RGB2Point: 3D Point Cloud Generation from Single RGB Images

Jae Joong Lee[1] and Bedrich Benes[1]

[1]Department of Computer Science, Purdue University, West Lafayette, USA
{lee2161, bbenes}@purdue.edu
https://www.jaejoonglee.com/wacv25_rgb2point

Abstract

We introduce RGB2Point, an unposed single-view RGB image to a 3D point cloud generation based on Transformer. RGB2Point takes an input image of an object and generates a dense 3D point cloud. Contrary to prior works based on CNN layers and diffusion-denoising approaches, we use pre-trained Transformer layers that are fast and generate high-quality point clouds with consistent quality over available categories. Our generated point clouds demonstrate high quality on a real-world dataset, as evidenced by improved Chamfer distance (51.15%) and Earth Mover's distance (36.17%) metrics compared to the current state-of-the-art. Additionally, our approach shows a better quality on a synthetic dataset, achieving better Chamfer distance (39.26%), Earth Mover's distance (26.95%), and F-score (47.16%). Moreover, our method produces 63.1% more consistent high-quality results across various object categories compared to prior works. Furthermore, RGB2Point is computationally efficient, requiring only 2.3GB of VRAM to reconstruct a 3D point cloud from a single RGB image, and our implementation generates the results 15,133× faster than a SOTA diffusion-based model.

1. Introduction

Generation of 3D point clouds from a single image is an open problem in Computer Vision, and the main challenge is handling occlusions given the limited viewpoint. The emergence of Deep Learning has alleviated this concern by leveraging 2D image features extracted from well-trained models [14, 44, 46] on extensive image datasets [8,21,24]. Recent works used the pre-trained models [14, 44, 46] as image feature extractors to reconstruct 3D objects in works [7, 53, 58, 59]. However, the introduction of the attention [52] mechanism and its usage in the Vision Transformer (ViT) model [10] has shown remarkable performance improvements in image classification tasks, particularly on the ImageNet [8]. ViT's unique architecture effectively captures global information through its self-attention mechanism, thus outperforming Convolutional Neural Networks (CNNs).

Many 3D object representations exist, and unstructured point clouds are primarily provided during the data acquisition tasks as they are provided by sensors, such as LiDAR [38], or by photogrammetric algorithms, such as the Structure from Motion [2, 41]. However, they have also been used as intermediate representations in many tasks [7, 36, 65].

The denoising diffusion probabilistic model showed excellent results on 2D image synthesis [15, 40] and 3D objects [25, 37]. Many diffusion-based models require extensive hardware resources due to their large size and the numerous iterations needed to transform a Gaussian distribution into a complex one during training. This demand for resources renders some models inaccessible because of their scale and the terabytes of data required for training on internet-scale image datasets. It is widely acknowledged that gathering large volumes of data is crucial for developing stable networks. However, not all data is easy to collect, and point cloud data, particularly, have significant challenges. Collecting point cloud data requires a sensor to scan entire objects and ensure complete area coverage to avoid data gaps. This process becomes even more complicated when the target object is large or inaccessible, thus significantly increasing the complexity of data collection. Instead of relying on hardware sensors, point clouds can be obtained using photogrammetry, such as COLMAP [42,43], which is grounded in Structure-from-Motion and Multi-View Stereo. Although this software-based approach offers a viable alternative, it is important to note that the quality of the generated point clouds may not be on par with those obtained from dedicated hardware sensors [35].

These challenges motivated us to develop a model that can reconstruct a point cloud from a single image while be-

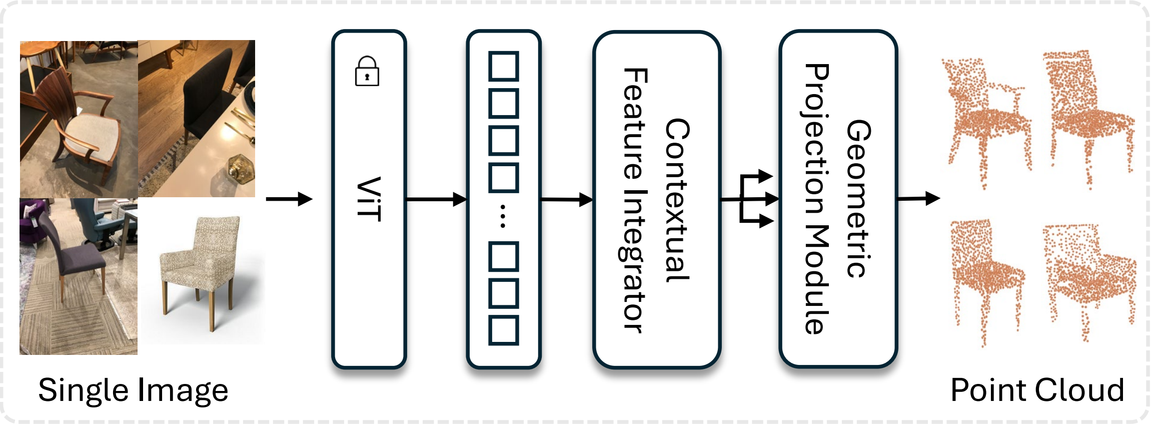

Figure 1. **Model Architecture.** *RGB2Point* takes a single view RGB image and extracts image features from the pre-trained ViT [10]. The Contextual Feature Integrator then refines these extracted features, which applies a multi-head attention mechanism [51] to highlight specific regions of interest within the features. The weighted features are forwarded to the Geometric Projection Module, which maps them into a 3D space, resulting in a point cloud representation. We carefully designed the model, *RGB2Point* which requires only 2.3GB of VRAM to generate a 3D point cloud from a single RGB image.

ing executable on widely available GPUs. Our *RGB2Point* is an accessible solution that addresses the limitations inherent in existing diffusion-based models. This ensures that the broader research community can effectively utilize our methodology, facilitating wider adoption and application.

We introduce a novel approach to reconstructing images into point clouds using the ViT. Our *RGB2Point* is a Deep Learning network for 3D point cloud reconstruction from a single image. As an image feature extractor, we employ the pre-trained Vision Transformer [10] on ImageNet [8]. Our network incorporates a Multi-head Attention (MHA) layer along with a Multi-Layer Perceptron (MLP) to generate a 3D point cloud, as illustrated in Fig. 1. Unlike heavily nested layers of models, our simple but powerful model provides high-quality but stable reconstructions over the categories. This simple architecture is cheap to train on a single desktop-level GPU and requires only 2GB of VRAM to reconstruct a 3D point cloud from a single RGB image. Besides the low memory requirements, it boosts speed as it takes 0.15 seconds per single RGB image to reconstruct a 3D point cloud.

We compare *RGB2Point* to previous approaches [18, 32–34, 59] and demonstrate its performance through evaluation on unseen objects from two datasets: ShapeNet [6], a synthetic dataset, and Pix3D [45], a manually 3D scanned dataset from the real-world.

We analyze the impact of pre-trained weights on the effectiveness of the ViT for 3D point cloud reconstruction tasks. Moreover, we swap the image feature extractor part to a pre-trained ResNet50 [14] to evaluate the effect of ViT. Our method shows reconstruction improvements, including quantitative and qualitative results over existing works, as *RGB2Point* outperforms in Chamfer distance (*39.26%* and *51.15%*) and Earth Mover's distance (*26.95%* and *36.17%*) compared to the current state-of-the-art. Also, our model shows 47.1% higher reconstruction quality in F-score compared to the diffusion-based method [32]. Contrary to previous work, our reconstructed point clouds show a 63.1% more consistently high quality for all object classes. **We claim the following contributions:**

- We propose a new model architecture that is VRAM efficient but also generates a high-quality point cloud from a single RGB image.
- We show Transformer model can generate higher quality 3D objects than a probabilistic denoising diffusion model.

2. Related Work

CNN based extractors such as pre-trained models [14, 44] are widely used to reconstruct 3D objects from 2D images such as an occupancy voxel [53, 58], mesh [47, 55], or 3D point cloud [7, 34]. Using extracted CNN-based image features, a network learns camera poses with point cloud data to use a differentiable renderer [18]. Our work does not need camera poses for point cloud generation to reduce extra efforts to maintain camera parameters for every image. Several algorithms reconstruct 3D point clouds using layers of CNN as encoder and decoder [11] and Recurrent Neural Network [7]. An extra loss function is introduced by calculating the similarity of the reconstructed point cloud of random viewpoints [34]. A differentiable render is used for the 3D point cloud reconstruction [33] to capture view consistent 3D objects.

Transformer uses the attention [52] mechanism and it has shown its outstanding performance in Natural Language

Processing (NLP) such as question answering [9, 62, 64], text generation [3, 49] and sentiment analysis [26, 63]. Transformer [52] has also shown superior performances in the Computer Vision area as it outperforms image classification tasks [10, 17], object detection [4, 12], semantic segmentation [27, 57]. Moreover, the performance of Transformer [52] continues in 3D space as well in different domains such as 3D point cloud completion task [65, 68], 3D reconstruction [54] and 3D tree generation [23]. We utilize the Vision Transformer as an image feature extractor of our model to reconstruct 3D point cloud data. Our approach exhibits improvements across various metrics, including *39.26%* and *51.15%* for Chamfer Distance, as well as *26.95%* and *36.17%* for Earth Mover's distance, evaluated on synthetic object datasets [6] and real object datasets [45]. In contrast to the existing methods [18, 33, 34] that rely on CNN-based feature extractors, our model demonstrates a better performance in 3D generation metrics as we shown in Tab. 2 from the synthetic dataset [6] and Tab. 3 from the real-world dataset [45].

Diffusion models are new proposing approaches for a generative deep learning model by iterative denoising process. Using denoising diffusion probabilistic models (DDPM) [15] or from a latent space [40]. By utilizing these two fundamental models, they contribute numerous applications such as a novel view synthesis [5, 69] and a 3D object generation [16, 22, 25, 37, 48].

In this emergence of diffusion-based models, researchers leverage its approach to point cloud such as an unconditional diffusion model to generate point clouds [29, 35, 50], point cloud completion using a diffusion model [30] and a single image to point cloud reconstruction [32]. The diffusion methods leverage probabilistic approaches that show a high quality of the unseen field. When the number of reconstructing categories is diverse, one of the critical elements of the model is stability over the categories. From Tab. 3, we infer the stability of each model by calculating a standard deviation of F-scores over the reconstructed categories, and the lower value gives a more stable generation quality. Also, we show *15,133* times faster generation along with better qualitative results by comparing a diffusion-based work [35] in Fig. 3.

Furthermore, our paper shows a Transformer-based model can also generate high quality of the unknown area like diffusion, and we show *47.1%* higher F-score than the diffusion-based method [32] as we show in Tab. 1 and visual comparisons in Fig. 3.

3. Approach

Overview: *RGB2Point* consists of three parts (Fig. 1): 1) 2D image feature extraction using a pre-trained Transformer, 2) Contextual Feature Integrator, and 3) Geometric Projection Module. The demonstrated strength of the transformers in vision tasks [10, 17, 28] motivates us to integrate it for the generation of point clouds.

The primary contribution of our work is more efficient (requires 2.3GB VRAM), faster (15,133 times faster than a diffusion-based model [35]), higher-quality reconstructions than prior works [18, 32, 34, 35]. A prior work [35] uses several millions of 3D objects. *RGB2Point* requires only less than 10% of the training dataset, and it gives a better reconstruction quality on complex real-world data as shown in Fig. 3.

Architecture: *RGB2Point* takes a 224 × 224 RGB image as its input and generates its corresponding 3D point cloud with N points. From the evaluation, Sect. 4.4, we show that our model is flexible on the number of output point clouds, such as 256, 1024, or 8192 points. We do not need any additional set of layers but simply change the size of the output layer during the training. *RGB2Point* consists of three parts. The first part is composed of a pre-trained vision Transformer [10] that extracts image features.

Contextual Feature Integrator (CFI) is a designed module to enhance the representation of specific regions within an image, which is important for generating accurate outputs. It consists of a feed-forward layer with size A and a multi-head attention mechanism with H heads. The feed-forward layer acts as a transformation network, mapping the input features into a higher-dimensional space that captures relationships among the features. The dimensionality, A, allows the network to model complex dependencies that might not be apparent in the original feature space. By applying non-linear activation functions, this layer introduces non-linearities into the model to learn complex patterns. The multi-head attention mechanism in the model is composed of H heads, each responsible for attending to different aspects of the input feature sequences. This method enables the model to capture a wide range of contextual information within the input sequence, allowing it to attend to multiple relevant features concurrently. Each attention head computes a weighted sum of the input features, where the weights are derived from the similarity between the feature vectors, which serve as both queries, keys, and values in a self-attention setup. The outputs from all attention heads are concatenated and passed through a linear layer, which projects them into a dimensional space appropriate for point cloud generation. This step ensures that the model effectively integrates and highlights important features, resulting in a more precise representation in the generated point cloud.

Geometric Projection Module (GPM) plays a key component in bridging the transition from high-dimensional feature representations to accurate 3D point cloud generation. This module maintains the spatial coherence and structural information of the input features to perform the effective translation of abstract feature spaces into meaningful geometric representations. Following the multi-head attention mechanism, this module employs a sequence of linear layers with Leaky ReLU activations [31] with a slope value of 0.2 to introduce necessary non-linearities and capture complex feature interactions. The module starts with the initial linear transformation, which projects the attention-weighted features into a higher-dimensional space. This step improves the model's capacity to capture complex spatial relationships and dependencies within the input data. Subsequent linear layers then refine these features, progressively distilling the essential information needed for accurate 3D reconstruction. The final layer of the module performs a critical projection, mapping the refined features into a three-dimensional coordinate space. This ensures that each output point corresponds to a precise x, y, and z coordinate in the generated point cloud.

We set $A = 1,024$, $H = 4$, and $D = 2,048$. We choose these values by conducting extensive experiments, as shown in Tab. 4.

4. Implementation, Datasets, and Experiments

4.1. Implementation

We train and test our model with Python3.9, Pytorch 2.1.0, Nvidia CUDA 12.1, a single NVIDIA RTX 4090 GPU with 24GB VRAM, and an Intel i9-13900KF.

4.2. Datasets

Since collecting RGB images with point cloud data is hard, we carefully selected our training dataset from existing data. ShapeNet [6] provides a wide selection of common object categories, and since it is a synthetic dataset, we can leverage its 3D information to train our model. Working well on a synthetic dataset does not give much information for an application aspect; we evaluate our model on a complex real-world dataset, Pix2D [45], where point cloud data are scanned from the real world. We use two 3D datasets: a synthetic object dataset, ShapeNet [6], and real-world objects from Pix3D [45]. We train *RGB2Point* on all available categories from ShapeNet [6]. We use the same train and test splits as the previous work [34] that 3D-R2N2 [7] proposed. We evaluate the robustness and generalization of our model by testing it against the real-world dataset, Pix3D [45], *without training our model on this dataset*.

4.3. Training

RGB2Point takes a 224 × 224 single RGB image from ShapeNet [6]. We use the ground truth point clouds from the same datasets in resolution 1024 points, denoted by G. For F-score calculation (Tab. 1), we simply increase the size from 1,024 to 8,192 to make fair comparisons with other works. We train our model by setting $(H = 4, D = 2048, A = 1024)$, from Fig. 1 with a 3D point cloud reconstruction loss L_{cd}, that calculates Chamfer distance between the ground truth point cloud data G and the generated point cloud data R where $G = \{x_i \in \mathbb{R}^3\}_{i=1}^n, R = \{x_j \in \mathbb{R}^3\}_{j=1}^n, N(x, P) = \arg\min_{y \in P} \|x - y\|$.

The model is optimized using Adam [20] with a learning rate set to 10^{-4} with its default parameters, using a batch size 32. During the training, we freeze the ViT [10] while minimizing the Chamfer distance loss L_{cd}, where $\alpha = 5$, *n=the number of point cloud size* and θ represents the trainable model parameters:

$$L_{cd} = \frac{1}{2n}\sum_{i=1}^{n}|x_i - N(x,R)| + \frac{1}{2n}\sum_{j=1}^{n}|x_i - N(x,G)| \quad (1)$$

$$\min_{\theta} \alpha L_{cd}\left(G, R, M(\theta)\right). \quad (2)$$

4.4. Evaluation

We validated *RGB2Point* using two datasets: a synthetic dataset ShapeNet [6] and real-world images from Pix3D [45]. We visualize the generated results of a single RGB image from ShapeNet [6] in Fig. 2 for a qualitative evaluation. However, the qualitative results with a side-by-side comparison with some other works were not possible because of the outdated software (Python 2 version) and hardware support (CUDA 8 version) that our workstation cannot run. We contacted authors to obtain pre-trained models but could not get them from them. However, the authors provide their datasets so we can evaluate them quantitatively. Also, we compare qualitative results between a recent single image-based 3D reconstruction works [16, 35, 48, 50] in complex real-world data in Fig. 3. In the test stage, *RGB2Point* demonstrates efficiency, utilizing merely 2.34 GB of VRAM when operating with a batch size of one. Furthermore, a diffusion-based model [35] takes 37 minutes and 50 seconds per image on average, but *RGB2Point* only takes 0.15 seconds per image on average. In other words, our model is 15,133 faster but generates a higher-quality point cloud in a complex real-world single image, as we show in Fig. 3.

We employ three quantitative assessment metrics to measure the similarity between the generated result and its target: Chamfer distance↓, Earth Mover's distance↓, and F-score↑. The first two metrics indicate that a lower value signifies better generation quality, while the last metric shows that a higher value represents better generation quality.

First, we compare our generations to a Diffusion-based approach [32] and prior works that generate a 3D occupancy grid, voxel, from a single-view image [7, 59, 61]. All the works are evaluated using 8192 points, and we replace the output layer size with 8,192 instead of 1,024. Based on the F-score that we show in Tab. 1, our model shows *47.2%* better than the current state-of-the-art model [32] that based on a probabilistic diffusion model which is recently showing high-quality 3D object generation. The SOTA method generates high-quality point clouds on specific categories,

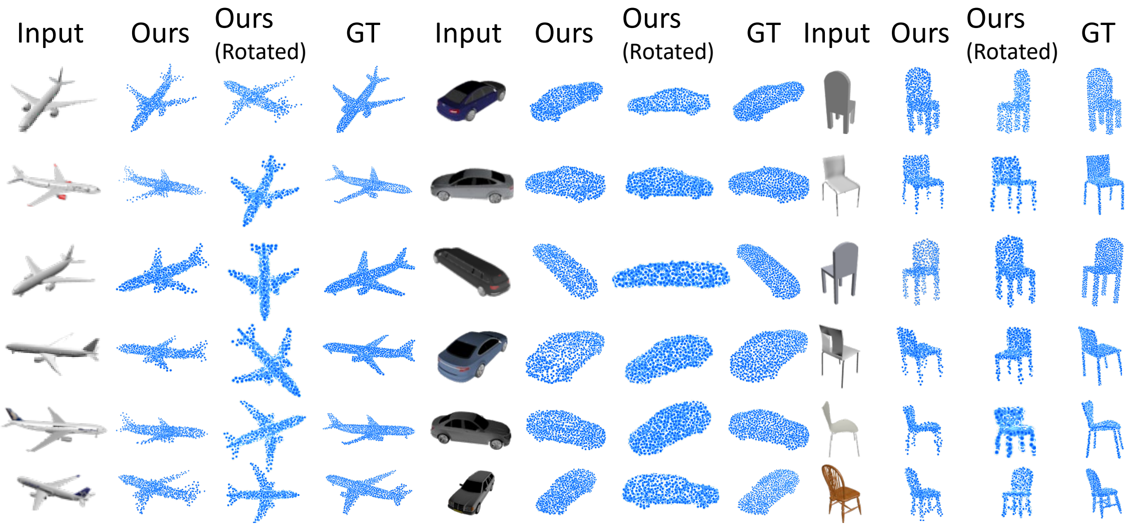

Figure 2. A qualitative analysis compares 3D point clouds generated by our method, *RGB2Point*, from single RGB images across airplane, car, and chair categories in ShapeNet against their target point clouds.

Category	[7]	[61]	[59]	[32]	[32]^2	**Ours**
airplane	0.225	0.215	0.266	0.473	**0.589**	0.581
bench	0.198	0.241	0.266	0.305	0.334	**0.511**
cabinet	0.256	0.308	0.317	0.203	0.211	**0.464**
car	0.211	0.220	0.268	0.359	0.372	**0.523**
chair	0.194	0.217	0.246	0.290	0.309	**0.544**
display	0.196	0.261	0.279	0.232	0.268	**0.487**
lamp	0.186	0.220	0.242	0.300	0.326	**0.471**
loudspeaker	0.229	0.286	0.297	0.203	0.210	**0.462**
rifle	0.356	0.364	0.410	0.522	**0.585**	0.567
sofa	0.208	0.260	0.277	0.205	0.224	**0.481**
table	0.263	0.305	0.327	0.270	0.297	**0.436**
telephone	0.407	0.575	**0.582**	0.331	0.389	0.483
watercraft	0.240	0.283	0.316	0.324	0.341	**0.552**
Average↑	0.244	0.289	0.315	0.309	0.343	**0.505**
Stdev.↓	0.067	0.097	0.092	0.099	0.123	**0.045**

Table 1. Comparison of the single image to point cloud generation on ShapeNet [7] with prior works. We use 0.01 as a distance threshold for the F-score that higher values represent better generation quality. We **bold** the best value and underline the current SOTA. Our model, *RGB2Point*, shows *47.16%* than the diffusion-based model [32], which is the SOTA model. We also calculate the standard deviation (Stdev.) over the categories to evaluate a stable generation quality. *RGB2Point* shows 63.1% stable generated point cloud quality compare to the SOTA [32]. [32]^2 uses image masks to guide its generation.

such as airplanes or rifles, but its overall stability over the categories is imbalanced. Unlike the biased performance, *RGB2Point* generates *64.1%* more stable generation quality than the SOTA method [32]. We calculate the performance stability using a standard deviation (Stdev.) and report it under the last row in Tab. 1. All the models are trained on the same ShapeNet [6]. This shows that our model is robust and capable of generating a high-quality, dense point cloud even if we expand the size of the output layer.

Moreover, *RGB2Point* achieved Chamfer distance of 4.05×10^2 (car category), 5.38×10^2 (chair), and 2.73×10^2 (aircraft). These scores represent improvements of 25.00%, 41.71%, and 51.08% over state-of-the-art benchmarks [18, 33]. Additionally, employing Earth Mover's distance, our model attains 3.59×10^2 (car), 7.80×10^2 (chair), and 5.01×10^2 (aircraft). This indicates improvement by 24.90%, 23.38%, and 32.57%, compared to the current state-of-the-art works [18, 33] as summarized in Tab. 2.

We validate the robustness of our model on the real 3D object dataset Pix3D [45], using the trained model on ShapeNet [6]. Our model generates targeted objects from the noisy background as we show generated results in Fig. 3. We follow the same evaluation setting as recent works [33, 34]. *RGB2Point* improves performs 54.57% and 42.1% better than the recent works [33, 34] in Chamfer distance, and Earth Mover's distance as shown in Tab. 3.

RGB2Point surpasses the current SOTA in performance with synthetic and real-world datasets, indicating *RGB2Point* is more robust in capturing the generated object. Using a single image, 3D Gaussian Splatting [19] offers a reconstruction method, albeit including the background. In contrast, our approach focuses on generating targeted objects trained amidst noisy backgrounds, as illustrated in Fig. 3.

4.5. Ablation Study

We conduct five additional experiments: 1) evaluating the influence of various parameter configurations, 2) assessing how the presence of the pre-trained ViT weights affects

Method	CD($\times 10^2$) $\downarrow$			EMD($\times 10^2$) $\downarrow$		
	Car	Chair	Aircraft	Car	Chair	Aircraft
Self-Sup. [34]	10.33	21.84	15.06	18.32	23.40	16.12
Self-Sup. [34]+L_C	6.39	13.58	8.66	6.42	16.46	12.53
Self-Sup. [34]+NN	5.48	10.91	7.11	4.95	14.93	11.07
DIFFER [33]	6.35	9.78	5.67	6.03	16.21	9.90
DIFFER [33]+L_G	5.63	9.23	5.58	5.35	13.07	9.44
ULSP [18]	6.64	10.49	5.70	6.89	10.93	7.43
ULSP [18]+L_G	6.13	10.0	7.37	5.83	10.24	9.99
ULSP [18]+Sup.	5.4	9.72	5.91	4.78	10.18	7.66
LION [50]	–	12.11	–	–	10.94	–
Ours	**4.05**	**5.38**	**2.73**	**3.59**	**7.80**	**5.01**
Ours without CFI	4.60	6.13	3.78	5.00	12.35	7.98
Ours without GPM	5.20	6.59	4.18	7.45	11.01	6.97

Table 2. The best values from different categories from ShapeNet [6] in Chamfer distance (CD) and Earth Mover's distance (CMD) are noted in **bold** and the current SOTA values are underlined. *RGB2Point* shows an average improvement of *39.26%* in Chamfer distance and *26.95%* in Earth Mover's distance among the three categories.

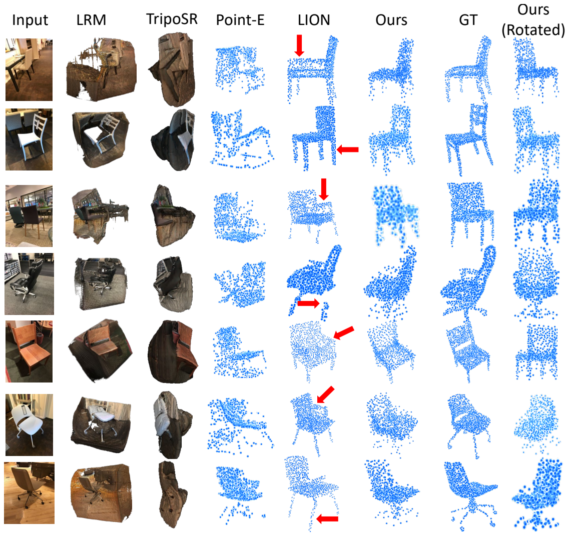

Figure 3. Generated point cloud data by *RGB2Point* using images from the real-world dataset Pix3D [45]. The first column shows an input RGB image, and the next two columns show a reconstructed mesh from LRM [16], TripoSR [48]. The third and fourth columns show reconstructed point clouds from Point-E [35] and LION [50]. The sixth left column shows generated point cloud data by *RGB2Point* and the column with *GT* shows its ground truth point cloud data. The red arrows highlight differences compared to *GT*. Also, we show a rotated view from our outputs in the last column.

Method	CD($\times 10^2$) $\downarrow$	EMD($\times 10^2$) $\downarrow$
Self-Sup. [34]	14.52	15.82
DIFFER [33]	14.33	16.09
LION [50]	16.97	14.68
Ours	**7.00**	**9.37**
Ours w/o CFI	8.53	11.5
Ours w/o GFM	8.82	13.23

Table 3. We test our model on the real 3D dataset Pix3D [45], and compared to the related works. The best values are shown in **bold** and the current SOTA is underlined. *RGB2Point* shows an improvement of *51.15%* in Chamfer distance (CD) and *36.17%* in Earth Mover's distance, compared to the previous state-of-the-art metrics.

the performance of 3D generation, 3) validating the effectiveness of ViT for point cloud generation task by replacing it to ResNet-50 [14], 4) evaluating the effectiveness of two modules: Contextual Feature Integrator and Geometric Projection Module, and 5) qualitative analysis of outputs using different n, (the number of points) where $n = 128$.

We experimented with 16 different combinations of model parameters (H, D, A) as illustrated in Fig. 1, and evaluated the trained models using Chamfer and Earth Mover's distance (Tab. 4). The optimal parameter set is $(H = 4, D = 2048, A = 1024)$, surpassing both *39.26%* and *26.95%* metrics compared to the current state-of-the-art performance [34] on the ShapeNet dataset. Interestingly, our analysis reveals that more attention heads do not consistently enhance generation performance.

We also analyzed the impact of pre-trained weights on the ViT by contrasting the two-generation quality metrics obtained from models with and without pre-trained weights. Using the models without pre-trained weights, the generations are worse with an average of 35.10%±20.59% and 93.11%±15.58% in Chamfer distance and Earth Mover's distance, respectively, compared to the models with pre-trained weights. This suggests that pre-trained weights play a crucial role in shaping 3D generations, highlighting the importance of selecting the right pre-trained weights for a particular task.

Vision Transformer [10] is widely leveraged in differ-

Hyper Parameters H	 D	 A	CD($\times 10^2$) ↓ Car	 Aircraft	 Chair	EMD($\times 10^2$) ↓ Car	 Aircraft	 Chair	Statistics($\times 10^2$) ↓ CD Avg.	 EMD Avg.
2	1024	1024	5.46	2.74	4.05	8.3	5.43	3.68	4.08	5.80
2	1024	2048	5.48	2.80	4.10	8.38	6.03	3.77	4.13	6.06
2	1024	4096	5.41	2.78	4.12	8.45	5.69	3.70	4.10	5.95
2	2048	1024	5.44	2.72	4.06	8.09	5.16	3.48	4.07	5.58
2	2048	2048	5.40	2.74	4.12	8.32	5.74	3.64	4.09	5.90
2	2048	4096	5.48	2.82	4.09	8.40	6.03	3.66	4.13	6.03
4	2048	2048	5.47	2.78	4.12	8.28	5.42	3.77	4.12	5.82
4	2048	1024	5.38	2.73	4.05	7.80	5.01	3.59	4.06	5.47
4	2048	4096	5.47	2.80	4.11	8.51	5.57	3.70	4.13	5.93
8	2048	2048	5.49	2.75	4.09	8.50	5.29	3.73	4.11	5.84
8	2048	4096	5.45	2.84	4.14	8.28	5.71	3.80	4.14	5.93
8	2048	1024	5.42	2.77	4.06	8.09	5.49	3.55	4.09	5.71
16	1024	1024	5.41	2.74	4.02	8.29	6.01	3.53	4.06	5.94
16	1024	2048	5.38	2.78	4.08	8.30	5.82	3.64	4.08	5.92
16	2048	1024	5.41	2.76	4.06	8.04	5.21	3.55	4.07	5.60
16	2048	2048	5.40	2.77	4.13	8.10	5.23	3.67	4.10	5.67

Table 4. The ablation study using ShapeNet [6] with different numbers of attention heads, H and the dimensions of the feedforward D, aggregator, A from Fig. 1. We evaluate various parameters using Chamfer Distance (CD) and Earth Mover's Distance (EMD). The best hyper-parameter set is underlined.

Hyper Parameters H	 D	 A	CD($\times 10^2$) ↓ Car	 Aircraft	 Chair	EMD($\times 10^2$) ↓ Car	 Aircraft	 Chair	Difference(%) CD	 EMD
2	1024	1024	6.93	4.07	5.54	14.23	9.78	12.80	35.11	111.49
2	1024	2048	6.94	3.33	5.14	14.18	9.13	10.85	24.37	87.87
2	1024	4096	7.02	3.26	6.09	14.06	8.74	13.80	33.04	105.05
2	2048	1024	7.26	3.40	4.88	14.08	9.35	9.51	27.29	96.76
2	2048	2048	6.92	3.59	4.94	13.91	9.26	10.03	25.91	87.56
2	2048	4096	10.64	6.99	7.79	11.33	6.84	10.83	105.14	60.36
4	2048	2048	6.96	3.67	5.23	13.82	9.42	11.99	28.30	101.79
4	2048	1024	6.92	3.33	5.24	14.18	8.99	11.38	27.18	110.52
4	2048	4096	8.41	5.48	5.98	11.25	6.60	10.75	60.33	60.78
8	2048	2048	6.84	3.36	5.21	13.78	9.20	11.63	25.05	97.55
8	2048	4096	6.86	3.24	5.70	14.04	9.09	12.98	27.19	102.98
8	2048	1024	6.91	3.87	4.88	13.99	9.46	9.33	27.62	91.38
16	1024	1024	7.32	3.34	4.83	13.92	9.04	8.78	27.16	78.13
16	1024	2048	6.89	4.39	5.27	13.82	10.01	12.07	35.14	102.16
16	2048	1024	6.84	3.67	5.16	13.83	9.40	11.31	28.31	105.58
16	2048	2048	7.07	3.36	4.87	13.81	9.13	9.33	24.41	89.76

Table 5. We show the generation results from ShapeNet [6] *without* pre-trained Vision Transformer weights in Chamfer distance (CD) and Earth Mover's distance (EMD) using different parameters including attention heads, H, dimensions of feedforward, D and aggregator, A. The last two columns represent performance differences depending on the existence of pre-trained weights.

ent computer vision tasks such as image segmentation [13, 60, 66], depth estimation [1, 39, 67] and 3D object detection [56]. However, we do not know the effectiveness of a single image-conditioned point cloud generation task. In this task, we evaluate the role of the image extractor by swapping out ViT to a pre-trained ResNet-50 [14]. We train a model on the same environment to compare the generation quality on ShapeNet [6] on three categories: airplane, car, and chair. As we show in Tab. 6, the ViT-based model generates 11.4 % and 33.9% better quality of point cloud given a single RGB image in terms of Chamfer Distance and Earth Mover's Distance, respectively. This study shows that similar to other computer vision tasks that leverage the power of ViT, 3D point cloud generation could be one of the fields.

Model	CD($\times 10^2$) ↓ Car	 Aircraft	 Chair	EMD($\times 10^2$) ↓ Car	 Aircraft	 Chair	Average($\times 10^2$) ↓ CD	 EMD
ViT	2.73	4.05	5.38	7.80	5.01	3.59	4.05	5.47
ResNet50	4.55	3.11	6.06	5.08	7.57	12.17	4.57	8.27

Table 6. We show single image conditioned 3D point cloud generated results using two different image extractor models: ViT [10] and ResNet50 [14]. Both models are trained on the same environment but the only difference is its image feature extractor. We validate them using Chamfer Distance and Earth Mover's Distance on their generated point cloud data.

Moreover, we validate the effectiveness of our two modules by training without them. We evaluate two models that remain the same but remove each module from the entire proposed architecture. We report a significant performance drop compared to metrics from the original architecture in Tab. 7. Even if a module is removed, Tab. 3 shows that our approach is at least 39.30% and 9.88% better in Chamfer Distance and Earth Mover's distance compared to the previous works.

Module	ShapeNet CD	 EMD	Pix3D CD	 EMD
CFI	-19.27%	-54.43%	-21.86%	-22.73%
GPM	-31.30%	-55.06%	-26.00%	-41.20%

Table 7. We validated the effectiveness of our two modules by removing them from the model architecture, and we show the average of all scenes from this paper. The performances were measured using Chamfer Distance and Earth Mover's Distances. When we removed the modules CFI and GPM, the performance **dropped** compared to metrics from the original pipeline.

We conduct an additional experiment by varying the number of point clouds, n. Specifically, we reduced the number of points to test whether this decrease affects the preservation of the object's overall shape during generation. As demonstrated in Fig. 4, even with a smaller n, the overall shapes are still generated accurately without losing details.

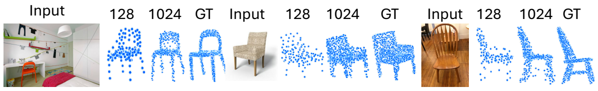

Figure 4. We compare the output of different numbers of point clouds. Our original pipeline generates 1,024 point clouds but we show 128 point clouds. The overall shape is preserved instead of missing a random region of point clouds.

4.6. Limitation

We report failure cases using images from a real-world dataset [45] in Fig. 5. The common issue (identified in Fig. 5 as (1-4)) is the lack of level of detail in our generated

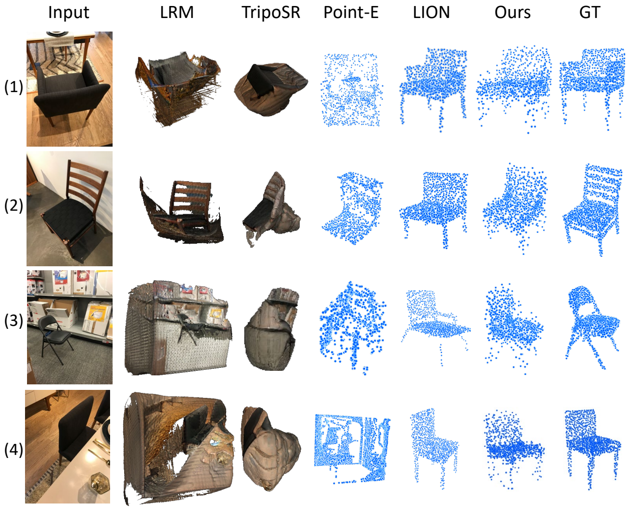

Figure 5. Three failure cases from a complex real-world dataset, [45], with their input images, single image-based 3D reconstructions as a mesh [16, 48] and 3D point cloud [35, 50], ours, and the ground truth 3D point cloud data.

point cloud data. For (1), *RGB2Point* generated a chair with shorter legs than the ground truth due to the limitation of viewpoint, especially a top view in this case. Some parts of the back support are missing in the cases of (2) and (3). And (4) shows an occlusion that is about 50% of its original shape by a desk. Compared to our failure generations, mesh reconstruction methods [16, 48] do not generate any related objects. Point-E [35] cannot generate a chair at all from this real-world case scenario. LION [50] gives better quality of point cloud than Point-E [35] but still it has low accuracy based on the given RGB image as we show this behavior in Tab. 3.

5. Conclusions and Future Works

We introduce a fast, high-quality 3D point cloud generative model from a single image. Leveraging the synthetic dataset from Shapenet [6], *RGB2Point* surpassed both Chamfer distance and Earth Mover's distance by *39.26%* and *26.95%* respectively. In addition, our work shows an improvement on the real-world dataset [45] outperforming the current SOTA by *51.15%* and *36.17%* using Chamfer distance and Earth Mover's distance. In addition to the higher quality generated point cloud, our model shows a **15,133** times faster inference time than eye-catching diffusion-based models [32, 35].

We explore the significance of utilizing pre-trained weights for the ViT model, showing performance on average disparities of 35.10%±20.59% and 93.11%±15.58% using Chamfer distance and Earth Mover's distance, respectively. We conclude that a pre-training weight on a 2D image affects the performance of 3D generation quality.

Furthermore, we first, as we know of, evaluate the effectiveness of ViT compared to CNN-based image feature extractor in the 3D point cloud generation field. ViT-based generation model provides 11.4% or 33.9% better quality to Chamfer Distance and Earth Mover's Distance.

Also, we validate the effectiveness of our modules, Context Feature Integrator and Geometric Projection Module, in Tab. 7 that shows an average of 23.93% and 31.97% **performance drop** using the real datset [45] in Chamfer Distance and Earth Mover's Distance.

Based on the performance and efficiency of our model, it can be used as a prior before getting actual lidar-sensor scanning, which requires multi-view scans. Our method generates a high-quality 3D point cloud from a single image in just 0.1 seconds, offering a fast and accurate alternative.

Future work could adapt *RGB2Point* for generating domain-specific objects by combining it with pre-trained weights, enabling 3D point cloud generation on desktop-level hardware. Expanding to multi-view images, with cross-attention mechanisms, could improve accuracy by leveraging complementary information from different perspectives, enhancing fidelity and robustness. Additionally, integrating a differentiable renderer for RGB texturing would increase visual quality. With VRAM efficiency, *RGB2Point* could be optimized for AR/VR deployment, achieving real-time, on-device use. Its fast generation time (0.15 seconds per image) makes it ideal for robotics tasks like path planning and object evasion.

A potential **negative societal impact** is on privacy considerations. Since our model generates a 3D point cloud from a single RGB image, a leaked photo of a new product could be used to estimate its dimensions or create a design template with minimal effort. For instance, if a new chair design is leaked before its official launch, others might attempt to produce a replicated version.

Acknowledgement

This work was supported in part by the U.S. National Science Foundation under awards No. 2412928 and 2417510. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect those of the National Science Foundation. This work is based upon efforts supported by the USDA-NIFA grants 2024-67021-42879 "Improving Forest Management Through Automated Measurement of Tree Geometry" and 2023-08563 "Optimizing soybean canopy architecture for efficient light and water use under climate change". The views and conclusions contained herein are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Government or NRCS. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright annotation therein.

References

[1] Shariq Farooq Bhat, Ibraheem Alhashim, and Peter Wonka. Adabins: Depth estimation using adaptive bins. In *CVPR*, 2021. 7

[2] Gabriel J Brostow, Jamie Shotton, Julien Fauqueur, and Roberto Cipolla. Segmentation and recognition using structure from motion point clouds. In *ECCV*, 2008. 1

[3] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *NeurIPS*, 2020. 3

[4] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*. Springer, 2020. 3

[5] Eric R Chan, Koki Nagano, Matthew A Chan, Alexander W Bergman, Jeong Joon Park, Axel Levy, Miika Aittala, Shalini De Mello, Tero Karras, and Gordon Wetzstein. Generative novel view synthesis with 3d-aware diffusion models. In *ICCV*, 2023. 3

[6] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015. 2, 3, 4, 5, 6, 7, 8

[7] Christopher B Choy, Danfei Xu, JunYoung Gwak, Kevin Chen, and Silvio Savarese. 3d-r2n2: A unified approach for single and multi-view 3d object reconstruction. In *ECCV*, 2016. 1, 2, 4, 5

[8] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 1, 2

[9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2018. 3

[10] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *ICLR*, 2021. 1, 2, 3, 4, 6, 7

[11] Haoqiang Fan, Hao Su, and Leonidas J Guibas. A point set generation network for 3d object reconstruction from a single image. In *CVPR*, 2017. 2

[12] Yuxin Fang, Bencheng Liao, Xinggang Wang, Jiemin Fang, Jiyang Qi, Rui Wu, Jianwei Niu, and Wenyu Liu. You only look at one sequence: Rethinking transformer in vision through object detection. *NeurIPS*, 2021. 3

[13] Jiaqi Gu, Hyoukjun Kwon, Dilin Wang, Wei Ye, Meng Li, Yu-Hsin Chen, Liangzhen Lai, Vikas Chandra, and David Z Pan. Multi-scale high-resolution vision transformer for semantic segmentation. In *CVPR*, 2022. 7

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 1, 2, 6, 7

[15] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *NeurIPS*, 2020. 1, 3

[16] Yicong Hong, Kai Zhang, Jiuxiang Gu, Sai Bi, Yang Zhou, Difan Liu, Feng Liu, Kalyan Sunkavalli, Trung Bui, and Hao Tan. Lrm: Large reconstruction model for single image to 3d. *ICLR*, 2024. 3, 4, 6, 8

[17] Qibin Hou, Zihang Jiang, Li Yuan, Ming-Ming Cheng, Shuicheng Yan, and Jiashi Feng. Vision permutator: A permutable mlp-like architecture for visual recognition. *IEEE TPAMI*, 2022. 3

[18] Eldar Insafutdinov and Alexey Dosovitskiy. Unsupervised learning of shape and pose with differentiable point clouds. In *NeurIPS*, 2018. 2, 3, 5, 6

[19] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM TOG*, 2023. 5

[20] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *ICLR*, 2015. 4

[21] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 1

[22] Jae Joong Lee, Bosheng Li, Sara Beery, Jonathan Huang, Songlin Fei, Raymond A Yeh, and Bedrich Benes. Tree-d fusion: Simulation-ready tree dataset from single images with diffusion priors. *ECCV*, 2024. 3

[23] Jae Joong Lee, Bosheng Li, and Bedrich Benes. Latent l-systems: Transformer-based tree generator. *ACM TOG*, 2023. 3

[24] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 1

[25] Ruoshi Liu, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. Zero-1-to-3: Zero-shot one image to 3d object. In *ICCV*, 2023. 1, 3

[26] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019. 3

[27] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *ICCV*, 2021. 3

[28] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *ICCV*, 2021. 3

[29] Shitong Luo and Wei Hu. Diffusion probabilistic models for 3d point cloud generation. In *CVPR*, 2021. 3

[30] Zhaoyang Lyu, Zhifeng Kong, Xudong Xu, Liang Pan, and Dahua Lin. A conditional point diffusion-refinement paradigm for 3d point cloud completion. *ICLR*, 2022. 3

[31] Andrew L Maas, Awni Y Hannun, Andrew Y Ng, et al. Rectifier nonlinearities improve neural network acoustic models. In *ICML*, 2013. 4

[32] Luke Melas-Kyriazi, Christian Rupprecht, and Andrea Vedaldi. Pc2: Projection-conditioned point cloud diffusion for single-image 3d reconstruction. In *CVPR*, 2023. 2, 3, 4, 5, 8

[33] K L Navaneet, Priyanka Mandikal, Varun Jampani, and R Venkatesh Babu. DIFFER: Moving beyond 3d reconstruction with differentiable feature rendering. In *CVPRW*, 2019. 2, 3, 5, 6

[34] K. L. Navaneet, A. Mathew, S. Kashyap, W. Hung, V. Jampani, and R. Venkatesh Babu. From image collections to point clouds with self-supervised shape and pose networks. CVPR, 2020. 2, 3, 4, 5, 6

[35] Alex Nichol, Heewoo Jun, Prafulla Dhariwal, Pamela Mishkin, and Mark Chen. Point-e: A system for generating 3d point clouds from complex prompts. *arXiv preprint arXiv:2212.08751*, 2022. 1, 3, 4, 6, 8

[36] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *CVPR*, 2017. 1

[37] Guocheng Qian, Jinjie Mai, Abdullah Hamdi, Jian Ren, Aliaksandr Siarohin, Bing Li, Hsin-Ying Lee, Ivan Skorokhodov, Peter Wonka, Sergey Tulyakov, et al. Magic123: One image to high-quality 3d object generation using both 2d and 3d diffusion priors. *ICLR*, 2024. 1, 3

[38] Thinal Raj, Fazida Hanim Hashim, Aqilah Baseri Huddin, Mohd Faisal Ibrahim, and Aini Hussain. A survey on lidar scanning mechanisms. *Electronics*, 2020. 1

[39] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *ICCV*, 2021. 7

[40] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022. 1, 3

[41] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *CVPR*, 2016. 1

[42] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *CVPR*, 2016. 1

[43] Johannes Lutz Schönberger, Enliang Zheng, Marc Pollefeys, and Jan-Michael Frahm. Pixelwise view selection for unstructured multi-view stereo. In *ECCV*, 2016. 1

[44] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition, 2015. 1, 2

[45] Xingyuan Sun, Jiajun Wu, Xiuming Zhang, Zhoutong Zhang, Chengkai Zhang, Tianfan Xue, Joshua B Tenenbaum, and William T Freeman. Pix3d: Dataset and methods for single-image 3d shape modeling. In *CVPR*, 2018. 2, 3, 4, 5, 6, 7, 8

[46] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *ICML*, 2019. 1

[47] Jiapeng Tang, Xiaoguang Han, Mingkui Tan, Xin Tong, and Kui Jia. Skeletonnet: A topology-preserving solution for learning mesh reconstruction of object surfaces from rgb images. *IEEE TPAMI*, 2021. 2

[48] Dmitry Tochilkin, David Pankratz, Zexiang Liu, Zixuan Huang, Adam Letts, Yangguang Li, Ding Liang, Christian Laforte, Varun Jampani, and Yan-Pei Cao. Triposr: Fast 3d object reconstruction from a single image. *arXiv preprint arXiv:2403.02151*, 2024. 3, 4, 6, 8

[49] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023. 3

[50] Arash Vahdat, Francis Williams, Zan Gojcic, Or Litany, Sanja Fidler, Karsten Kreis, et al. Lion: Latent point diffusion models for 3d shape generation. *NeurIPS*, 2022. 3, 4, 6, 8

[51] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *NeurIPS*, 2017. 2

[52] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *NeurIPS*, 2017. 1, 2, 3

[53] Dan Wang, Xinrui Cui, Xun Chen, Zhengxia Zou, Tianyang Shi, Septimiu Salcudean, Z Jane Wang, and Rabab Ward. Multi-view 3d reconstruction with transformers. In *ICCV*, 2021. 1, 2

[54] Dan Wang, Xinrui Cui, Xun Chen, Zhengxia Zou, Tianyang Shi, Septimiu Salcudean, Z Jane Wang, and Rabab Ward. Multi-view 3d reconstruction with transformers. In *ICCV*, 2021. 3

[55] Nanyang Wang, Yinda Zhang, Zhuwen Li, Yanwei Fu, Wei Liu, and Yu-Gang Jiang. Pixel2mesh: Generating 3d mesh models from single rgb images. In *ECCV*, 2018. 2

[56] Yikai Wang, TengQi Ye, Lele Cao, Wenbing Huang, Fuchun Sun, Fengxiang He, and Dacheng Tao. Bridged transformer for vision and point cloud 3d object detection. In *CVPR*, 2022. 7

[57] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. *NeurIPS*, 2021. 3

[58] Haozhe Xie, Hongxun Yao, Xiaoshuai Sun, Shangchen Zhou, and Shengping Zhang. Pix2vox: Context-aware 3d reconstruction from single and multi-view images. In *ICCV*, 2019. 1, 2

[59] Haozhe Xie, Hongxun Yao, Shengping Zhang, Shangchen Zhou, and Wenxiu Sun. Pix2vox++: Multi-scale context-aware 3d object reconstruction from single and multiple images. *IJCV*, 2020. 1, 2, 4, 5

[60] Jiarui Xu, Shalini De Mello, Sifei Liu, Wonmin Byeon, Thomas Breuel, Jan Kautz, and Xiaolong Wang. Groupvit: Semantic segmentation emerges from text supervision. In *CVPR*, 2022. 7

[61] Farid Yagubbayli, Yida Wang, Alessio Tonioni, and Federico Tombari. Legoformer: Transformers for block-by-block multi-view 3d reconstruction. *arXiv preprint arXiv:2106.12102*, 2021. 4, 5

[62] Ikuya Yamada, Akari Asai, Hiroyuki Shindo, Hideaki Takeda, and Yuji Matsumoto. LUKE: Deep contextualized entity representations with entity-aware self-attention. In *Empirical Methods in Natural Language Processing*, 2020. 3

[63] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. Xlnet: Generalized autoregressive pretraining for language understanding. *NeurIPS*, 2019. 3

[64] Michihiro Yasunaga, Jure Leskovec, and Percy Liang. Linkbert: Pretraining language models with document links. In *Association for Computational Linguistics*, 2022. 3

[65] Xumin Yu, Lulu Tang, Yongming Rao, Tiejun Huang, Jie Zhou, and Jiwen Lu. Point-bert: Pre-training 3d point cloud transformers with masked point modeling. In *CVPR*, 2022. 1, 3

[66] Bowen Zhang, Zhi Tian, Quan Tang, Xiangxiang Chu, Xiaolin Wei, Chunhua Shen, et al. Segvit: Semantic segmentation with plain vision transformers. *NeurIPS*, 2022. 7

[67] Chaoqiang Zhao, Youmin Zhang, Matteo Poggi, Fabio Tosi, Xianda Guo, Zheng Zhu, Guan Huang, Yang Tang, and Stefano Mattoccia. Monovit: Self-supervised monocular depth estimation with a vision transformer. In *International Conference on 3D Vision*, 2022. 7

[68] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip HS Torr, and Vladlen Koltun. Point transformer. In *ICCV*, 2021. 3

[69] Chuanxia Zheng and Andrea Vedaldi. Free3d: Consistent novel view synthesis without 3d representation. In *CVPR*, 2024. 3