3D RECONSTRUCTION BY PARAMETERIZED SURFACE MAPPING

Pierre-Alain Langlois[1] *Matthew Fisher*[3] *Oliver Wang*[3] *Vladimir Kim*[3]
Alexandre Boulch[2] *Renaud Marlet*[1,2] *Bryan Russell*[3]

[1] LIGM, Ecole des Ponts, Univ Gustave Eiffel, CNRS, Marne-la-Vallée, France
[2] valeo.ai, F-75008 Paris, France
[3] Adobe Research, 601 Townsend St, San Francisco, CA 94107, United States

2021 IEEE International Conference on Image Processing (ICIP) | 978-1-6654-4115-5/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICIP42928.2021.9506425

ABSTRACT

We introduce an approach for computing a 3D mesh from one or more views of an object by establishing dense correspondences between pixels in the views and 3D locations on a learnable parameterized surface. We propose a multi-view shape encoder that can be jointly trained with the AtlasNet surface parameterization. The shape is further refined using a novel geometric cycle-consistency loss between the learnable parameterized surface and input views. We demonstrate the efficacy of our approach on the ShapeNet-COCO dataset.

***Index Terms*—** 3D Reconstruction, multi-view, surface mapping, deformation, learning

1. INTRODUCTION

Classical multi-view 3D reconstructions approaches establish dense correspondences between pixels or image regions depicting the same surface locations, followed by triangulation to yield a 3D shape [1]. However, these approaches fail to find correspondences in flat, non-textured regions, and they cannot extrapolate to reconstruct nonvisible/occluded surfaces. In addition, they typically yield point clouds, whereas meshes are more useful in most real-world downstream applications.

Incorporating data-driven shape priors into the 3D reconstruction process has been used to address the above limitations [2, 3]. However, an ideal shape prior should be able to adapt in order to finely align with its depiction in the input views [4]. Furthermore, the prior should not require manual annotation of the surface parameterization, i.e., landmarks of shape parts, and should be learnable from training data.

To address the above issues, we propose a method that learns to establish dense correspondences between image pixels and a learned parameterized surface, which is trained on a corpus of shapes from the same class, without known surface parameterizations. The approach has two steps. First, as opposed to the Kulkarni *et al.* [5] who use a fixed template, we generate an initial shape template with a multi-view shape encoder, which extends AtlasNet [2] to multiple viewpoints. Second, this template, specific to the shape we want to reconstruct, is refined using a cycle-consistency loss which allows the shape to adapt and finely align with its depiction in the input views.

We evaluate our method compared to a recent state-of-the-art approach for 3D reconstruction from multiple views [6]. We show that our method outperforms the baselines when averaging over three shape categories. Moreover, as opposed to XNOCS [6] and traditional multi-view stereo methods [1], our shape parameterization naturally yields a mesh that can be used out-of-the-box for most graphics applications.

2. RELATED WORK

Single-view reconstruction. Most single-view methods learn general 3D priors over images or image patches [7, 8]. To relax the explicit 3D supervision, Kanazawa *et al.* [9] learn class-specific reconstruction priors over a deformable mesh. It generates impressive textured reconstructions from a single image, but still requires extra supervision in the form of corresponding keypoints during training, whereas we infer them. More recently, Canonical Surface Mapping (CSM) [5] predicts a UV mapping from a single image onto a canonical model, trained entirely using self-supervision, by introducing a geometric cycle-consistency term. For Kulkarni *et al.* [10], the same mapping is applied but the canonical surface mesh can deform given an articulation parameter, which allows shape alignment to an input image. Our work is inspired by these, but differs in two key ways. First, we do not assume a fixed, manually chosen, canonical model; we rather optimize over a learned surface parameterization to better match the observed data. Second, while CSM works only for single views, we use a multi-view cycle-consistency term to better integrate multi-view information when available. This is critical when reconstructing both the mapping and the underlying model jointly, due to the ill-posed nature of the problem. Indeed, using multi-view information resolves the depth ambiguities that arise when looking at a scene from a single static camera.

Data-driven multi-view reconstruction. Recently, data-driven methods have been introduced that attempt to extend classical multi-view stereo reconstruction aproaches to leverage much stronger priors to deal with ambiguities [11, 12]. Our

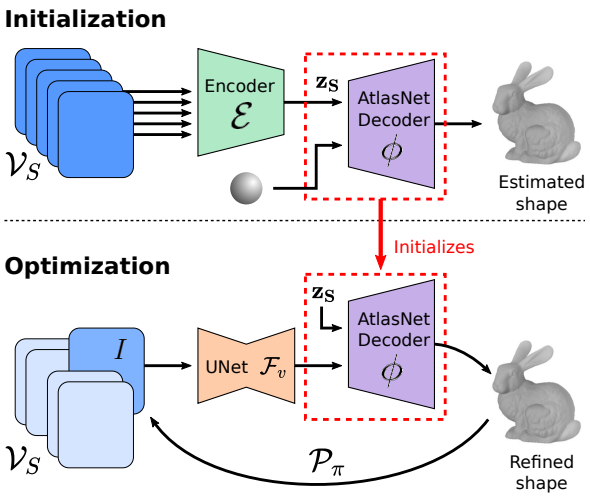

Fig. 1: **Networks for inferring a depicted shape.** Our pipeline consists of two steps. First, a trained encoder generates an initial latent shape representation from the calibrated images. Next, the decoder is iteratively improved through geometric and mask constraints to further improve reconstruction.

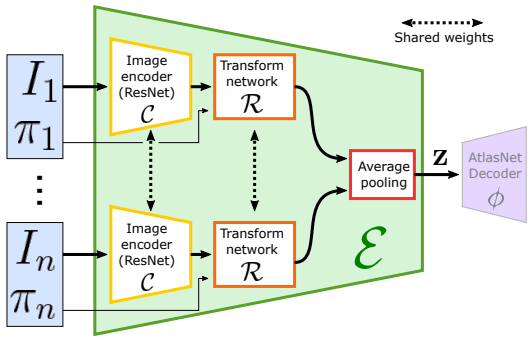

Fig. 2: **Multi-view shape encoder.** Our multi-view shape encoder $\mathcal{E}$ aggregates transformed features given any number of input views and corresponding camera parameters. A decoder ϕ then maps the encoded representation to the output mesh.

work differs from these works in that we reconstruct a mesh rather than a point cloud, which typically requires stronger data priors, as observed in the single-view setting.

Other approaches for learning mesh reconstruction rely on a manual alignment between the cameras and a shape generator, and optimize a photometric loss, which is not robust to illumination changes [4]. Alternately, Pixel2Mesh++ [13] presents a feed-forward neural network that maps input views and a base sphere mesh to an output adapted mesh. However, this approach differs fundamentally from ours in that it does not establish a dense surface mapping from pixels to the reconstructed mesh, which is required for fine-scale alignment.

More recently, a number of multi-view methods for novel-view synthesis have proposed eliminating the explicit mesh reconstruction altogether and instead learn features, e.g., in a voxel grid [14, 15] or a set of orthogonal planes spaced in 3D [16]. These approaches demonstrate strong view synthesis results, but cannot be used in cases where mesh reconstructions are still required. In this work, we compare against XNOCS [6] which is one of the most recent works that integrate the multi-view constraints in a deep reconstruction framework.

3. LEARNING A MULTI-VIEW PARAMETERIZED SURFACE MAPPING

We consider a shape S and a collection $\mathcal{V}_S$ of views of S, where a view $v = (I, \pi) \in \mathcal{V}_S$ consists of an image I with associated camera intrinsics and extrinsics π. We seek to output a surface mesh $\mathcal{M}$ of the shape in object coordinates. We assume that a template $\mathcal{T}$ parameterizes the output surface mesh and that the mapping $\mathcal{V}_S \to \mathcal{T} \to \mathcal{M}$ produces the output. Our goal is to learn this parameterized surface mapping from image pixel locations to the output surface. For this, we present a two-step approach, depicted in Figure 1. The first step (initialization) aims at estimating an initial template, represented by a latent vector $\mathbf{z_S}$ and the weights of an AtlasNet decoder ϕ [2]. The second step (optimization) improves the shape representation using a cycle-consistency loss.

3.1. Initialization

Multi-view shape encoder. To compute the initial latent shape feature $\mathbf{z_S}$, we propose a multi-view shape encoder $\mathcal{E}$ jointly trained with the shape parameterization ϕ, cf., Fig. 2.

For each view $v = (I, \pi) \in \mathcal{V}_S$, the image I is encoded using a convolutional neural network $\mathcal{C}$. Now the resulting features are expressed in a coordinate system linked to the view. Therefore, we apply a neural network $\mathcal{R}$ that transforms the encoded representation with respect to the view camera parameters π in a representation expressed in the canonical pose. The multi-view encoder $\mathcal{E}$ aggregates by average pooling the features extracted from each view. The resulting representation ensures equivariance with respect to the camera parameters and can handle a variable number of input views. Denoting $|\mathcal{V}_S|$ the size of the view collection $\mathcal{V}_S$, the output of our multi-view shape encoder $\mathcal{E}$ is the average of encoded transformed views:

$$\mathbf{z_S} = \mathcal{E}(\mathcal{V}_S) = \frac{1}{|\mathcal{V}_S|} \sum_{(I,\pi)\in\mathcal{V}_S} \mathcal{R}(\mathcal{C}(I), \pi). \qquad (1)$$

We assume here a spherical parameterization of a shape collection $\mathcal{S}$: the template $\mathcal{T}$ we consider is a sphere, which covers the range of genus-zero shapes. We learn our parametrization using an AtlasNet decoder ϕ. To do so, we need to establish correspondences between each shape $S \in \mathcal{S}$ in the collection. We achieve this goal by establishing per-shape correspondences $\mathcal{Q}_S$ via the template $\mathcal{T}$. While there are a

variety of spherical parameterizations of a shape [17], we use a gnomonic projection due to its simplicity. We jointly optimize the parameters of $\mathcal{E}$ and ϕ using the shape prior loss $\mathcal{L}_{\text{shape}}$ as the sum of L_1 losses over shapes $S \in \mathcal{S}$ and established point correspondences $(\mathbf{q}, \mathbf{q}') \in \mathcal{Q}_S$ between the template $\mathcal{T}$ and the shape S:

$$\mathcal{L}_{\text{shape}}(\phi, \mathcal{S}) = \sum_{S \in \mathcal{S}} \sum_{(\mathbf{q}, \mathbf{q}') \in \mathcal{Q}_S} \left\| \phi_{\mathcal{E}(\mathcal{V}_S)}(\mathbf{q}) - \mathbf{q}' \right\|_1 \quad (2)$$

Training procedure. Given a dataset of 3D shapes, we render multiple views and jointly train the parameters for the multi-view shape encoder $\mathcal{E}$ and the parameterization ϕ. We follow the network architectures and two-stage training procedure described in Groueix *et al.* [2] to train a single-view model comprising the parameterization ϕ and single-view encoder $\mathcal{C}$ using loss $\mathcal{L}_{\text{shape}}$. We then jointly train the multi-view shape encoder and parameterization by initializing the parameters for ϕ and $\mathcal{C}$ from the single-view training step.

In our implementation, $\mathcal{C}$ is a ResNet-18 [18], $\mathcal{R}$ is a multi-layer perceptron with two layers and the AtlasNet decoder ϕ is a multi-layer perceptron with four layers.

3.2. Optimization

Given the estimated shape produced at the initialization, it is possible to further improve the reconstruction.

Similar to CSM [5], we employ cycle consistency in our reconstruction loss to learn the parametric surface mapping: after mapping a pixel location $\mathbf{p}$ in an image to the surface mesh via the template, it should project back to the original location $\mathbf{p}$. We further encourage that the projected point lies within a segmentation mask $\mathcal{X}_v$ of the depicted shape, in each view v. The segmentation mask can be obtained with an off-the-shelf instance segmentation algorithm [19].

Let $v = (I, \pi)$ be a view in collection $\mathcal{V}_S$ and $\mathcal{F}_v : \mathbb{R}^2 \to \mathcal{T}$ be a learnable mapping from a 2D pixel location $\mathbf{p}$ in image I to a point on template $\mathcal{T}$. We note $\mathcal{P}_\pi : \mathbb{R}^3 \to \mathbb{R}^2$ the function projecting a 3D point on the image plane and define the cycle projection function $\mathcal{K} : \mathbb{R}^2 \to \mathbb{R}^2$ as $\mathcal{K}(\mathbf{p}) = \mathcal{P}_\pi \circ \phi_{\mathbf{z_S}} \circ \mathcal{F}_v(\mathbf{p})$.

Our reconstruction loss is the sum of squared-reprojection errors and squared-chamfer distances to the segmentation mask over all pixels, weighted by $\lambda = 0.25$:

$$\mathcal{L}_{\text{rec}}(\mathcal{F}_v, \phi_{\mathbf{z_S}}) = \sum_{\mathbf{p} \in \mathcal{X}_v} \|\mathcal{K}(\mathbf{p}) - \mathbf{p}\|_2^2 + \lambda \min_{\mathbf{p}' \in \mathcal{X}_v} \|\mathcal{K}(\mathbf{p}) - \mathbf{p}'\|_2^2 \quad (3)$$

Shape refinement procedure. We initialize the latent vector $\mathbf{z_S}$ and ϕ with the weights obtained at the initialization step. In our implementation, the mapping $\mathcal{F}_v$ is a 4-level U-Net [20] with a ResNet backbone. $\mathcal{F}_v$ has 3 output dimensions which are normalized in order to represent a UVW mapping on the template sphere $\mathcal{T}$. To properly initialize the cycles, the U-Net weights are set by training to reconstruct the UVW rendering of the initial shape. Once the pipeline is initialized, we optimize both the U-Net and the decoder using $\mathcal{L}_{\text{rec}}$.

4. EXPERIMENTS

We report qualitative and quantitative performance compared to the baselines. We consider the task of reconstructing a depicted shape given a set of views with known camera and object segmentation mask for each view.

4.1. Dataset and evaluation criteria

For our controlled setup, we evaluate on the publicly available ShapeNetCOCO dataset [6], which has 640x480 pixels rendered views for three shape categories ("airplane", "car", "chair") from ShapeNet [21]. Note that ShapeNetCOCO is more challenging than the dataset of Choy *et al.* [22] due to the composited natural image background and larger distances from the view's camera to the shape. We also found that for the Choy *et al.* [22] rendered views, the camera calibration was not accurate, which limits the accuracy of the reconstructions.

For evaluation, we sample 10k points on the ground-truth shape [13] and 10k points on the output shapes [6]. Given a threshold τ which is 1% of the ground-truth mesh bounding-box diagonal, we écalculate the fraction of output points finding a ground-truth point within τ (precision) and vice versa (recall). We follow Knapitsch *et al.* [23] and report the F1-score, which is the harmonic mean of precision and recall. We follow Sridhar *et al.* [6] and report squared-symmetric Chamfer distance (multiplied by 100). Note that these metrics are point-based, not meshed-based (which our network outputs), in order to be comparable with XNOCS [6].

4.2. Experimental results

Baselines. We compare against XNOCS [6] with multi-view aggregation. This approach trains separate models for each shape category. To be comparable, we train our approach by randomly sampling (only) five images for the input views.

We also compare against a fixed single-template baseline. For this baseline, we randomly choose a template shape for each shape class and compare the chosen template against the validation set. (The shapes are oriented and scaled consistently across the dataset.) We report the average across ten randomly selected templates for each category. For a particular selected template, this baseline is an upper bound for CSM [5].

Results compared to baselines. We train our method (*Our full model*) for each shape category separately. We test also with 5 input images to be comparable with XNOCS [6]. We report results in Table 1. Averaging over the three shape categories, our approach outperforms the baselines for both the F1-score and the Chamfer distance criteria. Note that the template baseline performs surprisingly well. In fact, as noted in Tatarchenko *et al.* [24], there is large overlap of the shapes in the train/val splits. We show qualitative results in Figure 3. The point clouds generated by XNOCS are very noisy. We

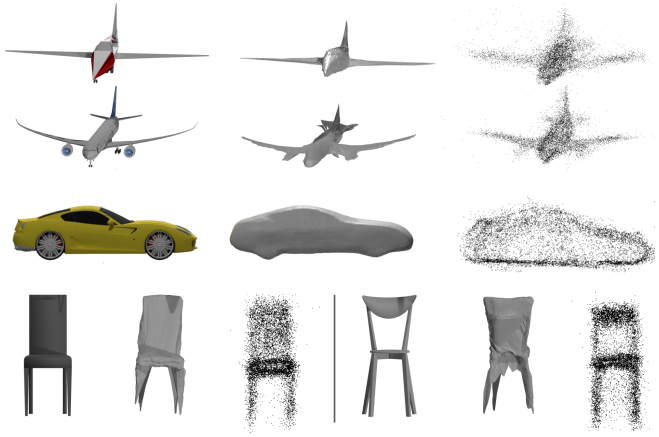

Fig. 3: **Qualitative results.** Ground truth (left), our reconstruction mesh (middle) and XNOCS output points (right).

observe that creating a mesh from the output point clouds may be difficult, while our approach produces meshes directly.

4.3. Ablation study

The results of an ablation study are presented in Table 2.

Impact of the optimization stage. We see that the multi-view encoder performs reasonably well, in particular on planes and cars, but it is not sufficient to reach the state of the art. It is however a good prior in the full setting: the model with optimization performs better than the multi-view encoder alone on every categories and metrics (from 16 to 24 points), given as input the estimated shape produced at the initialization step.

Chamfer loss vs gnomonic loss. We compare our $\mathcal{L}_{\text{shape}}$ loss defined with a gnomonic projection of the sphere on $\mathcal{S}$ to the Chamfer distance loss, that uses closest points to establish correspondences [2], which often leads to non-smooth and non-injective mappings with local self-intersections and foldovers. We first optimize the multi-view encoder and the AltlasNet decoder (*MV encoder*) and observe that the two models obtain similar scores. It seems that both losses are suitable for the task. However, taking the next step (*Full model*), we can observe that using the gnomonic projection leads to a higher F1-score. Our interpretation is that this is due the non-smooth surface reconstructed with the Chamfer loss. We show in Figure 4 two meshes reconstructed after MV-encoder training, with Chamfer distance (left) and gnomonic projection (right). The first model seems to better fit the plane shape, but has a lot of self-intersections (each color discontinuity) while the second model is very smooth. This smoothness is, in practice, a better starting point for the reconstruction optimization.

Discussion and limitations. While our approach outperforms the baselines, we found several limitations. First, while the gnomonic projection outperforms the Chamfer loss, there are still remaining reconstruction artifacts when mapping from a sphere to a target shape to due the lack of bijectivity; most shapes are not star-shaped volumes, even simple cars. Second, the spherical template shape restricts the outputs to the set of genus-zero shapes, which does not allow us to reconstruct well shapes with arbitrary topologies, which is the case of many chairs (cf. Figure 3). In fact, both issues are present in chairs.

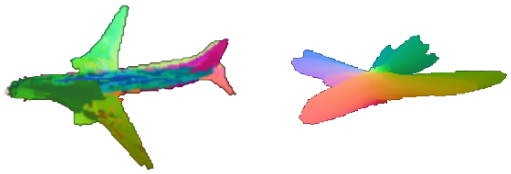

Fig. 4: **Visualization** of learned surface parameterization with a (left) Chamfer loss and (right) gnomonic mapping loss. Discontinuities in colors indicate self-intersections.

Table 1: **Multi-view shape reconstruction on the ShapeNetCOCO [6] validation set.** We report F1-score (F1) and squared-symmetric 100xChamfer distance (CD), averaged over each class.

Method	Airplane F1↑	Airplane CD↓	Car F1↑	Car CD↓	Chair F1↑	Chair CD↓	Avg. F1↑	Avg. CD↓
Single template	36.7	0.42	23.3	0.53	13.3	1.04	24.4	0.66
XNOCS [6]	**62.3**	0.08	33.2	0.19	**27.2**	**0.20**	40.9	0.16
Our full model	56.3	**0.04**	**41.7**	**0.07**	25.5	0.28	**41.2**	**0.13**

Table 2: **Ablation study.** Comparison of multi-view encoder and full model for Chamfer and gnomonic loss. †: AtlasNet (single view) makes inference in canonical coordinates as opposed to camera coordinates for our method.

Method	Loss	Airpl. F1↑	Car F1↑	Chair F1↑	Avg. F1↑
AtlasNet†	Chamfer	38.6	20.6	5.8	21.7
MV encoder	Chamfer	33.0	14.6	14.9	20.8
MV encoder	Gnomonic	32.9	24.2	9.2	22.1
Full model	MV Chamfer	49.7	34.2	24.3	36.1
Full model	MV Gnomonic	**56.3**	**41.7**	**25.5**	**41.2**

5. CONCLUSION

We have presented a method to learn a dense UVW mapping of a parameterizable shape template. We introduced a multi-view loss to resolve ambiguities inherent to single image shape and UVW prediction, and showed that our method outperforms a recent state-of-the-art approach. In addition, our method is able to reconstruct a mesh, rather than a point cloud, which makes it more useful for downstream 3D applications.

A possible followup is to use UVWs predicted from multiple images along with traditional projection-based techniques to optimize for mesh textures.

6. REFERENCES

[1] Steven M Seitz, Brian Curless, James Diebel, Daniel Scharstein, and Richard Szeliski, "A comparison and evaluation of multi-view stereo reconstruction algorithms," in *CVPR*. IEEE, 2006, vol. 1, pp. 519–528.

[2] Thibault Groueix, Matthew Fisher, Vladimir G. Kim, Bryan Russell, and Mathieu Aubry, "AtlasNet: A papier-mâché approach to learning 3D surface generation," in *CVPR*, 2018.

[3] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove, "DeepSDF: Learning continuous signed distance functions for shape representation," in *CVPR*, June 2019.

[4] Chen-Hsuan Lin, Oliver Wang, Bryan C Russell, Eli Shechtman, Vladimir G Kim, Matthew Fisher, and Simon Lucey, "Photometric mesh optimization for video-aligned 3D object reconstruction," in *CVPR*, 2019.

[5] Nilesh Kulkarni, Abhinav Gupta, and Shubham Tulsiani, "Canonical surface mapping via geometric cycle consistency," in *ICCV*, 2019.

[6] Srinath Sridhar, Davis Rempe, Julien Valentin, Sofien Bouaziz, and Leonidas J. Guibas, "Multiview aggregation for learning category-specific shape reconstruction," in *NeurIPS*, 2019.

[7] Ashutosh Saxena, Min Sun, and Andrew Y. Ng, "Make3D: Learning 3D scene structure from a single still image," *IEEE TPAMI*, vol. 31, no. 5, pp. 824–840, May 2009.

[8] David Eigen, Christian Puhrsch, and Rob Fergus, "Depth map prediction from a single image using a multi-scale deep network," in *NeurIPS*, 2014, pp. 2366–2374.

[9] Angjoo Kanazawa, Shubham Tulsiani, Alexei A. Efros, and Jitendra Malik, "Learning category-specific mesh reconstruction from image collections," in *ECCV*, 2018.

[10] Nilesh Kulkarni, Abhinav Gupta, David Fouhey, and Shubham Tulsiani, "Articulation-aware canonical surface mapping," in *CVPR*, 2020.

[11] Abhishek Kar, Christian Häne, and Jitendra Malik, "Learning a multi-view stereo machine," in *NeurIPS*, 2017, pp. 365–376.

[12] Po-Han Huang, Kevin Matzen, Johannes Kopf, Narendra Ahuja, and Jia-Bin Huang, "DeepMVS: Learning multi-view stereopsis," in *CVPR*, June 2018.

[13] Chao Wen, Yinda Zhang, Zhuwen Li, and Yanwei Fu, "Pixel2Mesh++: Multi-view 3D mesh generation via deformation," in *ICCV*, 2019, pp. 1042–1051.

[14] Vincent Sitzmann, Justus Thies, Felix Heide, Matthias Nießner, Gordon Wetzstein, and Michael Zollhofer, "DeepVoxels: Learning persistent 3D feature embeddings," in *CVPR*, 2019, pp. 2437–2446.

[15] Stephen Lombardi, Tomas Simon, Jason Saragih, Gabriel Schwartz, Andreas Lehrmann, and Yaser Sheikh, "Neural volumes: Learning dynamic renderable volumes from images," *ACM Transactions on Graphics (TOG)*, vol. 38, no. 4, pp. 65:1–65:14, July 2019.

[16] Tinghui Zhou, Richard Tucker, John Flynn, Graham Fyffe, and Noah Snavely, "Stereo magnification: Learning view synthesis using multiplane images," in *SIGGRAPH*, 2018.

[17] Emil Praun and Hugues Hoppe, "Spherical parametrization and remeshing," in *TOG*, 2003.

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *CVPR*, 2016.

[19] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick, "Mask R-CNN," in *ICCV*, 2017.

[20] Olaf Ronneberger, Philipp Fischer, and Thomas Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI)*. Springer, 2015, pp. 234–241.

[21] Angel X. Chang, Thomas A. Funkhouser, Leonidas J. Guibas, Pat Hanrahan, Qi-Xing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu, "ShapeNet: An information-rich 3D model repository," *ArXiv*, vol. abs/1512.03012, 2015.

[22] 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 *Proceedings of the European Conference on Computer Vision (ECCV)*, 2016.

[23] Arno Knapitsch, Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun, "Tanks and Temples: Benchmarking large-scale scene reconstruction," *ACM Transactions on Graphics (TOG)*, vol. 36, no. 4, 2017.

[24] Maxim Tatarchenko, Stephan Richter, René Ranftl, Zhuwen Li, Vladlen Koltun, and Thomas Brox, "What do single-view 3D reconstruction networks learn?," in *CVPR*, 2019.