ShapeNet was rendered from 24 different azimuth angles.

To show the generalization of the proposed method on real images, we tested it using **Pix3D** dataset [23]. Pix3D is a publicly available dataset of aligned real-world image and 3D model pairs. It contains a large diversity in terms of object shapes and backgrounds and is highly challenging. We will test and report the performance of the proposed method on the chair, sofa, and table categories from the Pix3D dataset.

C. BASELINES

We test the proposed model trained on the ShapeNet dataset. First, we test the proposed model on synthetic images and show that the proposed model can generate point clouds that describe the object in the input image. Then, we validate the benefit of using the initial point cloud to improve the final point clouds. Also, we compare the proposed model against PSGN [8] and 3D-LMNet [16] qualitatively and quantitatively. CD (Eq. 3) and EMD (Eq. 4) are used to report the quantitative evaluation. Finally, we test the proposed model on real images to validate its generalizability on unseen images.

V. EXPERIMENTAL RESULTS & COMPARISONS

To test the performance of the proposed model, we evaluate it from different directions. First, we show general results generated by the proposed model on the ShapeNet dataset for different classes. Then, we compare the results of the proposed model against similar approaches that target the same problem using point cloud representation quantitatively and qualitatively.

After that, we validate the proposed architecture by an ablation study. We validate the benefits of using the initial point cloud (the 3D sphere) to generate more accurate results. Also, we show how the proposed model deals with the input images that have an ambiguous view (some object structures are hidden). Moreover, we show that the learned latent vector can be utilized to transfer useful information from one shape to another shape by applying arithmetic operations on different extracted features.

To check the model generality, we demonstrate the performance of the proposed model on the Pix3D dataset that has real images and compares the results against other methods quantitatively and qualitatively. Finally, we report some failure cases that happened in some results because of the strange shapes or some new parts that do not usually exist in normal cases.

A. GENERAL RESULTS ON ShapeNet DATASET

We test the proposed model on the testing set of ShapeNet. The proposed model was trained on synthetic images of objects rendered from different viewpoints. The testing was performed on 13 different categories. Figure 3 shows the qualitative results of 8 different categories. It clearly demonstrates that the generated point clouds of the objects from a single view are very close to the ground-truth and they capture the object geometry. Also, the proposed model learns to generate the point clouds and keeps the salient features such as free spaces between the splats in the back of the chair and the holes between the back and the seat of the bench. Moreover, the proposed model successfully learned to generate some thin and rare parts such as the stretchers between the chair legs as these parts are not common in the chair category. Many categories have various geometrical shapes such as the top surface of the tables. In Figure 3 (last row), the proposed model generates the circular surface accurately as the input image with the cylindrical pillar and the four small legs. Furthermore, the proposed model generates complete and plausible shapes. The generated points are evenly distributed and cover the whole parts of the objects.

B. COMPARISON RESULTS AGAINST OTHER METHODS

We benchmark our proposed model against PSGN and 3D-LMNet. Both models were trained on the same training set of ShapeNet. PSGN is the first model to solve the problem of single-view reconstruction using CNN that generates point clouds. In [8], the reported results show that the point cloud-based models outperform the state-of-the-art voxel-based models significantly. Table 1 reports the comparison results of our proposed model against PSGN [8] and 3D-LMNet [16] on ShapeNet dataset. It demonstrates that our proposed model outperforms PSGN in 8 out of 13 categories in the Chamfer metric and in all 13 categories in the EMD metric. Also, our proposed model outperforms 3D-LMNet in 6 out of 13 categories in the Chamfer metric and in all 13 categories in the EMD metric. Overall, the average performance of our proposed model outperforms both models in both metrics despite that our proposed model is simple, yet efficient, comparing with the others. Looking deeper into Table 1, EMD values denote better visualization of the generated point clouds of the objects. Also, since EMD is a point-to-point distance, it results in a high penalty when computing the distance between the points, and the two point cloud sets should have the same number of points. In Chamfer distance,

TABLE 1. Quantitative comparison of single-view reconstruction results on ShapeNet. The metrics are computed on 1024 points after performing ICP alignment with the ground truth point cloud. All metrics are scaled by 100.

Category	Chamfer			EMD		
	PSGN	3D-LMNet	Ours	PSGN	3D-LMNet	Ours
airplane	3.74	3.34	**3.29**	6.38	7.44	**3.82**
bench	4.63	**4.55**	4.59	5.88	4.99	**4.31**
cabinet	6.98	6.09	**6.07**	6.04	6.35	**4.94**
car	5.20	4.55	**4.39**	4.87	4.10	**3.61**
chair	**6.39**	6.41	6.48	9.63	8.02	**6.45**
lamp	**6.33**	7.10	6.58	16.17	15.8	**8.45**
monitor	**6.15**	6.40	6.39	7.59	7.13	**5.94**
rifle	2.91	**2.75**	2.89	8.48	6.08	**4.25**
sofa	6.98	**5.85**	**5.85**	7.42	5.65	**5.03**
speakers	8.75	**8.10**	8.39	8.70	9.15	**7.37**
table	**6.00**	6.05	6.26	8.40	7.82	**6.05**
telephone	4.56	4.63	**4.27**	5.07	5.43	**3.77**
vessel	4.38	**4.37**	4.55	6.18	5.68	**4.89**
Mean	5.62	5.40	**5.38**	7.75	7.00	**5.30**

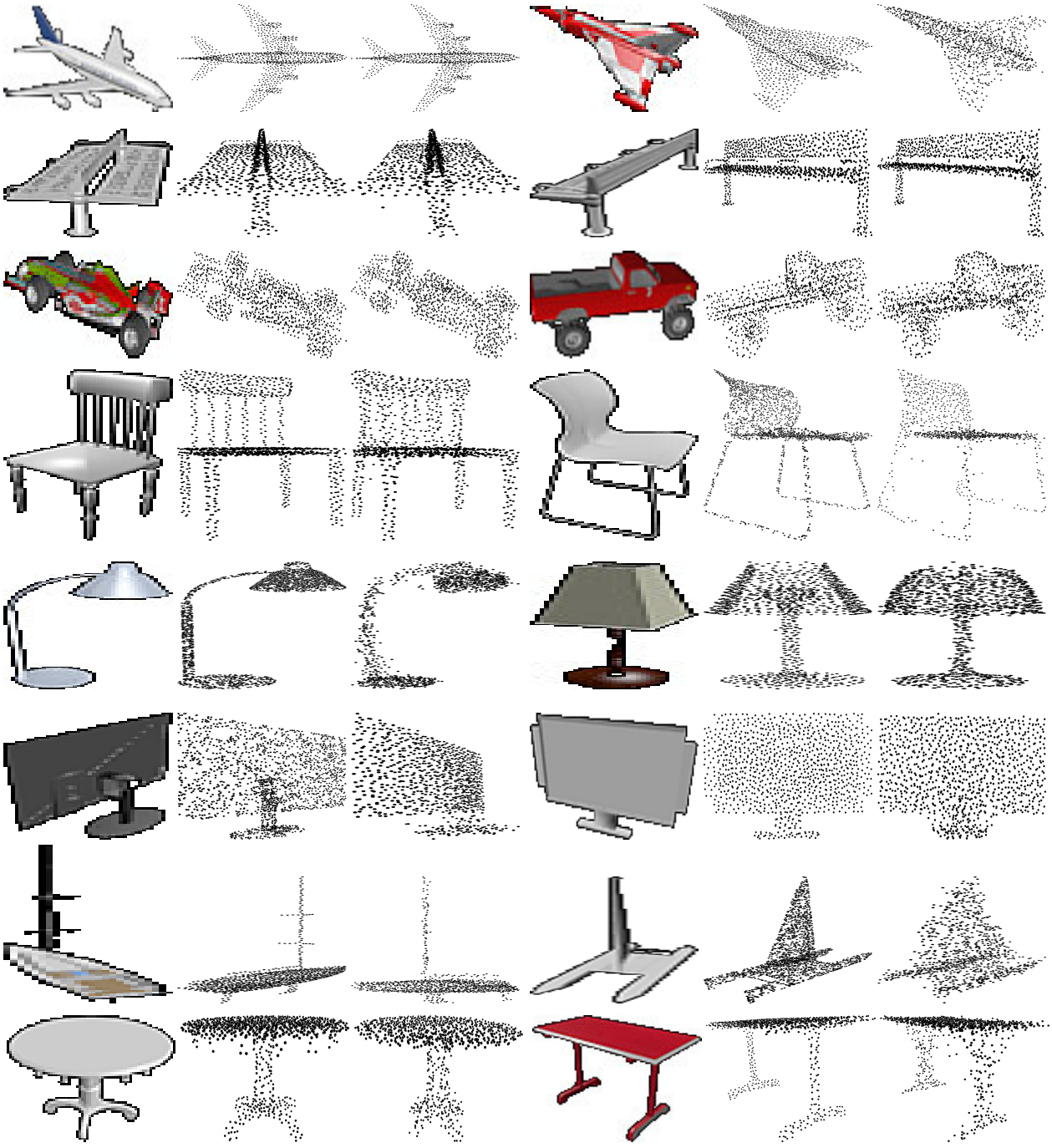


FIGURE 3. Qualitative results of ShapeNet on different categories. From left to right: input image, ground-truth, generated point cloud.

the nearest points are used to calculate the distance in a forward manner (from the generated point cloud to the ground-truth) and in a backward manner (from the ground-truth to the generated point cloud). It is not necessary that the generated point cloud and the corresponding ground-truth have the same number of points.

Figure 4 highlights the qualitative comparison. It clearly shows that the generated point clouds by our proposed model are visualized better than the ones generated by PSGN and 3D-LMNet. Our proposed model captures the details of the object and generates the object parts more accurately. In the riffle image (Figure 4, 4th row to the right), the small parts of the riffle are captured in more detail compared to the generated point cloud of 3D-LMNet as it doesn't generate the grip or the magazine and in PSGN where these parts are almost fused with each other and they cannot be separated. Also, our proposed model generates a well-distributed point cloud that the points are fairly distributed on the whole shape and not concentrated in one part or at the center of the shape. Moreover, this can be noticed in the chair image

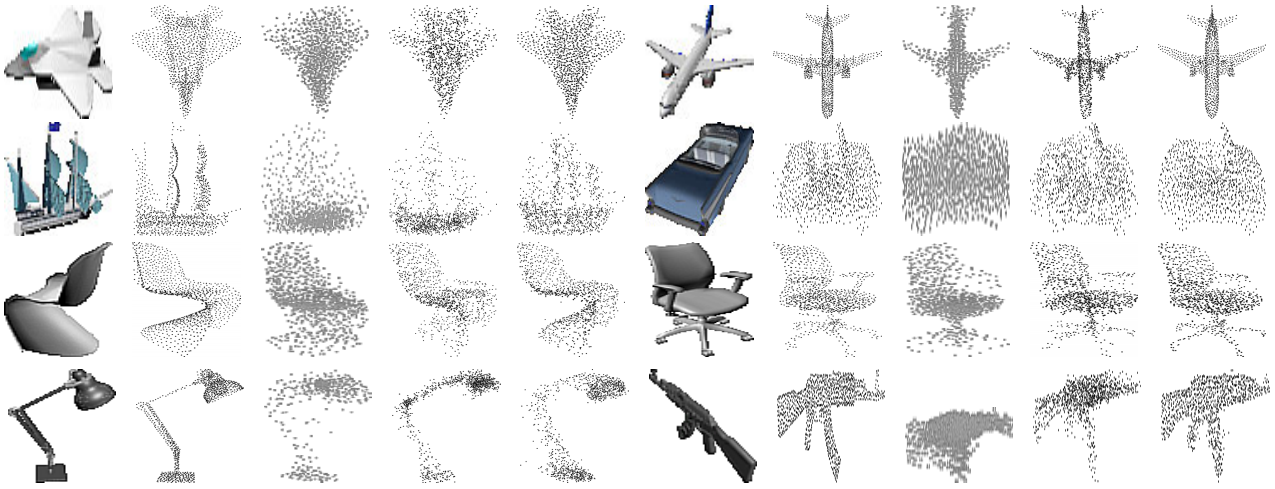


FIGURE 4. Comparison results between different methods on ShapeNet. From left to right: input image, ground-truth, resutls generated from PSGN, results generated from 3D-LMNet, and results generated from the proposed model.

(Figure 4, 3rd row) where our proposed model successfully generates and separates the chair legs and the armrest, but the other models have considered them either a fully connected part of the chair (*e.g.* the armrest) or one part (*e.g.* the chair legs). Thanks to the initial point cloud that helps in generating a well-distributed point cloud.

Moreover, we compare our proposed model against Pixel2Mesh [25]. Different from the proposed model, Pixel2Mesh uses an ellipsoidal mesh as an initial shape. It utilizes the extracted features from the image feature network from different stages and applies them to deform and add more details to the generated mesh in the mesh deformation network in a coarse-to-fine fashion. Table 2 reports the quantitative comparison between Pixel2Mesh model and our proposed model. With respect to CD, our model outperforms in some categories and is comparable to Pixel2Mesh results in other categories, and the average performance of our model outperforms Pixel2Mesh model. In EMD, our model outperforms Pixel2Mesh model in all categories and the average performance of the proposed model outperforms it with a large margin as reported in Table 2.

TABLE 2. Quantitative comparison between Pixel2Mesh [25] and ours on ShapeNet.

Category	Chamfer		EMD	
	Pixel2Mesh	Ours	Pixel2Mesh	Ours
airplane	4.77	**3.29**	5.79	**3.82**
bench	6.24	**4.59**	9.65	**4.31**
cabinet	**3.81**	6.07	25.63	**4.94**
car	**2.68**	4.39	12.97	**3.61**
chair	**6.10**	6.48	13.99	**6.45**
lamp	12.95	**6.58**	13.14	**8.45**
monitor	7.55	**6.39**	15.36	**5.94**
rifle	4.53	**2.89**	6.67	**4.25**
sofa	**4.90**	5.85	16.42	**5.03**
speakers	**7.39**	8.39	29.51	**7.37**
table	**4.98**	6.26	14.80	**6.05**
telephone	**4.21**	4.27	7.24	**3.77**
vessel	6.70	**4.55**	8.14	**4.89**
Mean	5.91	**5.38**	13.79	**5.30**

C. EFFECT OF THE INITIAL POINT CLOUD

To test the efficacy of using the initial point cloud in reconstructing a finer point cloud, we conduct an experiment to test and evaluate the performance of two different setups of the proposed model (Figure 1). The first model is the same as Figure 2 that uses an initial point cloud. The second setup has the same architecture as Figure 2 but without using the initial point cloud, and the point cloud is reconstructed directly from the input image. Both setups were trained on the training set of ShapeNet and were tested on the testing set of the same dataset.

Qualitatively, Figure 5 illustrates the results of the different setups of the proposed model. The point clouds generated by the proposed model without using an initial point cloud suffer from the uneven distribution of the points on the whole shape. Many points gather at some parts of the shape. In the chair example, many points are grouped at the back corners of the seats and fewer points are in the legs. However, the model with the initial point cloud produces chairs with well-distributed points and the chair legs are well reconstructed. Also, in the table examples, the point clouds generated without an initial point cloud have poor reconstructed legs, but they are well reconstructed using an initial point cloud during training. In the plane examples, the engines and the tail are not reconstructed and the points are concentrated on the body of the plane, but they are reconstructed accurately when using the initial point cloud. From Figure 5, we conclude that adding the initial point cloud to the proposed model improves the reconstructed point cloud, distributes the points evenly on the whole shape parts, and generates the object details accurately. Quantitatively, Table 3 reports a comparison between the different setups of the model. It is clearly noticed that the model with the initial point cloud outperforms the same model without using the initial point cloud with a large margin in both metrics.

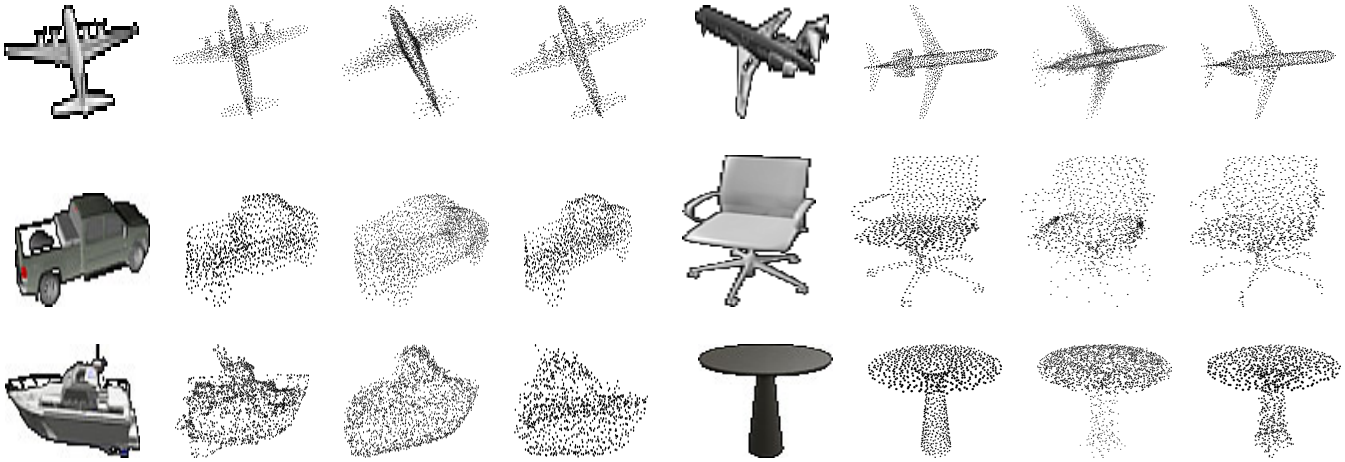


FIGURE 5. Qualitative results of the different setups of the proposed model on ShapeNet (Figure 1). From left to right: input image, ground-truth, results generated by the proposed model without the initial point cloud, and results generated by the proposed model with the initial point cloud.

TABLE 3. Quantitative comparison of different setups of the proposed model on ShapeNet.

Category	Chamfer		EMD	
	w/o PC	w PC	w/o PC	w PC
airplane	4.03	**3.29**	4.91	**3.82**
bench	**4.34**	4.59	10.20	**4.31**
cabinet	**5.97**	6.07	11.18	**4.94**
car	**4.21**	4.39	4.69	**3.61**
chair	7.00	**6.48**	7.30	**6.45**
lamp	**6.31**	6.58	32.08	**8.45**
monitor	6.62	**6.39**	19.83	**5.94**
rifle	**2.71**	2.89	11.06	**4.25**
sofa	6.49	**5.85**	6.24	**5.03**
speakers	**7.86**	8.39	20.61	**7.37**
table	6.47	**6.26**	7.00	**6.05**
telephone	**4.03**	4.27	6.36	**3.77**
vessel	5.64	**4.55**	6.58	**4.89**
Mean	5.52	**5.38**	13.7	**5.30**

D. GENERATING PLAUSIBLE SHAPES FROM AMBIGUOUS 2D INPUTS

To validate the performance of the proposed model, we conducted an experiment to test the model whether it can recognize and generate plausible shapes from 2D images of the chair class where the geometry of the objects is almost covered (the back-view of the chair). Figure 6 shows the qualitative results of this experiment. For each image, we show the back and the side views of the reconstructed model along with the ground-truth with the same viewpoint. It is clearly shown that the proposed model succeeded in guess the 3D geometry of the input image and generates plausible shapes that are consistent with the input images and the ground-truth. Also, the proposed model manages to memorize and reconstruct the chair parts such as the legs and the arms without seeing them in the 2D input images. Figure 6 proves that the proposed model can generate plausible shapes that are consistent with the ambiguous 2D images and are close enough to the ground-truth.

E. ARITHMETIC OPERATIONS ON THE 2D INPUT IMAGE FEATURE VECTOR

Another interesting experiment is to check if the extracted 2D features from the input images have meaningful information or not. To do so, we extract the 2D features from different 2D images of the same category and apply arithmetic operations on them to generate a new 3D shape. In [17], it was shown that vector(King)−vector(Man)+vector(Woman) gives a vector that the nearest neighbor to it was a vector for Queen. The experiment performs similar to this idea. We select random triples, extract their 2D features using the encoder network, and apply the arithmetic operations ($fv1 - fv2 + fv3$). The resulting feature is then passed to the generator to generate the 3D point cloud.

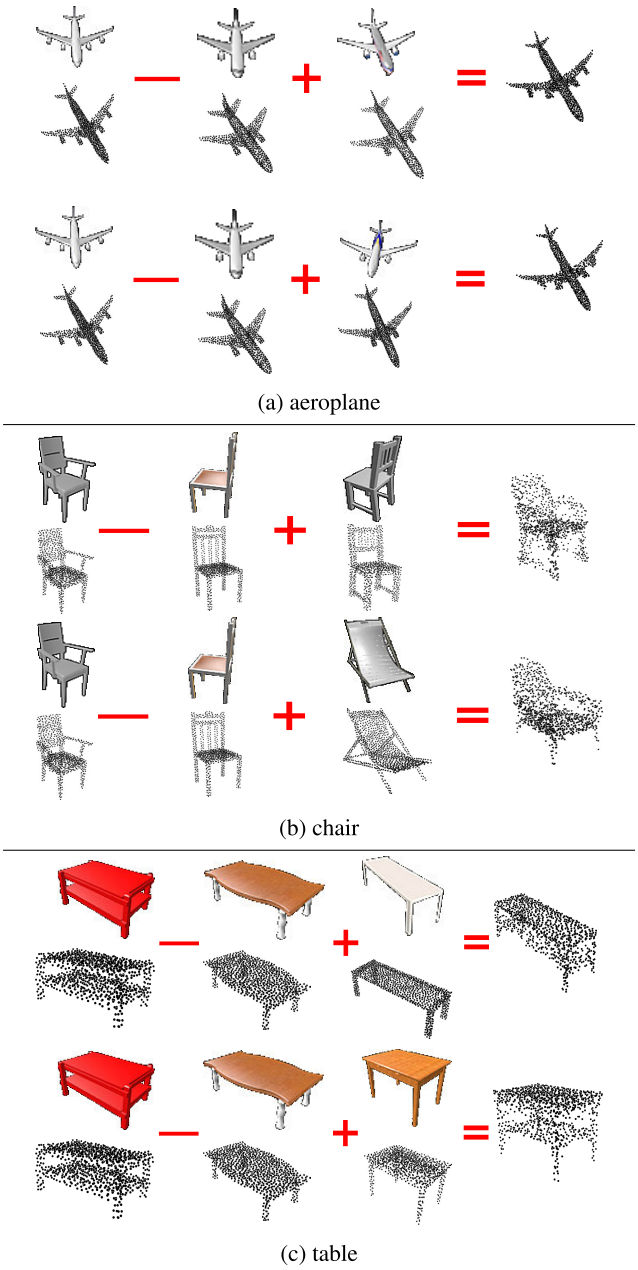
Figure 7 shows the results of applying the arithmetic operations of some categories. The first experiment was applied to the airplane category. In Figure 7a, the first image is an airplane with two engines on each side and the second image is an airplane with one engine on each side. We subtract the extracted features of both images and then add the difference to the third image of an airplane that has just one engine on each side. As shown in Figure 7a, the generated new shape is an airplane that has two engines on each side. This means that the difference between the first two images generates a feature of an engine and then adds it to the third image results in a new airplane with two engines.

The second example was applied to the chair category. The main image is for a chair with arms. The other images are chairs without arms. We want to test if we can subtract the arms from the first shape and add them to the new shape. Figure 7b shows that when subtracting the feature of a chair that doesn't have arms from a chair that has arms and then adds the new feature to a third one we get the same shape of the third chair but with arms. This means that the difference between the two features generates a feature that has the chair arms information. And when adding this feature to a new

FIGURE 6. Qualitative results of 3D reconstruction for the ambiguous 2D inputs. From left to right: 2D input image, grount-truth view-1, generated output view-1, ground-truth view-2, generated output view-2.

image generates a shape that is similar to the input image that contains the transferred arms.

A third example was applied to the table category. The first image is for a table with a bottom shelf and the second image is for a table without the bottom shelf. When we subtract the feature of the second image from the feature of the first image and add the result to the third feature of a new image results in a table with the bottom shelf. The generated table



(a) aeroplane

(b) chair

(c) table

FIGURE 7. Results of applying arithmetic operations on 2D features extracted by the encoder for different shapes.

is similar to the third image plus the bottom shelf. As can be seen in Figure 7c, the generated tables are similar to the third images where, for example, the table with long legs preserves its geometry after adding the new feature.

As shown in Figure 7, the proposed model extracts meaningful features that contain meaningful information. These features can be used to generate real shapes that have extra parts.

F. Pix3D DATASET RESULTS

The proposed model was trained on synthetic images that are clean and the objects appear well in the images. To test the performance of the model in real scenarios, the Pix3D dataset is used. This dataset contains a large collection of real images and the corresponding metadata such as masks along with ground-truth 3D CAD models of different object categories. The shared categories between ShepNet and Pix3D datasets are used to test and evaluate the proposed method. The testing images are preprocessed. The images are cropped to center-position the object of interest in the image and masked the background using the corresponding mask. Then the image is resized to match the training image size (128×128). The proposed model isn't fine-tuned on the Pix3D dataset, but it is directly tested on the dataset images.

Table 4 reports the quantitative results of comparing the proposed model against PSGN and 3D-LMNet on Pix3D images. The three models were trained on ShapeNet and tested on Pix3D. The reported performance of PSGN and 3D-LMNet are taken from [16]. Table 4 shows that the proposed model outperforms the other models by a large margin in both metrics and on all object categories. This demonstrates the efficiency of the proposed model on real data.

TABLE 4. Single-view reconstruction results on the real-world Pix3D dataset. All metrics are scaled by 100.

Category	Chamfer			EMD		
	PSGN	3D-LMNet	Ours	PSGN	3D-LMNet	Ours
chair	8.05	7.35	**6.82**	12.55	9.14	**7.45**
sofa	8.45	8.18	**3.95**	9.16	7.22	**3.28**
table	10.82	11.20	**5.22**	15.16	12.73	**5.17**
Mean	9.11	8.91	**5.33**	12.29	9.70	**5.30**

Figure 8 visualizes the reconstruction results of some selected Pix3D images generated from the proposed model along with 3D-LMNet. 3D-LMNet performs well on real-world images, but our model performs better and the generated point clouds are more accurate and very similar to the ground-truth. Our model distributes the points evenly on the whole object shape and covers the object parts accurately. This shows that the proposed model generalizes well to the real-world images and generates accurate models that describe the input images even though the images are from a different distribution than the training set.

G. FAILURE CASES

The proposed model fails to generate very accurate shapes in some cases. Figure 9 shows some failure cases. Most thinner and narrower parts of the objects are missed such as the chair armrests and the airplane tail. Also, the objects with extra parts that don't usually exist are also missed such as a monitor with two bases or a table with three legs on each side. Normally, the narrow and extra parts are missed because the network didn't learn to predict them. However, if this happens in one example, the network tries to generate and estimate the closest shape to the input image as the table with the six legs in Figure 9 (the last row). The proposed model reconstructs and generates a plausible point cloud that is close to the input image but it misses the leg in the middle.

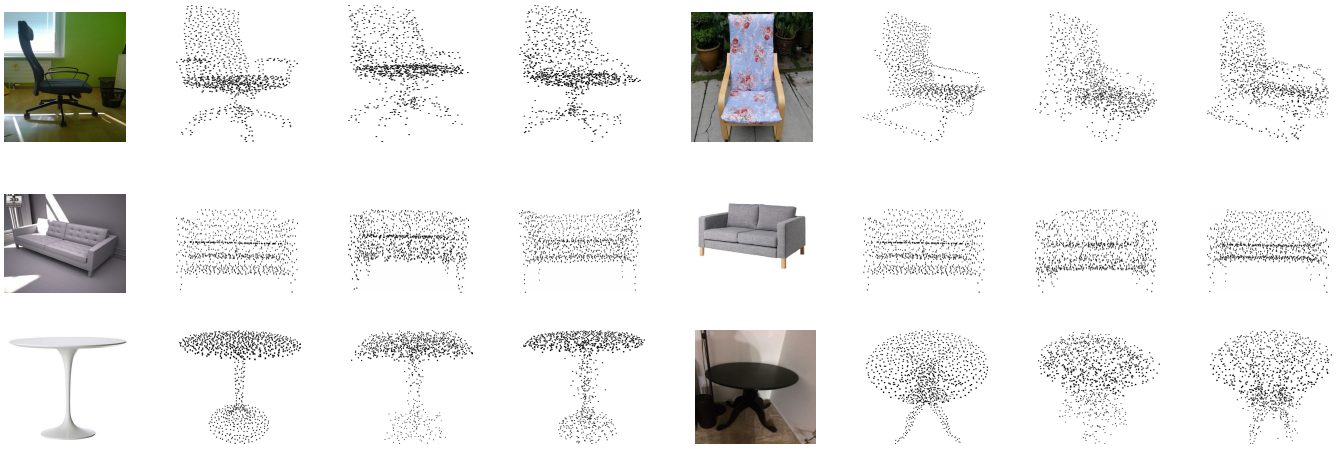


FIGURE 8. Qualitative results on chair, sofa, and table categories from Pix3D dataset. From left to right: input image, ground-truth, results generated from 3D-LMNet, and results generated from the proposed model.

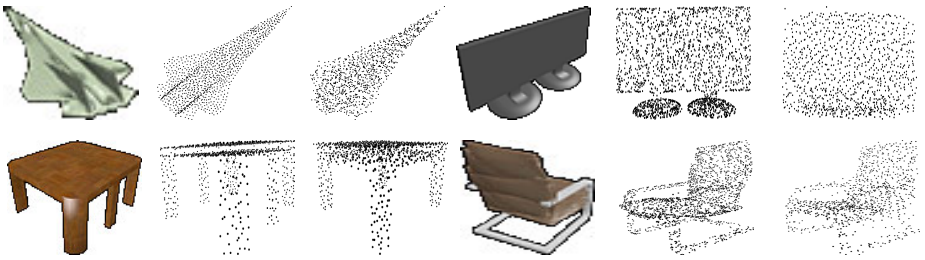


FIGURE 9. Failure cases of our method on ShapeNet. Failures happen because of extra unexpected or thin and narrow parts. From left to right: input image, ground-truth, generated point cloud.

VI. DISCUSSION & CONCLUSION

Though single-view 3D object reconstruction is a challenging task, the well-created human eyes have the ability to infer and predict the geometry of a scene and the objects within it from a single image. With more complicated scenarios such as high occlusion of the objects, the human brain is able to guess a number of plausible shapes that could match what is seen. This is because of the prior information that is stored in the human brain and is retrieved, utilized, and updated when seeing new scenes. Recently, different research fields have exploited the ability to reconstruct objects from a single image in many applications such as the field of robotics in object grasping and manipulation. However, it is an ill-posed problem and many plausible reconstructions could be a solution for one single view due to the uncertainty.

In this paper, we have proposed a simple, yet powerful, CNN model to generate the point clouds of an object from a single image. 3D data can be represented in different ways. Point clouds have been proven to be a common and simple representation. The proposed model trained end-to-end on synthetic data with 3D supervision. It takes a single image of an object and generates a point cloud with a fixed number of points ($N = 2048$). Qualitative and quantitative evaluations on synthetic and real data demonstrate that the proposed model is able to generate point clouds that are very close to the ground-truth and more accurate in comparison with other methods. Moreover, we show that the initial point cloud has improved the final results as it distributes the points on the whole object shape evenly. The qualitative results show that the points are grouped in some object parts densely while other parts have fewer points when the proposed model doesn't use the initial point cloud. Furthermore, the performance of the proposed model on the real-world dataset illustrates the outstanding generalization to the new and unseen images and scenes.

REFERENCES

[1] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, and M. Kudlur, "Tensorflow: A system for large-scale machine learning," in *Proc. 12th USENIX Symp. Operating Syst. Design Implement.*, 2016, pp. 265–283.

[2] J. T. Barron and J. Malik, "Shape, illumination, and reflectance from shading," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 8, pp. 1670–1687, Aug. 2015.

[3] G. Bresson, Z. Alsayed, L. Yu, and S. Glaser, "Simultaneous localization and mapping: A survey of current trends in autonomous driving," *IEEE Trans. Intell. Vehicles*, vol. 2, no. 3, pp. 194–220, Sep. 2017.

[4] M. M. Bronstein, J. Bruna, Y. LeCun, A. Szlam, and P. Vandergheynst, "Geometric deep learning: Going beyond Euclidean data," *IEEE Signal Process. Mag.*, vol. 34, no. 4, pp. 18–42, Jul. 2017.

[5] A. X. Chang, T. Funkhouser, L. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su, J. Xiao, L. Yi, and F. Yu, "ShapeNet: An information-rich 3D model repository," 2015, *arXiv:1512.03012*. [Online]. Available: http://arxiv.org/abs/1512.03012

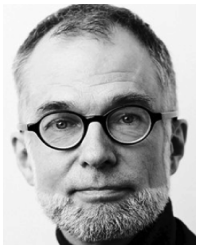
[6] C. B. Choy, D. Xu, J. Gwak, K. Chen, and S. Savarese, "3D-R2N2: A unified approach for single and multi-view 3D object reconstruction," in *Proc. Eur. Conf. Comput. Vis.* Springer, 2016, pp. 628–644.

[7] A. Criminisi, I. Reid, and A. Zisserman, "Single view metrology," *Int. J. Comput. Vis.*, vol. 40, no. 2, pp. 123–148, 2000.
[8] H. Fan, H. Su, and L. Guibas, "A point set generation network for 3D object reconstruction from a single image," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 605–613.
[9] 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.
[10] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge, U.K.: Cambridge Univ. Press, 2003.
[11] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*. [Online]. Available: http://arxiv.org/abs/1412.6980
[12] 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.
[13] C.-H. Lin, C. Kong, and S. Lucey, "Learning efficient point cloud generation for dense 3D object reconstruction," in *Proc. 32nd AAAI Conf. Artif. Intell.*, 2018, pp. 1–8.
[14] Q. Lu, Y. Lu, M. Xiao, X. Yuan, and W. Jia, "3D-FHNet: Three-dimensional fusion hierarchical reconstruction method for any number of views," *IEEE Access*, vol. 7, pp. 172902–172912, 2019.
[15] Q. Lu, M. Xiao, Y. Lu, X. Yuan, and Y. Yu, "Attention-based dense point cloud reconstruction from a single image," *IEEE Access*, vol. 7, pp. 137420–137431, 2019.
[16] P. Mandikal, K. L. NavaneetK, 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. BMVC*, 2018.
[17] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 3111–3119.
[18] M. R. Oswald, E. Töppe, K. Kolev, and D. Cremers, "Non-parametric single view reconstruction of curved objects using convex optimization," in *Proc. Joint Pattern Recognit. Symp.* Springer, 2009, pp. 171–180.
[19] M. R. Oswald, E. Töppe, C. Nieuwenhuis, and D. Cremers, "A review of geometry recovery from a single image focusing on curved object reconstruction," in *Innovations for Shape Analysis*. Springer, 2013, pp. 343–378.
[20] M. Prasad and A. Fitzgibbon, "Single view reconstruction of curved surfaces," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2, Dec. 2006, pp. 1345–1354.
[21] Y. Rubner, C. Tomasi, and L. J. Guibas, "The Earth mover's distance as a metric for image retrieval," *Int. J. Comput. Vis.*, vol. 40, no. 2, pp. 99–121, Nov. 2000.
[22] A. Sinha, A. Unmesh, Q. Huang, and K. Ramani, "SurfNet: Generating 3D shape surfaces using deep residual networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6040–6049.
[23] X. Sun, J. Wu, X. Zhang, Z. Zhang, C. Zhang, T. Xue, J. B. Tenenbaum, and W. T. Freeman, "Pix3D: Dataset and methods for single-image 3D shape modeling," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2974–2983.
[24] M.-P. Tran, "3D contour closing: A local operator based on Chamfer distance transformation," Version v2, Tech. Rep., Mar. 2013. [Online]. Available: https://hal.archives-ouvertes.fr/hal-00802068
[25] 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. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 52–67.
[26] J. Wu, C. Zhang, T. Xue, B. Freeman, and 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.
[27] Y. Xiang, W. Choi, Y. Lin, and S. Savarese, "Data-driven 3D voxel patterns for object category recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1903–1911.
[28] L. Zhang, G. Dugas-Phocion, J.-S. Samson, and S. M. Seitz, "Single-view modelling of free-form scenes," *J. Visualizat. Comput. Animation*, vol. 13, no. 4, pp. 225–235, 2002.
[29] Y. Zhang, Z. Liu, T. Liu, B. Peng, and X. Li, "RealPoint3D: An efficient generation network for 3D object reconstruction from a single image," *IEEE Access*, vol. 7, pp. 57539–57549, 2019.

AHMED J. AFIFI was born in 1985. He received the bachelor's and M.Sc. degrees in computer engineering from the Islamic University of Gaza, in 2008 and 2011, respectively. He is currently pursuing the Ph.D. degree with the Computer Vision and Remote Sensing Research Group, Technische Universität Berlin. During his master's degree, he was interested in digital image processing and pattern recognition. His research interests include computer vision, deep learning, 3D object reconstruction from a single image, and medical image analysis.

JANNES MAGNUSSON was born in Berlin, Germany, in 1994. He received the degree in computer science from Technische Universität Berlin, specialized in computer vision and medical image processing, and the master's degree in 2020. He is currently working as a Research Associate with the Institute for Image Science and Computational Modeling in Cardiovascular Medicine, Charité–Universitätsmedizin Berlin.

TOUFIQUE A. SOOMRO (Member, IEEE) received the B.E. degree in electronic engineering from the Mehran University of Engineering and Technology, Pakistan, in 2008, the M.Sc. degree in electrical and electronic engineering by research from University Technologi PETRONAS, Malaysia, in 2014, and the Ph.D. degree in AI and image processing from the School of Computing and Mathematics, Charles Sturt University, Bathurst, NSW, Australia, in 2018. He remained Research Assistant for six months at the School of Business Analytic in Cluster of Big Data Analysis, The University of Sydney, Sydney, NSW, Australia. He is currently an Assistant Professor with the Department of Electronic Engineering, QUEST, Larkana, Pakistan. His research interests include most aspects of image enhancement methods, segmentation methods, classifications methods, and image analysis for medical images.



OLAF HELLWICH (Senior Member, IEEE) was born in 1962. He received the B.S. degree in surveying engineering from the University of New Brunswick, Fredericton, NB, Canada, in 1986, and the Ph.D. degree in linienextraktion aus SAR-Daten mit einem Markoff-Zufallsfeld-Modell from the Technische Universität München, München, Germany, in 1997. He headed the Remote Sensing Group, Department of Photogrammetry and Remote Sensing, Technische Universität München. Since 2001, he has been a Professor with the Technische Universität Berlin (TUB), Berlin, Germany, initially for photogrammetry and cartography, and since 2004 for Computer Vision and Remote Sensing. From 2006 to 2009, he was the Dean of the Faculty of Electrical Engineering and Computer Science, TUB. His research interests include 3-D object reconstruction, object recognition, synthetic aperture radar remote sensing, and discovery and use of object shape priors in 3-D reconstruction. He was a recipient of the Hansa Luftbild Prize of the German Society for Photogrammetry and Remote Sensing, in 2000.