Supplementary Material
Topologically-Aware Deformation Fields for Single-View 3D Reconstruction

Shivam Duggal Deepak Pathak
Carnegie Mellon University

Appendix

We first provide additional experimental details, which include dataset details, implementation details and baselines' descriptions. Next, we provide additional experimental results in terms of further qualitative and quantitative analysis. Finally, we discuss the potential future works.

1. Additional Experimental Details

1.1. Dataset Details

We performed experiments on the following datasets: synthetic Shapenet [1], real-world Pascal 3D+ [17], real-world CUB-200-2011 [16] and real-world Pix3D chairs [13]:

Shapenet: We used car, planes and chairs category of the Shapenet v2 dataset [1] for our experiments. Shapenet Image dataset is generated by rendering the synthetic CAD models from different sampled viewpoints. Shapenet CAD models are associated with texture files (generally possessing only diffuse texture component). For all experiments, we followed SDF-SRN's data settings (mentioned in their paper section 4.1 and Appendix section B.1). To summarize: we used *2830 training, 809 validation and 405 test CAD model for airplanes, 2465 training, 359 validation and 690 test CAD models for cars category and 4744 training, 678 validation and 1356 test CAD objects for chairs category.*

Pascal3D+: Pascal3D+ [17] is a dataset of real-world camera images with annotated 3D CAD models. Compared to Shapenet, Pascal3D+ data is challenging as camera images are captured in real-world scenarios with variable lightning conditions, variable object occlusions, diverse object textures etc. We follow SDF-SRN's data settings for Pascal3D+ dataset (mentioned in their paper [10] section 4.2 and appendix section B.2). To summarize the data-splits: *we used 991 training and 974 validation examples for airplane category, 2847 training and 2777 validation examples for cars category, and 539 training and 514 validation examples for chairs category.* The object silhouettes used both during training and test phases to mask out the foreground object RGB from the background are generated by rendering a fixed set of CAD models. *Since the same CAD models are used to generate object silhouettes both during training and testing, the dataset posseses a bias (highlighted in Tulsiani* et al. *[15] Appendix A2.2).* While prior work [2, 10] showcase generalization results only on this biased dataset, we address this issue by using silhouette masks generated by an off-the-shelf instance segmentation network [9] as done by Tulsiani *et al*. [15]. Results on the unbiased Pascal3D+ planes dataset are shown in Figure 13. For results on unbiased chairs dataset, refer to Pix3D results (Figure 18, 19).

Pix3D chairs dataset: Similar to Pascal3D+ [17] dataset, Pix3D dataset [13] is a real-world dataset, containing 2D image to 3D CAD model mappings. However, unlike Pascal3D+ dataset, (a) the 3D CAD models align better with the 2D images, (b) different set of CAD models are used for training and test set images, therefore the dataset is unbiased. Some images of the Pix3D chairs dataset are highly occluded/ truncated. We removed such images using the annotated truncation tag associated with each image and also by manual filtering. Overall, we used 2196 Pix3D chair images for training and 637 chair images for test. Since the overall dataset is significantly small (compared to tens of thousands of images in shapenet dataset), we augment Pix3D training set with the 539 Pascal3D training chair images. To highlight, *this dataset is still significantly smaller* considering the amount of variations (in terms of light, material, texture) present in the real-world image collection. Also, since each CAD model is rendered at multiple viewpoints to generate a 2D-3D mapping, the overall 3D information used to train the reconstruction networks is much smaller.

CUB-200-2011 dataset: We used the annotated CUBS dataset released alongside the CMR [8] codebase. Overall, the training set has 5964 images and the test set has 2874 images. Each image is associated with a silhouette map and a weak-perspective camera pose generated using 2D annotated keypoints and SFM registration of the keypoints. Please refer to CMR [8] (section 3.1) for more details.

1.2. Implementation Details

Our deformable reconstruction pipeline (as shown in paper Figure 2) consists of *DeformNet*, *Canonical Shape Generator* and (an LSTM-based) *Differentiable Renderer* modules. We need to ensure that each module performs its desired task, despite the lack of explicit supervision. Simply jointly training the three modules fails to ensure that, and hence results in poor performance. In order to

effectively train these modules, we follow a curriculum learning strategy. We split the training phase into two main stages and two intermediate pre-training stages:

In Stage 1, we directly learn to reconstruct the 3D shape (in form of signed-distance field) given the corresponding input image captured from a known input viewpoint. We adopt SDF-SRN [10] for this task and *train the shape generator module* [1]*, image encoder and the differentiable renderer module in this stage*. Given an input image, we first map it to a latent-code using Imagenet pre-trained Resnet encoder [7]. The latent-code is used by a hyper-network to generate the weights of the shape generator module. The shape generator module then learns to map any 3D point in the object space to its corresponding SDF value. The differentiable renderer module is also trained alongside to render the learned geometry from the given input viewpoint. It is an LSTM module which takes as input the intermediate-level features from the shape generator module (corresponding to the sampled 3D point) and predicts the ray marching step along the input ray direction. Using the 3D point and the ray-marching step, the next point along the input ray direction is generated. The above mentioned procedure is then repeated for a fixed number of ray-marching steps. In order to ensure that shape generator module can operate on higher-dimensional input (3D point + point features) in Stage 2, we additionally pass "un-conditioned" point-features as input to the shape generator module. These point-features (4-dimensional in our experiments) are generated by simply passing the input 3D points through a two-layer MLP (which is not conditioned on the input-image). Instead of passing the 3D points plus the point features to the shape generator network, we pass the concatenation of 3D points, their position encoding and the positional encoding of the point features as input to the network.

In Stage 2, *we train the DeformNet module, while fine-tuning the image encoder, shapenet generator module and the differentiable render.* The DeformNet module takes as input a 3D point in the object space and maps it to a higher-dimensional (7-dimensional in our experiments) canonical point (3D point deformation + 4D object-space point features) using the learned higher dimensional deformation field. Alongside predicting the deformation field, it also predicts the view-independent RGB value for the input 3D point. Next, given the higher-dimensional canonical point, the shape generator module learns to predict the corresponding SDF value. In stage 2, the shape generator module only focuses on reconstructing the canonical 3D shape, and hence is termed as the canonical shape generator. The differentiable renderer in stage 2 takes as input the object-space features of the sampled 3D point (sampled along the input ray) as learned by the DeformNet. Like stage 1, weights of both DeformNet and Canonical Shape Generator are learned through hyper-networks. Unlike Stage 1, where the hyper-network for the shape generator is conditioned on the input image latent-code, the weights of the canonical shape generator are predicted by a hyper-network conditioned on a canonical shape latent-code (which is optimized jointly). The canonical shape-latent code is initialized by the mean of all training images' latent codes predicted in Stage 1. Like the shape generator, the input to the Deformnet is the concatenation of the 3D point and its positional encoding.

[1] In Stage 1, the shape generator module is trained to reconstruct any 3D shape given the corresponding camera image. In stage 2, this shape generator module is fine-tuned for reconstructing only the canonical shape and is thus termed as Canonical Shape Generator.

Pre-training phases: Following SDF-SRN [10], prior to Stage 1 training, we first pre-train the shape reconstruction module to predict the SDF-space of a zero centered 3D sphere (conditioned on random latent code in place of image-based latent code used in Stage 1). This helps the network better learn the 3D object signed-distance fields in stage 1. Prior to Stage 2 (and post stage 1 training), we overfit the DeformNet module to deform points belonging to the initial canonical space (SDF space generated using initial canonical shape-latent code and the pre-trained shape generator module) to a 3D sphere, such that SDF of the initial point in the canonical space is equal to the SDF of the deformed point w.r.t the 3D sphere.

Architecture details: We now provide the architecture details for the three modules: The shape generator module is implemented as an MLP with two output heads, one used to predict the SDF value for the input 3D point and the other used to predict the point's RGB value (during stage 1). The shared MLP backbone between the two output heads has multiple linear layers with LayerNorm and ReLU activation, while the output heads are just linear layers. The weights and the biases for each layer are generator by different hyper-networks, which themselves are MLPs. The high-level architecture is adopted from SDF-SRN [10]. The architecture for the Deformnet is similar to the shape generator module, with three output heads learning 3D point deformation, 4D point features and the RGB value for each input 3D point. For the LSTM module of the differentiable renderer [12], we kept the output and the hidden state dimension to be 32.

1.3. Baselines

SDF-SRN [10]: We directly used the open-sourced code-base and pretrained models of SDF-SRN [10] to generate the results. Note, the open-sourced pre-trained Pascal3D+ models belong to the default biased Pascal3D+ dataset.

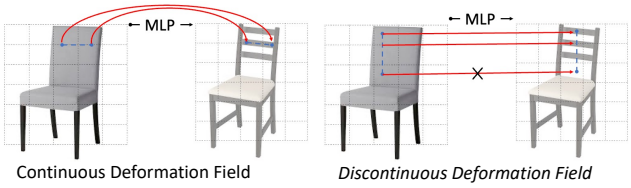

Figure 9. **Discontinuities in a deformation field:** (Left) Example of continuous mapping of a set of 3D points from source chair to target chair. (Right) Example of dis-continuous mapping from source to target chair.

SoftRas [11]: We used the released codebase and trained the SoftRas shape reconstruction model on the Shapenet dataset. For fair comparison, we commented out the multi-view consistency loss in the open-source implementation and rather rendered the reconstructed mesh only at the source viewpoint to supervise the training pipeline.

CMR [8]: We used the released codebase for training CMR [8] on Pascal3D+ dataset. For fair comparison, we did not use the key-point loss and only used the RGB and the silhouette loss for supervising the pipeline. Prior to training, we initialized their mean shape prior using the 3D mesh template shared alongside the CMR codebase. For chairs category, because of the large intra-category topological variations, we found out that using a template mesh (which was not isomorphic to sphere) for initialization of the mean shape leads to poor training and hence poor reconstructions. Therefore, (following SDF-SRN [10]) we used a 3D sphere to initialize the mean shape for Pascal3D+ chairs reconstruction task.

1.4. Metrics

We used symmetric Chamfer distance (CD) and Earth Mover's distance (EMD) to quantitatively measure the fidelity of the reconstructed meshes. CD is defined as the sum of squared distance of each 3D point on the ground-truth shape ($\mathbf{X}$) to the closest surface point on the reconstructed shape ($\mathbf{Y}$) and vice-versa.

$$\text{CD}(\mathbf{X},\mathbf{Y}) = \frac{1}{2|\mathbf{X}|}\sum_{x\in\mathbf{X}}\min_{y\in\mathbf{Y}}\|x-y\|_2 + \frac{1}{2|\mathbf{Y}|}\sum_{y\in\mathbf{Y}}\min_{x\in\mathbf{X}}\|x-y\|_2$$

EMD is defined as the sum of the squared distance of each point in the GT point cloud ($\mathbf{X}$) to its bijective mapping in the reconstructed point cloud.

$$\text{EMD}(\mathbf{X},\mathbf{Y}) = \min_{\phi:X\to Y}\sum_{x\in X}\|x-\phi(x)\|_2$$

We also used Precision and Recall as robust alternatives [14] to chamfer distance.

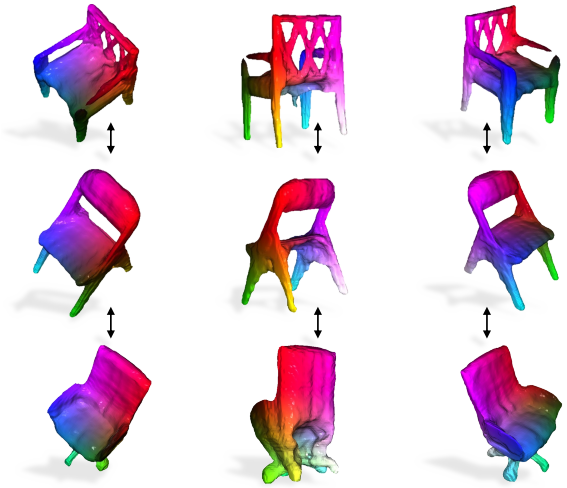

Figure 10. **Visualization - Topologically-aware deformation field:** The learned deformation fields map similar structures/ parts of different object instances to similar canonical space regions.

$$\text{Precision}(\mathbf{X},\mathbf{Y}) = \frac{1}{|\mathbf{Y}|}\sum_{y\in\mathbf{Y}}\left[\min_{x\in\mathbf{X}}\|x-y\|_2 <= t\right]$$

$$\text{Recall}(\mathbf{X},\mathbf{Y}) = \frac{1}{|\mathbf{X}|}\sum_{x\in\mathbf{X}}\left[\min_{y\in\mathbf{Y}}\|x-y\|_2 <= t\right]$$

We set the true-positive threshold to 0.1.

2. Analysis: Topologically-Aware Deformation Field

Figure 9 motivates the need of the topologically-aware deformation field. Our approach reconstructs the target shape by mapping 3D points from the target space to the source space using the learned deformation field. Since the deformation field is learned implicitly using an MLP, the inductive continuous nature of the MLP would deform 3D points continuously from the target space to the source space. Thus such a deformation field can truly reconstruct the target shape from the source shape only when both the shapes are of similar topologies (Figure 9 left). To over come this topological restriction, we take inspiration from Level Set Methods and learn additional per-point features which potentially guide the network on how to modify the deformation field to truly reconstruct the target shape from the source.

3. Additional Experimental Results

3.1. Qualitative Analysis

Single-View 3D Reconstruction on Shapenet: Figure 15, Figure 16 and Figure 17 compare our proposed approach

with SDF-SRN [10] baseline on the Shapenet dataset's cars, planes and chairs category respectively. Compared to the mesh-based baseline (SoftRasterizer [11], as shown in main-paper Figure 6) both our proposed approach and SDF-SRN [10] perform significantly well, thanks to the use to neural implicit modeling. Thanks to the inherently learned category-specific structural priors (inherent property of deformable models) our shapes have fewer artifacts compared to SDF-SRN's shapes *(see SDF-SRN's noisy reconstructions in row 1 chair 1, row 2 chair 2(sofa), row 5 chair 1 of Figure 16)*.

Single-View 3D Reconstruction on CUBS-200-2011: Figure 11 and Figure 12 showcase additional qualitative comparisons between SDF-SRN [10] and our approach on the CUBS-200-2011 dataset. Our approach showcase consistent improvement over prior works in terms of (a) less-noisy reconstructions (row 3 right, row 4 left), (b) better articulations (row 2 left, row 4 right) and (c) better capture of overall geometric structure (reconstructed beaks, foots, legs etc).

Single-View 3D Reconstruction on Pascal3D+ (default): Figure 20, Figure 21 and Figure 22 compare our proposed approach with SDF-SRN [10] baseline on the Pascal3D+ dataset's cars, planes and chairs category respectively. Compared to Shapenet, Pascal3D+ is a challenging real-world dataset with object images having diverse textures, captured under variable environmental (lightning) conditions and under variable nature of object occlusion. As a result, SDF-SRN reconstructions on Pascal3D+ are much noisier compared to their results on Shapenet *(see ripples on car surfaces in Figure 20 and noisy reconstructed chairs in Figure 22)*. By learning to deform all object instances to a particular category-level canonical shape, we are able to regularize the reconstruction procedure and hence generate smoother shapes with much fewer artifacts. Moreover, compared to SDF-SRN [10], our reconstructions are able to capture the finer shape details captured in the input image *(eg: reconstructed front wheel on planes in row 1, row 2 and row 4, reconstructed plane propeller in row 6 of Figure 21)*. Our shapes also maintain the topological details of the GT shape underlying the input image *(see chairs in Figure 22)*.

Single-View 3D Reconstruction on Pascal3D+ (unbiased) planes: Figure 13 and Figure 14 showcases additional results on the unbiased Pascal3D+ planes dataset. While, in comparison to the reconstructed planes of the default Pascal3D dataset, the unbiased Pascal3D reconstructions are of comparatively low fidelity, our approach still demonstrates significant improvement (in terms of less noisy reconstructions with better overall 3D structure) over the prior state of the art works of CMR [8] (as shown in paper Figure 3) and SDF-SRN [10]. The last two rows of Figure 13 and Figure 14 showcase the examples where the input images (captured at the specific input viewpoints) do not provide enough geometric cues to the reconstruction pipeline to enable high-fidelity reconstruction. *The significantly smaller size of the Pascal3D+ planes dataset is potentially the core reason behind such failures.*

Method	# training examples	Chamfer ↓ acc.	cov.	overall
SDF-SRN [10]	500	**0.475**	0.422	**0.448**
	1000	**0.442**	0.385	**0.413**
	2000	0.423	0.349	0.386
TARS (ours)	500	0.495	**0.402**	**0.448**
	1000	0.462	**0.366**	0.414
	2000	**0.423**	**0.347**	**0.385**

Table 3. **Dataset size ablation:** # training examples vs reconstruction metrics.

Method	Implicit 3D Shape	Dense Correspondences	Chamfer ↓ acc.	cov.	overall
SDF-SRN [10]	✓		**0.352**	0.315	0.333
DIT [18]	✓	✓	0.386	0.326	0.356
DIF [4]	✓	✓	0.376	0.0308	0.342
TARS (ours)	✓	✓	0.353	**0.312**	**0.332**

Table 4. **Comparison with Deformable Implicit Reconstruction approaches on ShapeNet Chairs dataset**

Single-View 3D Reconstruction on Pix3D Chairs: We showcase additional qualitative comparison on the Pix3D chairs dataset in Figure 18 and Figure 19. Figure 18 highlights the synthetic to real generalization capability (trained on Shapenet, tested of Pix3D) of the reconstruction approaches. For Figure 19, we trained both our approach and SDF-SRN [10] on the combined Pascal3D+ and Pix3D train dataset. While the results are much noisier (potentially because of smaller but challenging training set), the reconstructed shapes capture the overall geometry of the GT shapes and also maintain the topological structures of the GT chairs the majority of the times *(see row 1, row 3, row 5 of Figure 19)*. Like Pascal3D planes, the failure cases for the Pix3D chairs occur usually for input observations captured at some particular camera viewpoints which do not provide the reconstruction pipeline with enough geometric cues.

3.2. Quantitative Analysis

Dataset size ablation: We ablate the performance of our proposed approach as a factor of number of training examples on the Shapenet chairs dataset. For all the experiments under this ablation, we randomly sample a subset of CAD models. The training data is then generated by rendering each CAD model at only one randomly sampled viewpoint. From Table 3, we see that both our proposed approach and SDF-SRN [10] consistently performs well on all subsets of the Shapenet chairs dataset. Furthermore, increase in the dataset size does help the model achieve higher shape fidelity (in terms of reconstruction metrics). To re-emphasize on the need of larger training datasets: we think that compar-

atively less fidelity of the reconstructed real-world shapes (Pascal3D+, Pix3D) is because of the large variations (textural, environmental lightning, structural) in the real-world objects, but much smaller training datasets.

Comparison with Deformable Implicit Reconstruction approaches on Shapenet chairs dataset: Recently, Zheng *et al*. [18] and Deng *et al*. [4] learned category-specific deformation fields and signed-distance fields jointly. While Zheng *et al*. [18] *(DIT: Deep Implicit Templates)* learned a 3D deformation field, Deng *et al*. [4] *(DIF: Deformed Implicit Fields)* learned a 3D deformation field + SDF correction field to handle the topological variations. Both the approaches required dense 3D supervision during training. *In comparison to them, we address the task of single-view 3D reconstruction by learning higher-dimensional* topologically-aware *deformation fields without using any form of dense supervision (multi-view images or dense 3D)*. In Table 4, we compare single-view analogs of their proposed approaches. We trained two ablations of our proposed approach: (a) single-view reconstruction using only 3D deformation fields (similar to the MLP-based deformation approach of Zheng *et al*. [18]), (b) single-view 3D reconstruction using 3D deformation fields and 3D SDF correction fields (similar to Deng *et al*. [4]). For both ablations, we do not learn any additional point features. We trained both the ablations using only single-view supervision exactly same as our proposed approach. From Table 4, we can see that solely learning deformation fields results in drop of the reconstruction metric (chamfer) compared to the reconstruction only approach (SDF-SRN). Adding SDF-correction fields on top of 3D deformation fields does ease the task of implicit deformation estimation and hence leads to better reconstructions. Our proposed approach performs better than both the deformation based implicit reconstruction approaches and is also at par with reconstruction only SDF-SRN approach *(while learning deformations for free)*.

4. Future Work

While, overall our results represent an encouraging step towards generalization of reconstruction systems to the internet of images, there is still more future work to be done to achieve scalable generalization. The immediate next step to unlock internet generalization is the removal of the requirement of known poses during training. Other directions could be the exploration of joint learning among multiple object categories, and efficient incorporation of adversarial learning to enable high-fidelity reconstruction even from input images captured at challenging viewpoints. Also, as we have witnessed the role of large annotated 2D datasets (like Imagenet [3]) in rise of self-supervised learning in 2D [5, 6], any potential work of generating much larger unbiased datasets like Pix3D could turn out to be a major step towards scalable single-view reconstruction.

5. Statement on Potential Negative Impact

We feel that the field of single-view 3D reconstruction is still in its nascent stage. So, we do not think this work has any immediate potential negative impact.

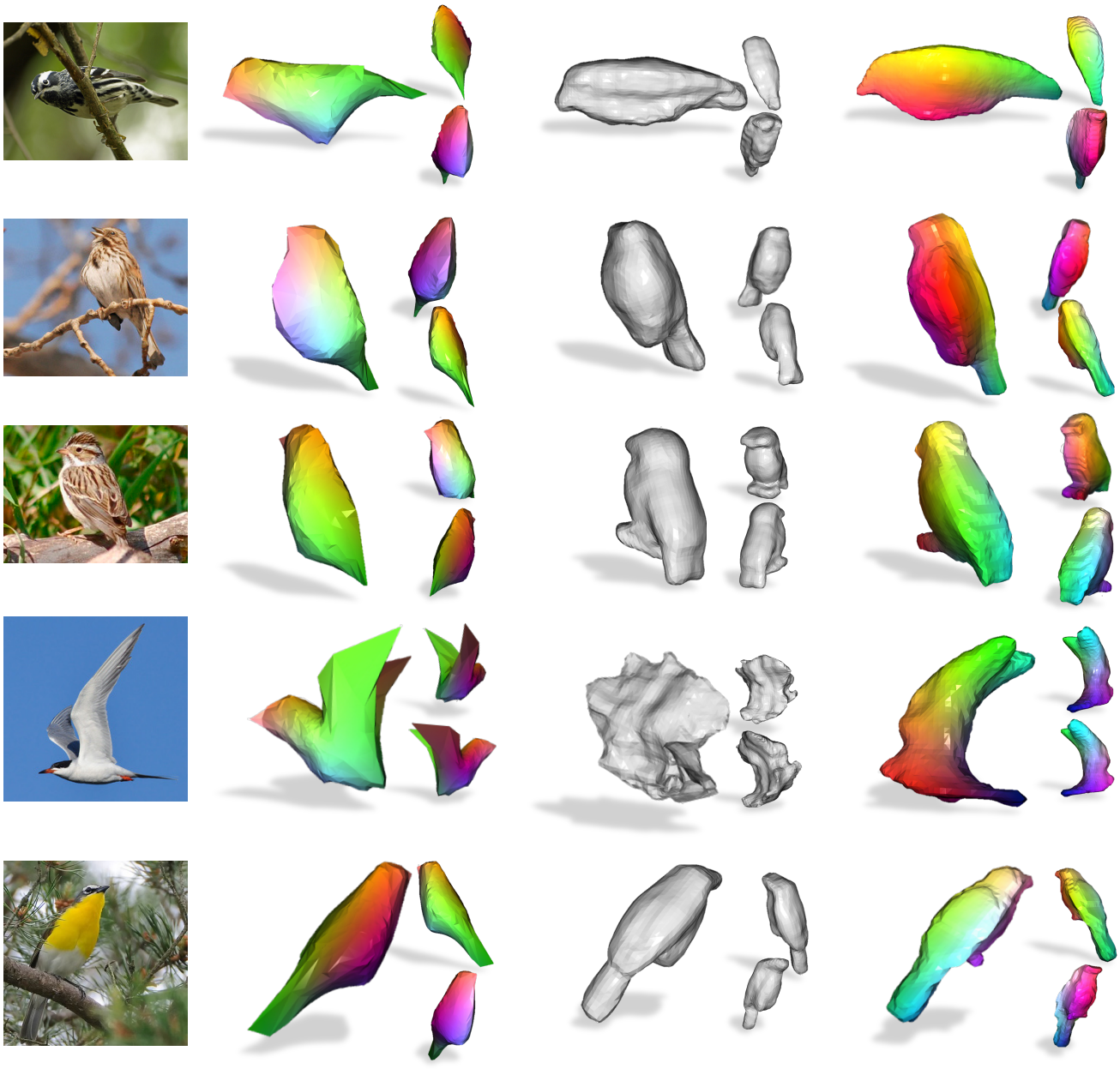

Figure 11. **3D Reconstruction on CUB-200-2011 from Single 2D Image**

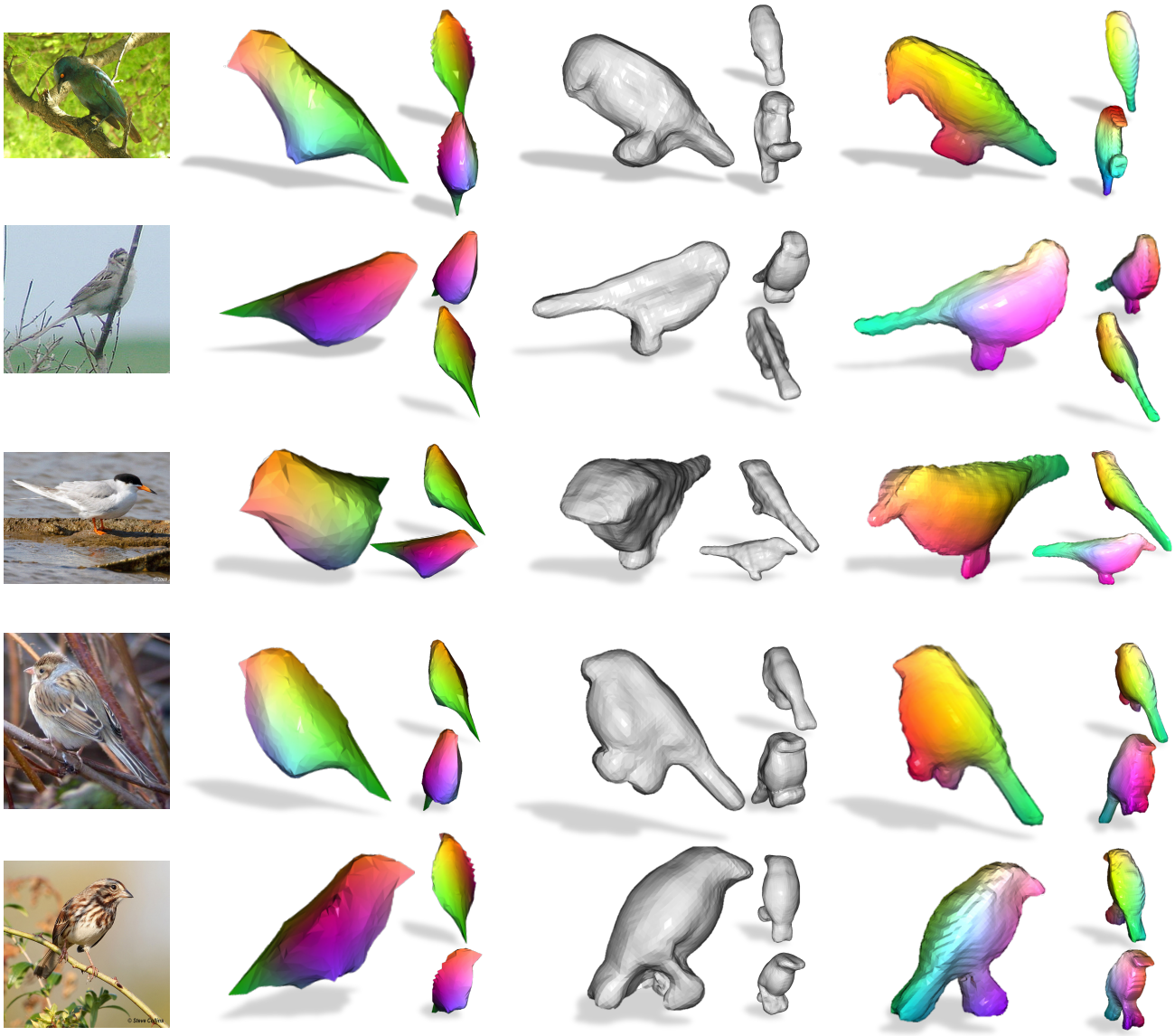

Figure 12. **3D Reconstruction on CUB-200-2011 from Single 2D Image**

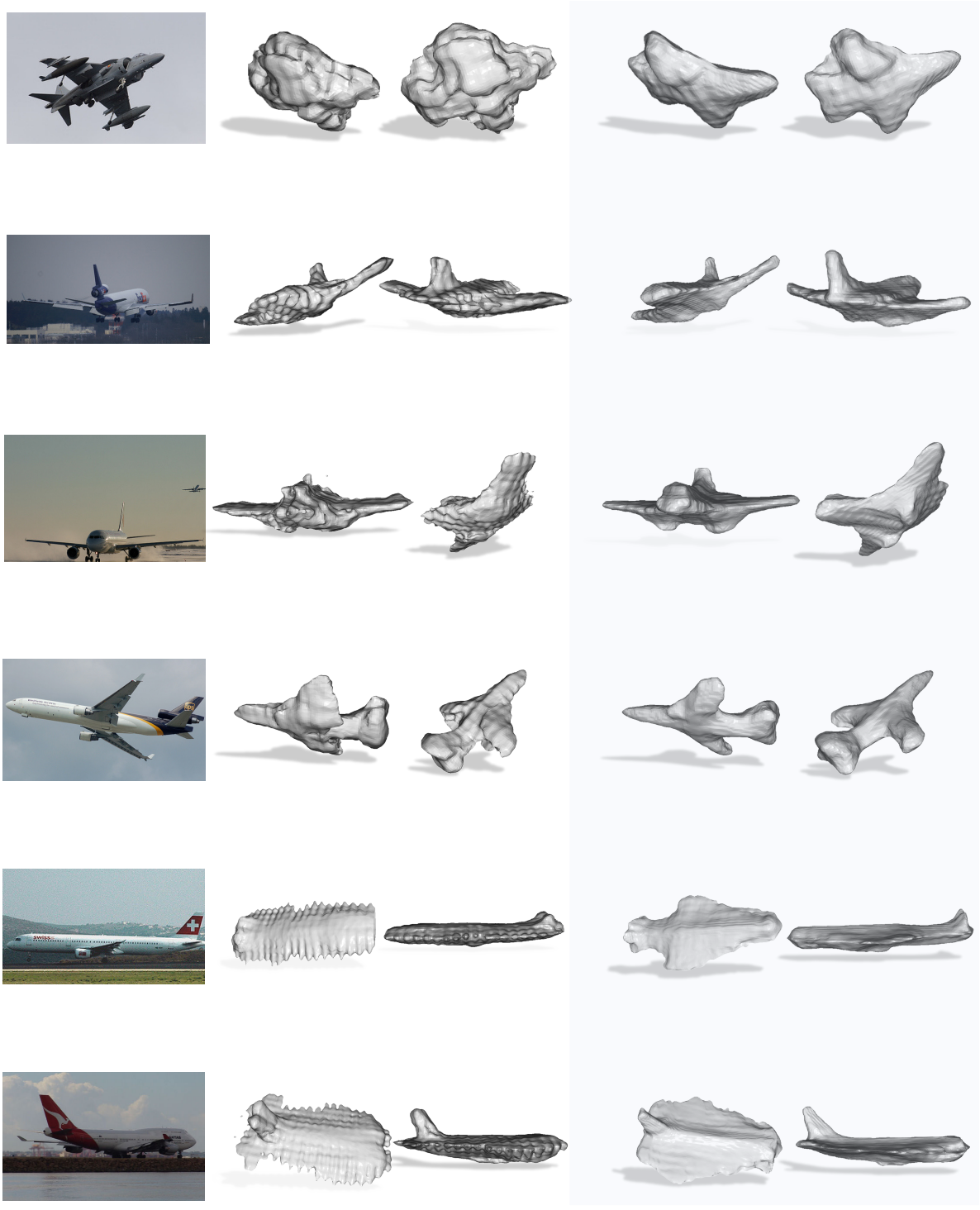

Figure 13. **3D Reconstruction on Pascal3D+ (unbiased) Airplanes from Single 2D Image**

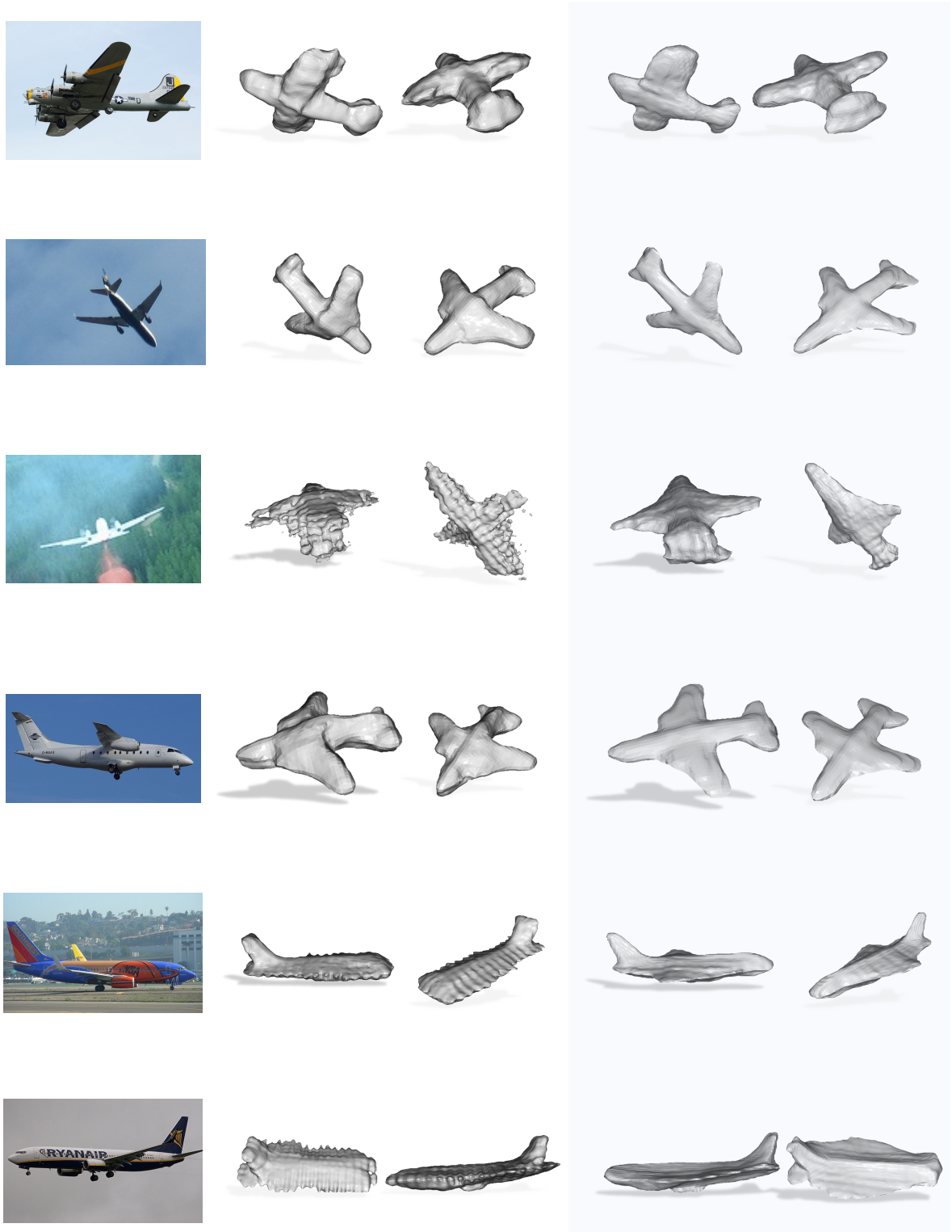

Figure 14. **3D Reconstruction on Pascal3D+ (unbiased) Airplanes from Single 2D Image**

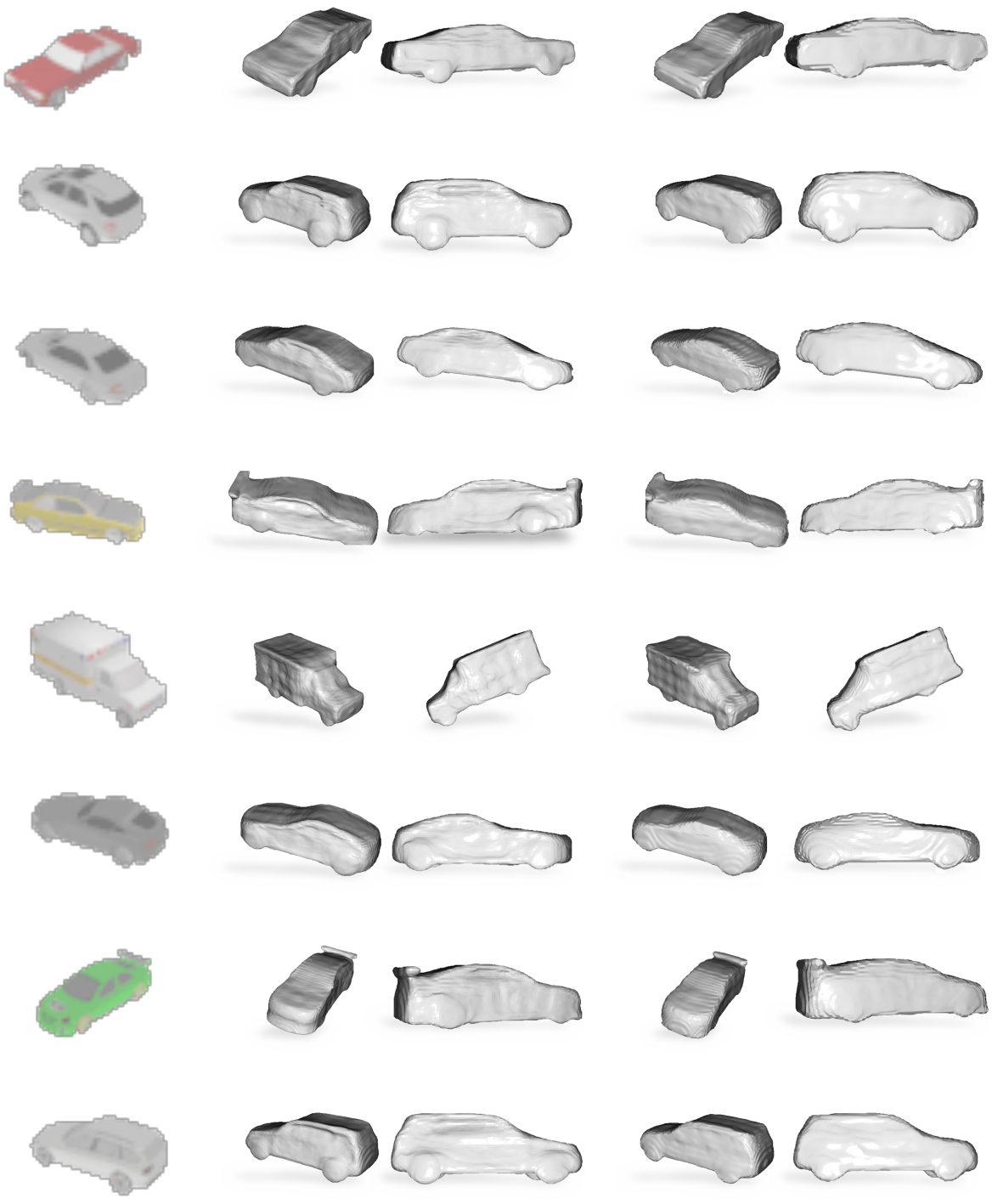

Figure 15. **3D Reconstruction on Shapenet Cars from Single 2D Image** Our approach amtches the shape fidelity of SDF-SRN while leveraging cross-instance correspondences for free.

Figure 16. **3D Reconstruction on Shapenet Chairs from Single 2D Image** As can be seen, our reconstructions are less noiser, thanks to the learned deformation field which acts as a regularizer.

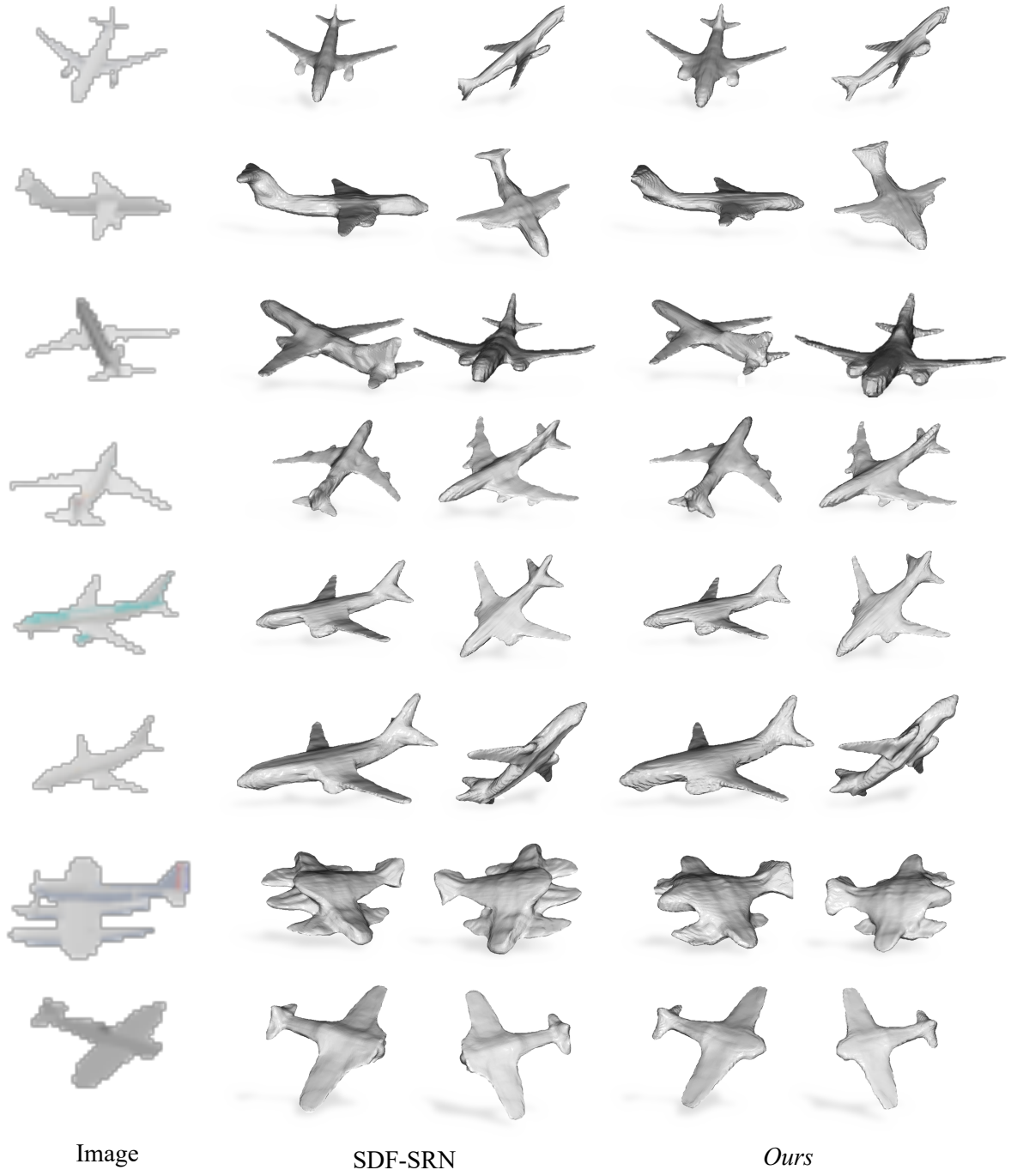

Figure 17. **3D Reconstruction on Shapenet airplanes from Single 2D Image**

Figure 18. **3D Reconstruction on Pix3D Chairs from Single 2D Image (trained on Shapenet, tested on Pix3D val)**

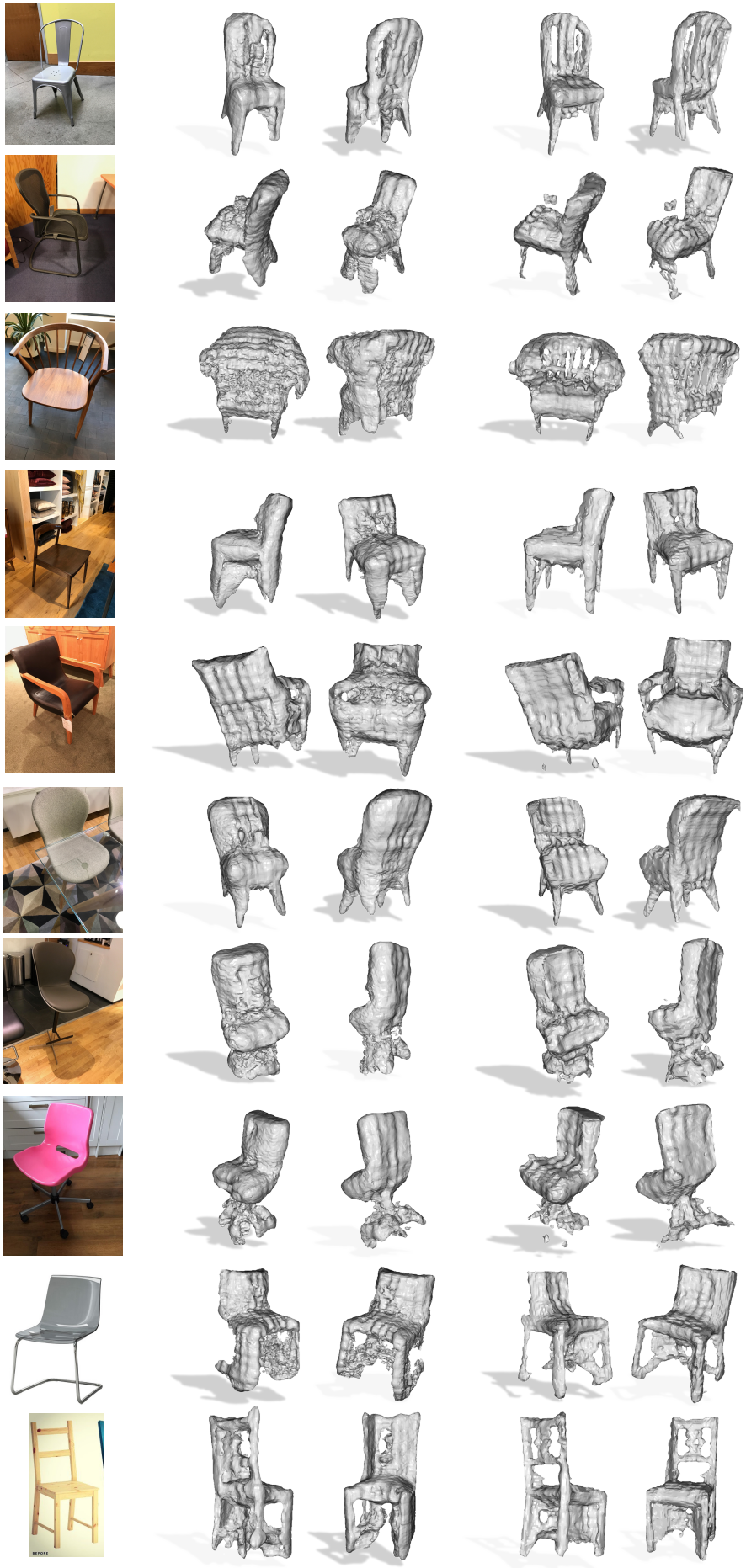

Figure 19. **3D Reconstruction on Pix3D Chairs (trained on Pix3D train + Pascal3D chairs, tested on Pix3D val)**

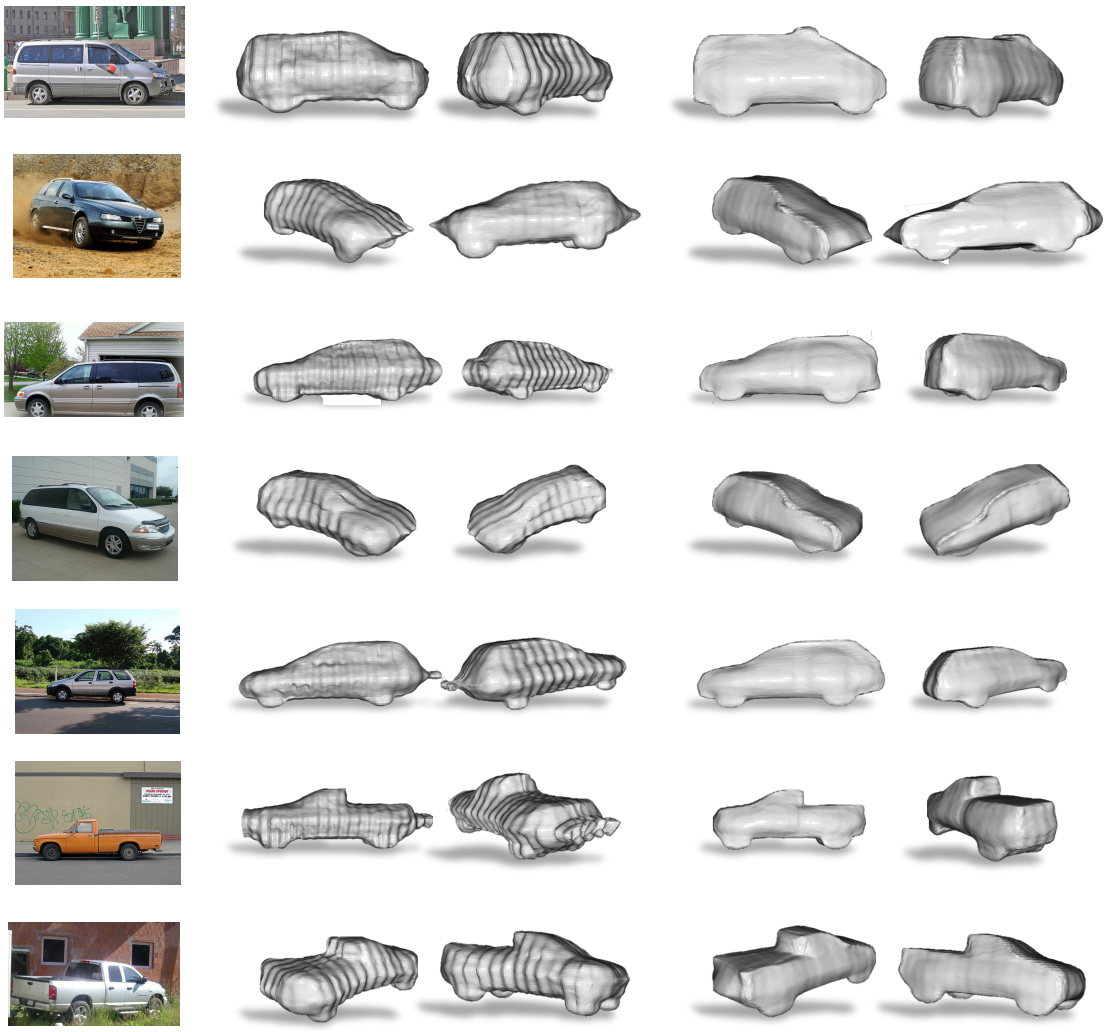

Figure 20. **3D Reconstruction on Pascal3D+ (default) Cars from Single 2D Image**

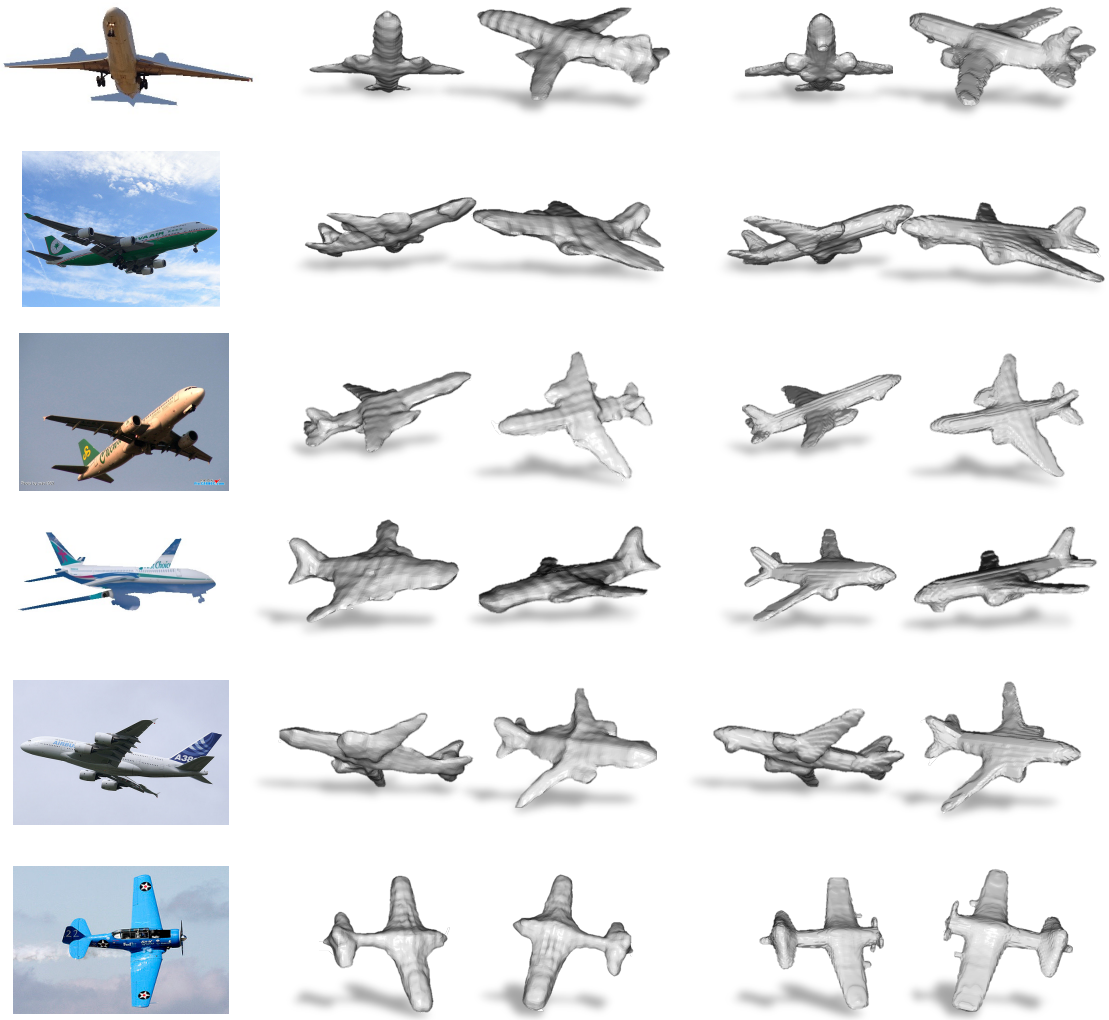

Figure 21. **3D Reconstruction on Pascal3D+ (default) Airplanes from Single 2D Image**

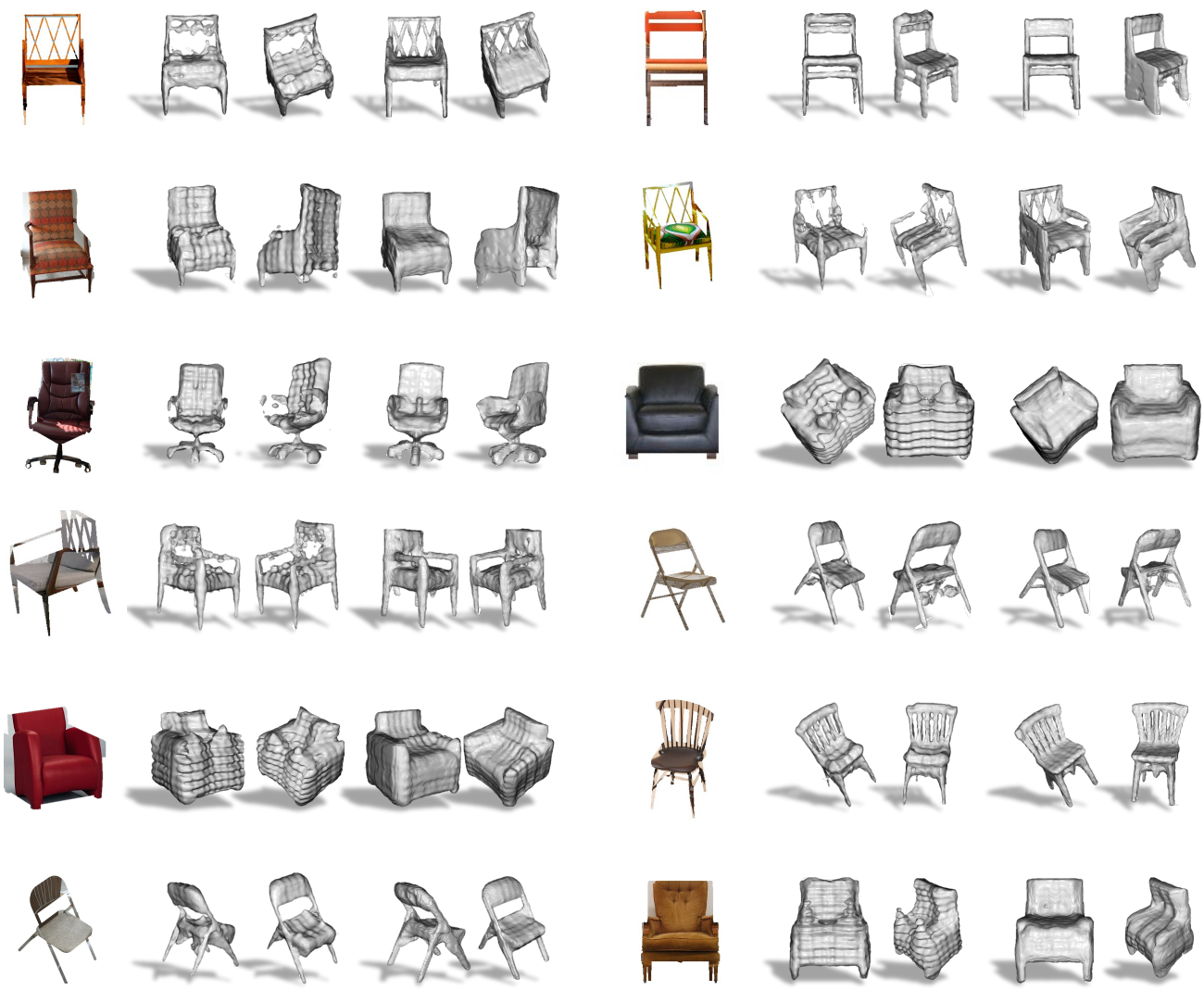

Figure 22. **3D Reconstruction on Pascal3D+ (default) Chairs from Single 2D Image:** Our approach not only yields high fidelity reconstructions, but also provides with dense cross-instance correspondences.

References

[1] Angel X. Chang, Thomas A. Funkhouser, Leonidas J. Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, L. Yi, and Fisher Yu. Shapenet: An information-rich 3d model repository. *ArXiv*, abs/1512.03012, 2015. 1

[2] 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

[3] 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. 5

[4] Yu Deng, Jiaolong Yang, and Xin Tong. Deformed implicit field: Modeling 3d shapes with learned dense correspondence. In *CVPR*, 2021. 4, 5

[5] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Ávila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent: A new approach to self-supervised learning. *CoRR*, abs/2006.07733, 2020. 5

[6] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross B. Girshick. Momentum contrast for unsupervised visual representation learning. *CoRR*, abs/1911.05722, 2019. 5

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385*, 2015. 2

[8] Angjoo Kanazawa, Shubham Tulsiani, Alexei A. Efros, and Jitendra Malik. Learning category-specific mesh reconstruction from image collections. In *ECCV*, 2018. 1, 3, 4

[9] Ke Li, Bharath Hariharan, and Jitendra Malik. Iterative instance segmentation. *CoRR*, abs/1511.08498, 2015. 1

[10] Chen-Hsuan Lin, Chaoyang Wang, and Simon Lucey. Sdf-srn: Learning signed distance 3d object reconstruction from static images. In *NeurIPS*, 2020. 1, 2, 3, 4

[11] Shichen Liu, Tianye Li, Weikai Chen, and Hao Li. Soft rasterizer: A differentiable renderer for image-based 3d reasoning. *ICCV*, Oct 2019. 3, 4

[12] Vincent Sitzmann, Michael Zollhöfer, and Gordon Wetzstein. Scene representation networks: Continuous 3d-structure-aware neural scene representations. In *NeurIPS*, 2019. 2

[13] 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. 1

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

[15] Shubham Tulsiani, Tinghui Zhou, Alexei A. Efros, and Jitendra Malik. Multi-view supervision for single-view reconstruction via differentiable ray consistency. *CoRR*, abs/1704.06254, 2017. 1

[16] P. Welinder, S. Branson, T. Mita, C. Wah, F. Schroff, S. Belongie, and P. Perona. Caltech-UCSD Birds 200. Technical report, California Institute of Technology, 2010. 1

[17] Yu Xiang, Roozbeh Mottaghi, and Silvio Savarese. Beyond pascal: A benchmark for 3d object detection in the wild. In *WACV*, 2014. 1

[18] Zerong Zheng, Tao Yu, Qionghai Dai, and Yebin Liu. Deep implicit templates for 3d shape representation, 2020. 4, 5