Monocular 3D Object Reconstruction with GAN Inversion

Junzhe Zhang[1,3], Daxuan Ren[1,3], Zhongang Cai[1,3], Chai Kiat Yeo[2], Bo Dai[4]*, and Chen Change Loy[1]

[1] S-Lab, Nanyang Technological University
[2] Nanyang Technological University
[3] SenseTime Research
[4] Shanghai AI Laboratory
{junzhe001,daxuan001,caiz0023}@e.ntu.edu.sg,
{asckyeo,ccloy}@ntu.edu.sg, {daibo}@pjlab.org.cn

Abstract. Recovering a textured 3D mesh from a monocular image is highly challenging, particularly for in-the-wild objects that lack 3D ground truths. In this work, we present **MeshInversion**, a novel framework to improve the reconstruction by exploiting the *generative prior* of a 3D GAN pre-trained for 3D textured mesh synthesis. Reconstruction is achieved by searching for a latent space in the 3D GAN that best resembles the target mesh in accordance with the single view observation. Since the pre-trained GAN encapsulates rich 3D semantics in terms of mesh geometry and texture, searching within the GAN manifold thus naturally regularizes the realness and fidelity of the reconstruction. Importantly, such regularization is directly applied in the 3D space, providing crucial guidance of mesh parts that are unobserved in the 2D space. Experiments on standard benchmarks show that our framework obtains faithful 3D reconstructions with consistent geometry and texture across both observed and unobserved parts. Moreover, it generalizes well to meshes that are less commonly seen, such as the extended articulation of deformable objects. Code is released at https://github.com/junzhezhang/mesh-inversion.

1 Introduction

We consider the task of recovering the 3D shape and texture of an object from its monocular observation. A key challenge in this task is the lack of 3D or multi-view supervision due to the prohibitive cost of data collection and annotation for object instances in the wild.

Prior attempts resort to weak supervision based on 2D silhouette annotations of monocular images to solve this task. For instance, Kanazawa *et al*. [19] propose the use of more readily available 2D supervisions including keypoints as the supervision. To further relax the supervision constraint, several follow-up studies propose to learn the 3D manifold in a self-supervised manner, only requiring

* Bo Dai completed this work when he was with S-Lab, NTU.

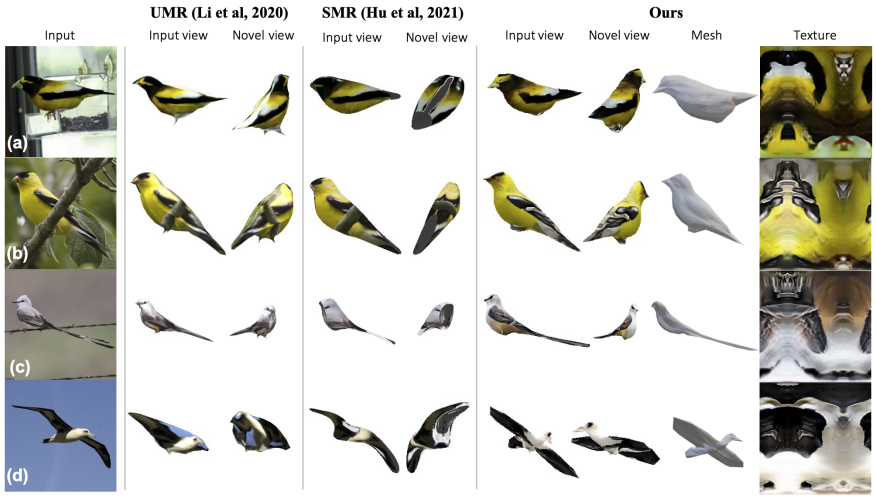

Fig. 1. We propose an alternative approach to monocular 3D reconstruction by exploiting generative prior encapsulated in a pre-trained GAN. Our method has three major advantages: **1)** It reconstructs highly faithful and realistic 3D objects, even when observed from novel views; **2)** The reconstruction is robust against occlusion in (b); **3)** The method generalizes reasonably well to less common shapes, such as birds with (c) extended tails or (d) open wings.

single-view images and their corresponding masks for training [24,10,4,15]. Minimizing the reconstruction error in the 2D domain tends to ignore the overall 3D geometry and back-side appearance, leading to a shortcut solution that may look plausible only from the input viewpoint, *e.g.*, SMR [15] in Fig. 1 (a)(c). While these methods compensate the relaxed supervision by exploiting various forms of prior information, *e.g.*, categorical semantic invariance [24] and interpolated consistency of the predicted 3D attributes [15], this task remains challenging.

In this work, we propose a new approach, **MeshInversion**, that is built upon generative prior possessed by Generative Adversarial Networks (GANs) [11]. GANs are typically known for their exceptional ability to capture comprehensive knowledge [5,21,41], empowering the success of GAN inversion in image restoration [12,37] and point cloud completion [52]. We believe that by training a GAN to synthesize 3D shapes in the form of a topology-aligned texture and deformation map, one could enable the generator to capture rich prior knowledge of a certain object category, including high-level semantics, object geometries, and texture details.

We propose to exploit the appealing generative prior through GAN inversion. Specifically, our framework finds the latent code of the pre-trained 3D GAN that best recovers the 3D object in accordance with the single-view observation. Given the RGB image and its associated silhouette mask estimated by an off-the-shelf segmentation model, the latent code is optimized towards minimizing

2D reconstruction losses by rendering the 3D object onto the 2D image plane. Hence, the latent manifold of the 3D GAN *implicitly* constrains the reconstructed 3D shape within the realistic boundaries, whereas minimization of 2D losses *explicitly* drives the 3D shape towards a faithful reflection of the input image.

Searching for the optimal latent code in the GAN manifold for single-view 3D object reconstruction is non-trivial due to following challenges: 1) Accurate camera poses are not always available for real-world applications. Inaccurate camera poses easily lead to reprojection misalignment and thus erroneous reconstruction. 2) Existing geometric losses that are computed between 2D masks inevitably discretize mesh vertices into a grid of pixels during rasterization. Such discretization typically makes the losses less sensitive in reflecting the subtle geometric variations in the 3D space. To address the misalignment issue, we propose a **Chamfer Texture Loss**, which relaxes the one-to-one pixel correspondences in existing losses and allows the match to be found within a local region. By jointly considering the appearance and positions of image pixels or feature vectors, it provides a robust texture distance despite inaccurate camera poses and in the presence of high-frequency textures. To improve the geometric sensitivity, we propose a **Chamfer Mask Loss**, which intercepts the rasterization process and computes the Chamfer distance between the projected vertices before discretization to retain information, with the foreground pixels of the input image projected to the same continuous space. Hence, it is more sensitive to small variations in shape and offers a more accurate gradient for geometric learning.

MeshInversion demonstrates compelling performance for 3D reconstruction from real-world monocular images. Even with the assumption of inaccurate masks and camera poses, our method still gives highly plausible and faithful 3D reconstruction in terms of both appearance and 3D shape, as depicted in Fig. 1. It achieves state-of-the-art results on the perceptual metric, *i.e.*, FID, when evaluating the textured mesh from various viewpoints, and is on-par with the existing CMR-based frameworks in terms of geometric accuracy. In addition, while its holistic understanding of the objects benefits from the generative prior, it not only gives a realistic recovery of the back-side texture but also generalizes well in the presence of occlusion, *e.g.*, Fig. 1 (b). Furthermore, MeshInversion also demonstrates significantly better generalization for 3D shapes that are less commonly seen, such as birds with open wings and long tails, as shown in Fig. 1 (d) and (c) respectively.

2 Related Work

Single-view 3D Reconstruction. Many methods have been proposed to recover the 3D information of an object, such as its shape and texture, from a single-view observation. Some methods use image-3D object pairs [46,35,32,39] or multi-view images [33,28,51,47,34] for training, which limit the scenarios to synthetic data. Another line of work fits the parameters of a 3D prior morphable model, *e.g.*, SMPL for humans and 3DMM for faces [8,40,18], which are typically expensive to build and difficult to extend to various natural object categories.

To relax the constraints on supervision, CMR [19] reconstructs category-specific textured mesh by training with a collection of monocular images and associated 2D supervisions, *i.e.*, 2D key-points, camera poses, and silhouette masks. Thereafter, several follow-up studies further relax the supervision, *e.g.*, masks only, and improves the reconstruction results by exploiting different forms of prior. Specifically, they incorporate the prior by enforcing various types of cycle consistencies, such as texture cycle consistency [24,4], rotation adversarial cycle consistency [4], and interpolated consistency [15]. Some of these methods also leverage external information, *e.g.*, category-level mesh templates [10,4], and semantic parts provided by an external SCOPS model [24]. In parallel, ShelfSup [7] first gives a coarse volumetric prediction, and then converts the coarse volume into a mesh followed by test-time optimization. Without categorical mesh templates in existing approaches, this design demonstrates its scalability to categories with high-genus meshes, *e.g.*, chairs and backpacks.

For texture modeling, direct regression of pixel values in the UV texture map often leads to blurry images [10]. Therefore, the mainstream approach is to regress pixel coordinates, *i.e.*, learning *texture flow* from the input image to the texture map. Although texture flow is easier to regress and usually provides a vivid front view result, it often fails to generalize well to novel views or occluded regions. Our approach directly predicts the texture pixel values by incorporating a pre-trained GAN. In contrast to the texture flow approach, it benefits from a holistic understanding of the objects given the generative prior and offers high plausibility and fidelity at the same time.

GAN Inversion. A well-trained GAN usually captures useful statistics and semantics underlying the training data. In the 2D domain, GAN prior has been explored extensively in various image restoration and editing tasks [3,12,37]. GAN inversion, the common method in this line of work, finds a latent code that best reconstructs the given image using the pre-trained generator. Typically, the target latent code can be obtained via gradient descent [29,27], projected by an additive encoder that learns the inverse mapping of a GAN [2], or a combination of them [53]. There are recent attempts to apply GAN inversion in the 3D domain. Zhang *et al*. [52] use a pre-trained point cloud GAN to address shape completion in the canonical pose, giving remarkable generalization for out-of-domain data such as real-world partial scans. Pan *et al*. [36] recover the geometric cues from pre-trained 2D GANs and achieve exceptional reconstruction results, but the reconstructed shapes are limited to 2.5D due to limited poses that 2D GANs can synthesize. In this work, we directly exploit the prior from a 3D GAN to reconstruct the shape and texture of complete 3D objects.

Textured Mesh Generation. 3D object generation approaches that use voxels [48,9,43,54,50] or point clouds [1,41] typically require some form of 3D supervision and are unfriendly for modeling texture. Chen *et al*. [6] propose DIB-R, a GAN framework for textured mesh generation, where 3D meshes are differentiably rendered into 2D images and discriminated with multi-view images of synthetic objects. Later on, Henderson *et al*. [13] relax the multi-view restriction and propose a VAE framework [22] that leverages a collection of single-view nat-

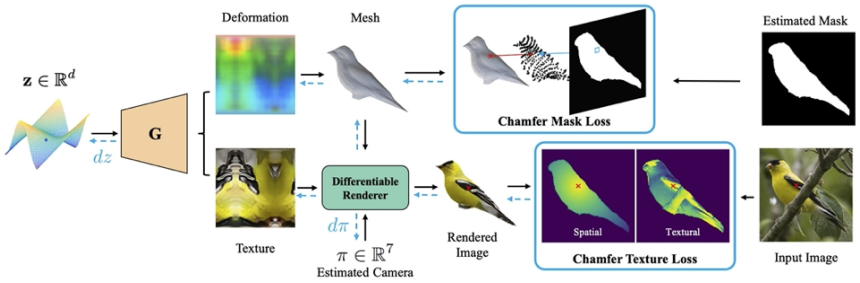

Fig. 2. Overview of the MeshInversion framework. We reconstruct a 3D object from its monocular observation by incorporating a pre-trained textured 3D GAN **G**. We search for the latent code $\mathbf{z}$ and fine-tune the imperfect camera π that minimizes 2D reconstruction losses via gradient descent. To address the intrinsic challenges associated with 3D-to-2D degradation, we propose two Chamfer-based losses: 1) **Chamfer Texture Loss** (Sec 3.2) relaxes the pixel-wise correspondences between two RGB images or feature maps, and factorizes the pairwise distance into spatial and textural distance terms. We illustrate the distance maps between one anchor point from the rendered image to the input image, where brighter regions correspond to smaller distances. 2) **Chamfer Mask Loss** (Sec 3.3) intercepts the discretization process and computes the Chamfer distance between the projected vertices and the foreground pixels projected to the same continuous space. No labeled data is assumed during inference: the mask and camera are estimated by off-the-shelf pre-trained models.

ural images. The appearance is parameterized by face colors instead of texture maps, limiting the visual detail of generated objects. Under the same setting, ConvMesh [38] achieves more realistic 3D generations by generating 3D objects in the form of topology-aligned texture maps and deformation maps in the UV space, where discrimination directly takes place in the UV space against pseudo ground truths. Pseudo deformation maps are obtained by overfitting a mesh reconstruction baseline on the training set. Subsequently, the associated pseudo texture maps can then be obtained by projecting natural images on the UV space. Our proposed method is built upon a pre-trained ConvMesh model to incorporate its generative prior in 3D reconstruction.

3 Approach

Preliminaries. We represent a 3D object as a textured triangle mesh $\mathbf{O} \equiv (\mathbf{V}, \mathbf{F}, \mathbf{T})$, where $\mathbf{V} \in \mathbb{R}^{|\mathbf{v}| \times 3}$ represents the location of the vertices, $\mathbf{F}$ represents the faces that define the fixed connectivity of vertices in the mesh, and $\mathbf{T}$ represents the texture map. An individual mesh is iso-morphic to a 2-pole sphere, and thus we model the deformation $\Delta\mathbf{V}$ from the initial sphere template, and then obtain the final vertex positions by $\mathbf{V} = \mathbf{V}_{sphere} + \Delta\mathbf{V}$. Previous methods [19,24,10] mostly regress deformation of individual vertices via a fully connected network (MLP). In contrast, recent studies have found that using a 2D convolutional neural network (CNN) to learn a deformation map in the UV

space would benefit from consistent semantics across the entire category [38,4]. In addition, the deformation map $\mathbf{S}$ and the texture map $\mathbf{T}$ are topologically aligned, so both the values can be mapped to the mesh via the same predefined mapping function.

We assume a weak-perspective camera projection, where the camera pose π is parameterized by scale $\mathbf{s} \in \mathbb{R}$, translation $\mathbf{t} \in \mathbb{R}^2$, and rotation in the form of quaternion $\mathbf{r} \in \mathbb{R}^4$. We use DIB-R [6] as our differentiable renderer. We denote $\mathbf{I} = R(\mathbf{S}, \mathbf{T}, \pi)$ as the image rendering process. Similar to previous baselines [19,24,15,10], we enforce reflectional symmetry along the x axis, which both benefits geometric performance and reduces computation cost.

3.1 Reconstruction with Generative Prior

Our study presents the first attempt to explore the effectiveness of generative prior in monocular 3D construction. Our framework assumes a pre-trained textured 3D GAN. In this study, we adopt ConvMesh [38], which is purely trained with 2D supervisions from single-view natural images. With the help of GAN prior, our goal is to recover the geometry and appearance of a 3D object from a monocular image and its associated mask. Unlike SMR [15] that uses ground truth masks, our method takes silhouettes estimated by an off-the-shelf segmentation model [23].

Next, we will detail the proposed approach to harness the meaningful prior, such as high-level semantics, object geometries, and texture details, from this pre-trained GAN to achieve plausible and faithful recovery of 3D shape and appearance. Note that our method is not limited to ConvMesh, and other pre-trained GANs that generate textured meshes are also applicable. More details of ConvMesh can be found in the supplementary materials.

Pre-training Stage. Prior to GAN inversion, we first pre-train the textured GAN on the training split to capture desirable prior knowledge for 3D reconstruction. As discussed in Sec. 2, the adversarial training of ConvMesh takes place in the UV space, where generated deformation maps and texture maps are discriminated against their corresponding pseudo ground truth. In addition to the UV space discrimination, we further enhance the photorealism of the generated 3D objects by introducing a discriminator in the image space, following the architecture of PatchGAN as in [16]. The loss functions for the pre-training stage are shown as follows, where D_{uv} and D_I refer to the discriminators in the UV space and image space respectively, and λ_{uv} and λ_I are the corresponding weights. We use least-squares losses following [30]. An ablation study on image space discrimination can be found in the supplementary materials.

$$\mathcal{L}_G = \lambda_{uv}\mathbb{E}_{\mathbf{z}\sim P_{\mathbf{z}}}[(D_{uv}(G(\mathbf{z})) - 1)^2] + \lambda_I \mathbb{E}_{\mathbf{z}\sim P_{\mathbf{z}}(\mathbf{z})}[(D_I(R(G(\mathbf{z})), \pi) - 1)^2]. \quad (1)$$

$$\mathcal{L}_{D_{uv}} = \mathbb{E}_{\mathbf{S},\mathbf{T}\sim P_{pseudo}}[(D_{uv}(\mathbf{S}, \mathbf{T}) - 1)^2] + \mathbb{E}_{\mathbf{z}\sim P_{\mathbf{z}}(\mathbf{z})}[(D_{uv}(G(\mathbf{z})))^2]. \quad (2)$$

$$\mathcal{L}_{D_I} = \mathbb{E}_{\mathbf{I}\sim P_{data}}[(D_I(\mathbf{I}) - 1)^2] + \mathbb{E}_{\mathbf{z}\sim P_{\mathbf{z}}(\mathbf{z})}[(D_I(R(G(\mathbf{z})), \pi)^2]. \quad (3)$$

Inversion Stage. We now formally introduce GAN inversion for single-view 3D reconstruction. Given a pre-trained ConvMesh that generates a textured mesh from a latent code, $\mathbf{S}, \mathbf{T} = G(\mathbf{z})$, we aim to find the $\mathbf{z}$ that best recovers the 3D object from the input image $\mathbf{I}_{in}$ and its silhouette mask $\mathbf{M}_{in}$. Specifically, we search for such $\mathbf{z}$ via gradient descent towards minimizing the overall reconstruction loss $\mathcal{L}_{inv}$, which can be denoted by

$$\mathbf{z}^* = \arg\min_{\mathbf{z}} \mathcal{L}_{inv}(R(G(\mathbf{z}), \pi), \mathbf{I}_{in}). \tag{4}$$

Given the single-view image and the associated mask, we would need to project the reconstructed 3D object to the observation space for computing $\mathcal{L}_{inv}$. However, such 3D-to-2D degradation is non-trivial. Unlike existing image-based GAN inversion tasks where we can always assume pixel-wise image correspondence in the observation space, rendering 3D objects in the canonical frame onto the image space is explicitly controlled via camera poses. For real-world applications, unfortunately, perfect camera poses are not always available to guarantee such pixel-wise image correspondence. While concurrently optimizing the latent code and the camera pose from scratch seems a plausible approach, this often suffers from camera-shape ambiguity [24] and leads to erroneous reconstruction. To this end, we initialize the camera with a camera pose estimator (CMR [19]), which can be potentially inaccurate, and jointly optimize it in the course of GAN inversion, for which we have:

$$\mathbf{z}^*, \pi^* = \arg\min_{\mathbf{z}, \pi} \mathcal{L}_{inv}(R(G(\mathbf{z}), \pi), \mathbf{I}_{in}). \tag{5}$$

As the 3D object and cameras are constantly optimized throughout the inversion stage, it is infeasible to assume perfect image alignment. In addition, the presence of high-frequency textures, *e.g.*, complex bird feathers, often leads to blurry appearance even with slight discrepancies in pose. Consequently, it calls for a robust form of texture loss in Sec 3.2.

3.2 Chamfer Texture Loss

To facilitate searching in the GAN manifold without worrying about blurry reconstructions, we reconsider the appearance loss by relaxing the pixel-aligned assumption in existing low-level losses. Taking inspiration from the point cloud data structure, we treat a 2D image as a set of 2D colored points, which have both appearance attributes, *i.e.*, RGB values, and spatial attributes, the values of which relate to their coordinates in the image grid. Thereafter, we aim to measure the dissimilarity between the two colored point sets via Chamfer distance,

$$\mathcal{L}_{CD}(\mathbb{S}_1, \mathbb{S}_2) = \frac{1}{|\mathbb{S}_1|} \sum_{x \in \mathbb{S}_1} \min_{y \in \mathbb{S}_2} \mathbf{D}_{xy} + \frac{1}{|\mathbb{S}_2|} \sum_{y \in \mathbb{S}_2} \min_{x \in \mathbb{S}_1} \mathbf{D}_{yx}. \tag{6}$$

Intuitively, defining the pairwise distance between pixel x and pixel y in the two respective images should jointly consider their appearance and location. In this

regard, we factorize the overall pairwise distance $\mathbf{D}_{xy}$ into an appearance term $\mathbf{D}^a_{xy}$ and a spatial term $\mathbf{D}^s_{xy}$, both of which are L2 distance. Like conventional Chamfer distance, single-sided pixel correspondences are determined by column-wise or row-wise minimum in the distance matrix $\mathbf{D}$.

Importantly, we desire the loss to be tolerant and only tolerant of local misalignment, as large misalignment will potentially introduce noisy pixel correspondences that may jeopardize appearance learning. Inspired by the focal loss for detection [25], we introduce an exponential operation in the spatial term to penalize those spatially distant pixel pairs. Therefore, we define the overall distance matrix $\mathbf{D} \in \mathbb{R}^{|\mathbb{S}_1| \times |\mathbb{S}_2|}$ as follows

$$\mathbf{D} = \max((\mathbf{D}^s + \epsilon_s)^\alpha, 1) \otimes (\mathbf{D}^a + \epsilon_a), \tag{7}$$

where $\mathbf{D}^a$ and $\mathbf{D}^s$ are the appearance distance matrix and spatial distance matrix respectively; $\otimes$ denotes element-wise product; ϵ_s and ϵ_a are residual terms to avoid incorrect matches with identical location or identical pixel value respectively; α is the scaling factor for flexibility. Specifically, we let $\epsilon_s < 1$ so that the spatial term remains one when two pixels are slightly misaligned. Note that the spatial term is not differentiable and it only serves as a weight matrix for appearance learning. By substituting the resulting $\mathbf{D}$ into Eq. 6, we thus have the final formulation of our proposed **Chamfer Texture Loss**, denoted as $\mathcal{L}_{CT}$.

The proposed relaxed formulation provides a robust measure of texture distance, which effectively eases searching of the target latent code while preventing blurry reconstructions; in return, although $\mathcal{L}_{CT}$ only concerns about local patch statistics but not photorealism, the use of GAN prior is sufficient to give realistic predictions. Besides, the GAN prior also allows computing $\mathcal{L}_{CT}$ with a down-sampled size of colored points. In practice, we randomly select 8096 pixels from each image as a point set.

The proposed formulation additionally gives flexible control between appearance and spatial attributes. The appearance term is readily extendable to accept misaligned feature maps to achieve more semantically faithful 3D reconstruction. Specifically, we apply the Chamfer texture loss between the (foreground) feature maps extracted with a pre-trained VGG-19 network [42] from the rendered image and the input image. It is worth noting that the feature-level Chamfer texture loss is somewhat related to the contextual loss [31], which addresses the misalignment issue for image transfer. The key difference is that the contextual loss only considers the feature distances but ignores their locations. We compare against the contextual loss in the experiment.

3.3 Chamfer Mask Loss

Conventionally, the geometric distance is usually computed between two binary masks in terms of L1 or IoU loss [19,10,24,15]. However, obtaining the mask of the reconstructed mesh usually involves rasterization that discretizes the mesh into a grid of pixels. This operation inevitably introduces information loss and thus inaccurate supervision signals. This is particularly harmful to a well-trained

ConvMesh, the shape manifold of which is typically smooth. Specifically, a small perturbation in $\mathbf{z}$ usually corresponds to a slight variation in the 3D shape, which may translate to an unchanged binary mask. This usually leads to an insensitive gradient for back-propagation, which undermines geometric learning. We analyze the sensitivity of existing losses in the experiment.

To this end, we propose a **Chamfer Mask Loss**, or $\mathcal{L}_{CM}$, to compute the geometric distance in an unquantized 2D space. Instead of rendering the mesh into a binary mask, we directly project the 3D vertices of the mesh onto the image plane, $\mathbb{S}_v = P(\mathbf{S}, \mathbf{T}, \pi)$. For the foreground mask, we obtain the positions of the foreground pixels by normalizing their pixel coordinates in the range of $[-1, 1]$, denoted as $\mathbb{S}_f$. Thereafter, we compute the Chamfer distance between $\mathbb{S}_v$ and $\mathbb{S}_f$ as the Chamfer mask loss. Note that one does not need to distinguish visible and occluded vertices, as eventually they all fall within the rendered silhouette. The bidirectional Chamfer distance between the sparse set $\mathbb{S}_v$ and dense set $\mathbb{S}_f$ would regularize the vertices from highly uneven deformation.

3.4 Overall Objective Function

We apply the pixel-level Chamfer texture loss $\mathcal{L}_{pCT}$ and the feature-level one $\mathcal{L}_{fCT}$ as our appearance losses, and the Chamfer mask loss $\mathcal{L}_{CM}$ as our geometric loss. Besides, we also introduce two regularizers: the smooth loss $\mathcal{L}_{smooth}$ that encourages neighboring faces to have similar normals, *i.e.*, low cosine; the latent space loss $\mathcal{L}_z$ that regularizes the L2 norm of $\mathbf{z}$ to ensure Gaussian distribution. In summary, the overall objective function is shown in Eq. 8.

$$\mathcal{L}_{inv} = \mathcal{L}_{pCT} + \mathcal{L}_{fCT} + \mathcal{L}_{CM} + \mathcal{L}_{smooth} + \mathcal{L}_z. \tag{8}$$

4 Experiments

Datasets and Experimental Setting. We primarily evaluate MeshInversion on CUB-200-2011 dataset [45]. It consists of 200 species of birds with a wide range of shapes and feathers, making it an ideal benchmark to evaluate 3D reconstruction in terms of both geometric and texture fidelity. Apart from the organic shapes like birds, we also validate our method on 11 man-made rigid car categories from PASCAL3D+ [49].

We use the same train-validation-test split as provided by CMR [19]. The images in both datasets are annotated with foreground masks and camera poses. Specifically, we pre-train ConvMesh on the pseudo ground truths derived from the training split following a class conditional setting [38]. During inference, we conduct GAN inversion on the test split without assuming additional labeled data compared to existing methods. We use the silhouette masks predicted by an off-the-shelf instance segmentation method PointRend [23] pre-trained on COCO [26], which gives an IoU of 0.886 against ground truth masks. We use camera poses predicted by CMR [19], which can be inaccurate. In particular, the poses estimated yields 6.03 degree of azimuth error and 4.33 degree of elevation

Table 1. Quantitative results on CUB show the effectiveness of applying generative prior in 3D reconstruction. As all the baseline methods are regression-based whereas our method involves optimization during inference, we report both baselines and test-time optimization (TTO) results for existing methods, if applicable, with access to masks estimated by PointRend [23]. SMR baseline uses ground truth mask, and it shows noticeable IoU drop with estimated mask. †: We report results from [4] since no implementation released; the results are based on ground truth cameras, whereas our method optimizes from imperfect cameras.

Methods	TTO	input mask	IoU ↑	FID_1 ↓	FID_{10} ↓	FID_{12} ↓
CMR [19]		-	0.703	140.9	176.2	180.1
UMR [24]		-	0.734	40.0	72.8	86.9
U-CMR [10]		-	0.701	65.0	314.9	315.2
View-gen [4] †		-	0.629	-	-	70.3
SMR [15]		estimated	0.751	55.9	65.7	85.6
SMR		ground truth	**0.800**	52.9	63.2	79.3
CMR	✓	estimated	0.717	121.6	150.5	158.4
UMR	✓	estimated	0.739	38.8	78.2	91.3
Ours	✓	estimated	0.752	**37.3**	**38.7**	**56.8**

error compared to the ground truth cameras via structure-from-motion (SfM). During evaluation, we report quantitative results based on ground truth masks.

Evaluation Strategy. Since there are no 3D ground truths available for CUB, we evaluate MeshInversion against various baselines from three aspects: 1) We evaluate the geometry accuracy in the 2D domain by IoU between the rendered masks and the ground truths. 2) We evaluate the appearance quality by the image synthesis metric FID [14], which compares the distribution of test set images and the render of reconstruction. Since a plausible 3D shape should look photo-realistic observed from multiple viewpoints, we report both single-view FID (**FID_1**) and multi-view FIDs. Following SMR [15] and View-gen [4], we render our reconstructed 3D shape from 12 different views (**FID_{12}**), which covers azimuth from 0° to 360° at an interval of 30°. We additionally report **FID_{10}** since the exact front view (90°) and the exact back view (270°) are rarely seen in CUB. Note that this is in favour of existing methods that do not use any GAN prior as ours. 3) Apart from extensive qualitative results, we conduct a user study to evaluate human preferences in terms of both shape and appearance. For PASCAL3D+, it provides approximated 3D shapes using a set of 10 CAD models, which allows us to evaluate geometric performance in terms of 3D IoU.

4.1 Comparison with Baselines

We compare MeshInversion with various existing methods on the CUB dataset, and report quantitative results in Tab. 1. Overall, MeshInversion achieves state-of-the-art results on perceptual metrics, particularly multi-view FIDs, and is on par with existing methods in terms of IoU. The qualitative results in Fig. 1, Fig. 3 and Fig. 4 show that MeshInversion achieves highly faithful and realistic

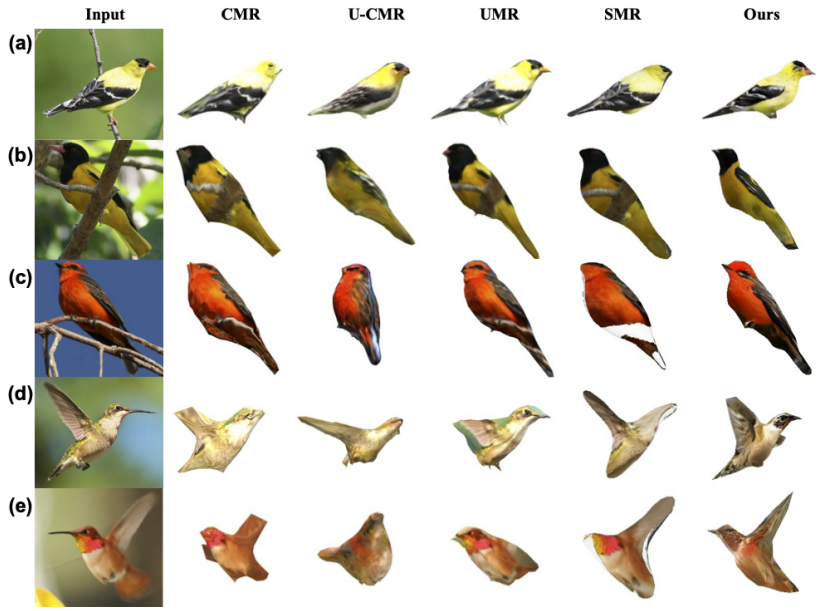

Fig. 3. Qualitative results on CUB. Our method achieves highly faithful and realistic 3D reconstruction. In particular, it exhibits superior generalization under various challenging scenarios, including with occlusion (b, c) and extended articulation (d, e).

3D reconstruction, particularly when observed from novel views. Moreover, our method generalizes reasonably well to highly articulated shapes, such as birds with long tails and open wings, where many of the existing methods fail to give satisfactory reconstructions, as Fig. 1 (c)(d) and Fig. 3 (d)(e) show. We note that although SMR gives the competitive IoU with estimated mask, it lacks fine geometric details and looks less realistic, *e.g.*, the beak in Fig. 1 (c) and Fig. 3.

Texture Flow vs. Texture Regression. Texture flow is extensively adopted in existing methods, except for U-CMR. Although texture flow-based methods are typically easier to learn and give superior texture reconstruction for visible regions, they tend to give incorrect predictions for invisible regions, *e.g.*, abdomen or back as shown in Fig. 1. In contrast, MeshInversion, which performs direct regression of textures, benefits from a holistic understanding of the objects and gives remarkable performance in the presence of occlusion, while texture flow-based methods only learn to copy from the foreground pixels including the obstacles, *e.g.*, twig, from the bird, as shown in Fig. 1 (a) and Fig. 3 (b)(c). Due to the same reason, these methods also tend to copy background pixels onto the reconstructed object when the shape prediction is inaccurate, as shown in Fig. 1 (d) and Fig. 3 (d). More qualitative results and multi-view comparisons can be found in the supplementary materials.

Test-time Optimization. While existing methods mostly adopt an auto-encoder framework and perform inference with a single forward pass, MeshInversion is

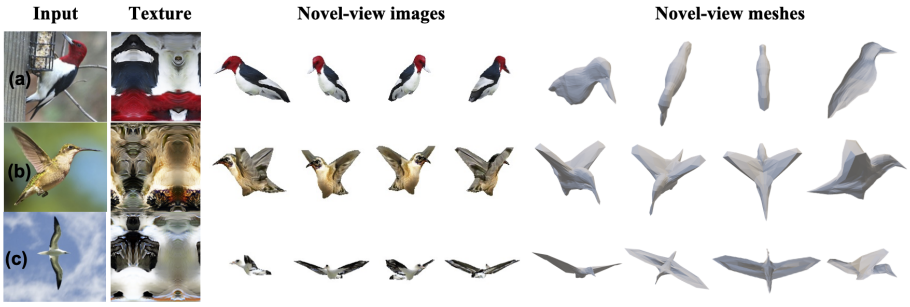

Fig. 4. Novel-view rendering results on CUB. Our method gives realistic and faithful 3D reconstruction in terms of both 3D shape and appearance. It generalizes fairly well to invisible regions and challenging articulations.

optimization-based. For a fair comparison, we also introduce test-time optimization (TTO) for baseline methods, if applicable, with access to predicted masks as well. Specifically, CMR and UMR have a compact latent code with a dimension of 200, which is desirable for efficient fine-tuning. As shown in Tab. 1, TTO of existing methods overall yields higher fidelity, but our proposed method remains highly competitive in terms of perceptual and geometric performance. Interestingly, UMR with TTO achieves marginal improvement in terms of IoU and single-view FID at the cost of worsening novel-view FID. This further shows the superiority of generative prior captured through adversarial training over that captured in an auto-encoder, including UMR that is coupled with adversarial training, and its effectiveness of such appealing prior in 3D reconstruction.

User Study. We further conduct a user preference study on multi-view renderings of 30 randomly selected birds, and ask 40 users to choose the most realistic and faithful reconstruction in terms of texture, shape, and overall 3D reconstruction. Tab. 2 shows that MeshInversion gives the the most preferred results, whereas all texture flow-based methods give poor results mainly due to their incorrect prediction for unseen regions.

Table 2. User preference study on CUB in terms of the quality and faithfulness of texture, shape, and overall 3D reconstruction.

Criterion	CMR	U-CMR	UMR	SMR	Ours
Texture	2.7%	15.7%	13.2%	6.6%	**61.8%**
Shape	2.7%	19.6%	14.6%	4.2%	**58.9%**
Overall	2.5%	19.8%	12.8%	3.3%	**61.5%**

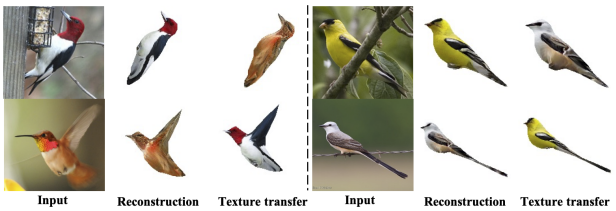

Fig. 5. Our method enables faithful and realistic texture transfer between bird instances even with highly articulated shapes.

4.2 Texture Transfer

As the shape and texture are topologically and semantically aligned in the UV space, it allows us to modify the surface appearance across bird instances. In Fig. 5, we sample pairs of instances and swap their texture maps. Thanks to the categorical semantic consistency, the resulting new 3D objects remain highly realistic even for extended articulations like open wings and long tails.

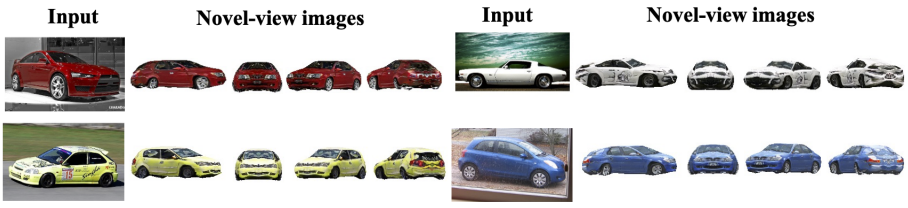

Fig. 6. Qualitative results on PASCAL3D+ Car. Our method gives reasonably good performance across different car models and appearances.

4.3 Evaluation on PASCAL3D+ Car

We also evaluate MeshInversion on the man-made rigid car category. As demonstrated in Fig. 6, it performs reasonably well across different car models and appearances. Unlike [4] and [10] that explicitly use one or multiple mesh templates provided by PASCAL3D+, the appealing GAN prior implicitly provides a rich number of templates that makes it possible to reconstruct cars of various models. Given the approximated 3D ground truths in PASCAL3D+, we show in Tab 3 that MeshInversion performs comparably to baselines in terms of 3D IoU.

Table 3. 3D IoU on PASCAL3D+ Car. Both the deformable model fitting-based method CSDM [20] and volume-based method DRC [44] do not predict object texture.

	CSDM	DRC	CMR	Ours
3D IoU	0.60	**0.67**	0.64	0.66

4.4 Ablation Study

Effectiveness of Chamfer Mask Loss. As compared to Chamfer mask loss in Tab. 4, both conventional mask losses give significantly worse reconstruction results. As existing mask losses are obtained through rasterization, the induced information loss often makes them less accurate in capturing subtle shape variations, which undermines geometry recovery. A detailed sensitivity study of various mask losses can be found in the supplementary materials.

Effectiveness of Chamfer Texture Loss. In the presence of imperfect poses and high-frequency details in textures, we show in Tab. 4 that our proposed pixel- and feature-level Chamfer texture losses are highly effective compared to existing losses. In particular, pixel-to-pixel L1 loss tends to give blurry reconstructions. Feature-based losses, perceptual loss [17] and contextual loss [31], are generally

Table 4. Ablation study. Compared to conventional texture and mask losses, our proposed pixel-level Chamfer texture losses $\mathcal{L}_{pCT}$ and feature-level one $\mathcal{L}_{fCT}$, and Chamfer mask loss $\mathcal{L}_{CM}$ are effective to address the challenges due to misalignment and quantization during rendering. Despite using not-so-accurate camera poses, our method gives compelling geometric and perceptual performance by jointly optimizing 3D shape and camera during GAN inversion.

Mask loss	Texture loss	Camera	IoU ↑	FID_1 ↓	FID_{10} ↓	FID_{12} ↓
IoU loss	L1 + perceptual loss	fine-tuned	0.580	82.8	78.3	92.01
L_{CM}	L1 loss	fine-tuned	0.732	58.9	78.3	74.7
L_{CM}	L1 + perceptual loss	fine-tuned	0.741	56.3	44.0	64.9
L_{CM}	contextual loss	fine-tuned	0.718	69.0	57.7	75.5
L1 loss	L_{pCT} + L_{fCT}	fine-tuned	0.589	71.8	73.2	74.1
IoU loss	L_{pCT} + L_{fCT}	fine-tuned	0.604	72.1	71.8	70.5
L_{CM}	L_{pCT} + L_{fCT}	fixed	0.695	40.9	41.38	60.0
L_{CM}	L_{pCT} + L_{fCT}	fine-tuned	0.752	**37.3**	**38.7**	**56.8**

more robust to misalignment, but they are usually not discriminative enough to reflect the appearance details between two images. Although the contextual loss is designed to address the misalignment issue for image-to-image translation, it only considers feature distances while ignoring their positions in the image.

Effectiveness of Optimizing Camera Poses. In Tab. 4, we show that compared to only optimizing latent code, jointly fine-tuning imperfect camera poses effectively improves geometric performance.

5 Discussion

We have presented a new approach for monocular 3D object reconstruction. It exploits generative prior encapsulated in a pre-trained GAN and reconstructs textured shapes through GAN inversion. To address reprojection misalignment and discretization-induced information loss due to 3D-to-2D degradation, we propose two Chamfer-based losses in the 2D space, *i.e.*, Chamfer texture loss and Chamfer mask loss. By efficiently incorporating the GAN prior, MeshInversion achieves highly realistic and faithful 3D reconstruction, and exhibits superior generalization power for challenging cases, such as in the presence of occlusion or extended articulations. However, this challenging problem is far from being solved. In particular, although we can faithfully reconstruct flying birds with open wings, the wings are only represented by a few vertices due to semantic consistency across the entire category, which strictly limits the representation power in terms of geometry and texture details. Therefore, future work may explore more flexible solutions, for instance, an adaptive number of vertices can be assigned to articulated regions to accommodate richer details.

Acknoledgement. This study is supported under the RIE2020 Industry Alignment Fund – Industry Collaboration Projects (IAF-ICP) Funding Initiative, Singapore MOE AcRF Tier 2 (MOE-T2EP20221-0011), Shanghai AI Laboratory, as well as cash and in-kind contribution from the industry partners.

References

1. Achlioptas, P., Diamanti, O., Mitliagkas, I., Guibas, L.: Learning representations and generative models for 3D point clouds. In: ICML (2018) 4
2. Bau, D., Strobelt, H., Peebles, W., Zhou, B., Zhu, J.Y., Torralba, A., et al.: Semantic photo manipulation with a generative image prior. In: SIGGRAPH (2019) 4
3. Bau, D., Zhu, J.Y., Wulff, J., Peebles, W., Strobelt, H., Zhou, B., Torralba, A.: Seeing what a GAN cannot generate. In: ICCV (2019) 4
4. Bhattad, A., Dundar, A., Liu, G., Tao, A., Catanzaro, B.: View generalization for single image textured 3D models. In: CVPR (2021) 2, 4, 6, 10, 13
5. Brock, A., Donahue, J., Simonyan, K.: Large scale GAN training for high fidelity natural image synthesis. In: ICLR (2019) 2
6. Chen, W., Ling, H., Gao, J., Smith, E., Lehtinen, J., Jacobson, A., Fidler, S.: Learning to predict 3D objects with an interpolation-based differentiable renderer. In: NeurIPS (2019) 4, 6
7. *et al*, Y.: Shelf-supervised mesh prediction in the wild. In: CVPR (2021) 4
8. Gecer, B., Ploumpis, S., Kotsia, I., Zafeiriou, S.: GANFIT: Generative adversarial network fitting for high fidelity 3D face reconstruction. In: CVPR (2019) 3
9. Girdhar, R., Fouhey, D.F., Rodriguez, M., Gupta, A.: Learning a predictable and generative vector representation for objects. In: ECCV (2016) 4
10. Goel, S., Kanazawa, A., Malik, J.: Shape and viewpoint without keypoints. In: ECCV (2020) 2, 4, 5, 6, 8, 10, 13
11. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: NeurIPS (2014) 2
12. Gu, J., Shen, Y., Zhou, B.: Image processing using multi-code GAN prior. In: CVPR (2020) 2, 4
13. Henderson, P., Tsiminaki, V., Lampert, C.H.: Leveraging 2D data to learn textured 3D mesh generation. In: CVPR (2020) 4
14. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: Gans trained by a two time-scale update rule converge to a local nash equilibrium. In: NeurIPS (2017) 10
15. Hu, T., Wang, L., Xu, X., Liu, S., Jia, J.: Self-supervised 3D mesh reconstruction from single images. In: CVPR (2021) 2, 4, 6, 8, 10
16. Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. In: CVPR (2017) 6
17. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: ECCV (2016) 13
18. Kanazawa, A., Black, M.J., Jacobs, D.W., Malik, J.: End-to-end recovery of human shape and pose. In: CVPR (2018) 3
19. Kanazawa, A., Tulsiani, S., Efros, A.A., Malik, J.: Learning category-specific mesh reconstruction from image collections. In: ECCV (2018) 1, 4, 5, 6, 7, 8, 9, 10
20. Kar, A., Tulsiani, S., Carreira, J., Malik, J.: Category-specific object reconstruction from a single image. In: CVPR (2015) 13
21. Karras, T., Laine, S., Aila, T.: A style-based generator architecture for generative adversarial networks. In: CVPR (2019) 2
22. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. In: ICLR (2014) 4
23. Kirillov, A., Wu, Y., He, K., Girshick, R.: PointRend: Image segmentation as rendering. In: CVPR (2020) 6, 9, 10

24. Li, X., Liu, S., Kim, K., De Mello, S., Jampani, V., Yang, M.H., Kautz, J.: Self-supervised single-view 3D reconstruction via semantic consistency. In: ECCV (2020) 2, 4, 5, 6, 7, 8, 10
25. Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal loss for dense object detection. In: ICCV (2017) 8
26. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft COCO: Common objects in context. In: ECCV (2014) 9
27. Lipton, Z.C., Tripathi, S.: Precise recovery of latent vectors from generative adversarial networks. CoRR **arXiv:1702.04782** (2017) 4
28. Liu, S., Chen, W., Li, T., Li, H.: Soft rasterizer: Differentiable rendering for unsupervised single-view mesh reconstruction. In: ICCV (2019) 3
29. Ma, F., Ayaz, U., Karaman, S.: Invertibility of convolutional generative networks from partial measurements. In: NeurIPS (2018) 4
30. Mao, X., Li, Q., Xie, H., Lau, R.Y., Wang, Z., Paul Smolley, S.: Least squares generative adversarial networks. In: ICCV (2017) 6
31. Mechrez, R., Talmi, I., Zelnik-Manor, L.: The contextual loss for image transformation with non-aligned data. In: ECCV (2018) 8, 13
32. Mescheder, L., Oechsle, M., Niemeyer, M., Nowozin, S., Geiger, A.: Occupancy networks: Learning 3D reconstruction in function space. In: CVPR (2019) 3
33. Niemeyer, M., Mescheder, L., Oechsle, M., Geiger, A.: Differentiable volumetric rendering: Learning implicit 3D representations without 3D supervision. In: CVPR (2020) 3
34. Oechsle, M., Peng, S., Geiger, A.: UNISURF: Unifying neural implicit surfaces and radiance fields for multi-view reconstruction. In: ICCV (2021) 3
35. Pan, J., Han, X., Chen, W., Tang, J., Jia, K.: Deep mesh reconstruction from single rgb images via topology modification networks. In: ICCV (2019) 3
36. Pan, X., Dai, B., Liu, Z., Loy, C.C., Luo, P.: Do 2D gans know 3D shape? unsupervised 3D shape reconstruction from 2D image GANs. In: ICLR (2021) 4
37. Pan, X., Zhan, X., Dai, B., Lin, D., Loy, C.C., Luo, P.: Exploiting deep generative prior for versatile image restoration and manipulation. PAMI (2021) 2, 4
38. Pavllo, D., Spinks, G., Hofmann, T., Moens, M.F., Lucchi, A.: Convolutional generation of textured 3D meshes. In: NeurIPS (2020) 5, 6, 9
39. Rematas, K., Martin-Brualla, R., Ferrari, V.: ShaRF: Shape-conditioned radiance fields from a single view. In: ICML (2021) 3
40. Sanyal, S., Bolkart, T., Feng, H., Black, M.J.: Learning to regress 3D face shape and expression from an image without 3D supervision. In: CVPR (2019) 3
41. Shu, D.W., Park, S.W., Kwon, J.: 3D point cloud generative adversarial network based on tree structured graph convolutions. In: ICCV (2019) 2, 4
42. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: ICLR (2015) 8
43. Smith, E.J., Meger, D.: Improved adversarial systems for 3D object generation and reconstruction. In: CoRL (2017) 4
44. Tulsiani, S., Zhou, T., Efros, A.A., Malik, J.: Multi-view supervision for single-view reconstruction via differentiable ray consistency. In: CVPR (2017) 13
45. Wah, C., Branson, S., Welinder, P., Perona, P., Belongie, S.: The caltech-ucsd birds-200-2011 dataset (2011) 9
46. Wang, N., Zhang, Y., Li, Z., Fu, Y., Liu, W., Jiang, Y.G.: Pixel2Mesh: Generating 3D mesh models from single rgb images. In: ECCV (2018) 3
47. Wang, P., Liu, L., Liu, Y., Theobalt, C., Komura, T., Wang, W.: NeuS: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. In: NeurIPS (2021) 3

48. Wu, J., Zhang, C., Xue, T., Freeman, B., Tenenbaum, J.: Learning a probabilistic latent space of object shapes via 3D generative-adversarial modeling. In: NeurIPS (2016) 4
49. Xiang, Y., Mottaghi, R., Savarese, S.: Beyond PASCAL: A benchmark for 3D object detection in the wild. In: WACV (2014) 9
50. Xie, J., Zheng, Z., Gao, R., Wang, W., Zhu, S.C., Wu, Y.N.: Learning descriptor networks for 3D shape synthesis and analysis. In: CVPR (2018) 4
51. Yariv, L., Kasten, Y., Moran, D., Galun, M., Atzmon, M., Basri, R., Lipman, Y.: Multiview neural surface reconstruction by disentangling geometry and appearance. In: NeurIPS (2020) 3
52. Zhang, J., Chen, X., Cai, Z., Pan, L., Zhao, H., Yi, S., Yeo, C.K., Dai, B., Loy, C.C.: Unsupervised 3D shape completion through GAN inversion. In: CVPR (2021) 2, 4
53. Zhu, J., Shen, Y., Zhao, D., Zhou, B.: In-domain GAN inversion for real image editing. In: ECCV (2020) 4
54. Zhu, J.Y., Zhang, Z., Zhang, C., Wu, J., Torralba, A., Tenenbaum, J., Freeman, B.: Visual object networks: Image generation with disentangled 3D representations. In: NeurIPS (2018) 4