GSIR: Generalizable 3D Shape Interpretation and Reconstruction

Jianren Wang[(✉)] and Zhaoyuan Fang

Carnegie Mellon University, Pittsburgh, USA
{jianrenw,zhaoyuaf}@andrew.cmu.edu

Abstract. 3D shape interpretation and reconstruction are closely related to each other but have long been studied separately and often end up with priors that are highly biased towards the training classes. In this paper, we present an algorithm, *Generalizable 3D Shape Interpretation and Reconstruction (GSIR)*, designed to jointly learn these two tasks to capture generic, class-agnostic shape priors for a better understanding of 3D geometry. We propose to recover 3D shape structures as cuboids from partial reconstruction and use the predicted structures to further guide full 3D reconstruction. The unified framework is trained simultaneously offline to learn a generic notion and can be fine-tuned online for specific objects without any annotations. Extensive experiments on both synthetic and real data demonstrate that introducing 3D shape interpretation improves the performance of single image 3D reconstruction and vice versa, achieving the state-of-the-art performance on both tasks for objects in both seen and unseen categories.

Keywords: Shape interpretation · 3D reconstruction

1 Introduction

Single image 3D geometry has attracted much attention in recent years due to its numerous applications, such as robotics, medicine and film industry. To fully understand 3D geometry, it is essential to know structure properties (*e.g.*, symmetry, compactness, planarity, and part to part relations) [8,30,43] and surface properties (*e.g.*, texture and curvature). In this paper, we address these problems simultaneously, i.e., 3D shape interpretation and reconstruction, in which these two tasks have been known to be closely related to each other [28,55].

For single image 3D reconstruction, the difficulty is mainly reflected in two aspects: how to extract geometric information from high dimensional images and how to utilize prior shape knowledge to pick the most reasonable prediction from many 3D explanations. Recent research tackles these problems through deep learning [13,18,56], since it has shown great success in image information distillation tasks like classification [27], detection [24] and segmentation [22]. Many algorithms have explored ways to utilize shape prior knowledge. For example, *ShapeHD* [61] integrated deep generative models with adversarially learned shape priors and penalized the model only if its outputs were unrealistic.
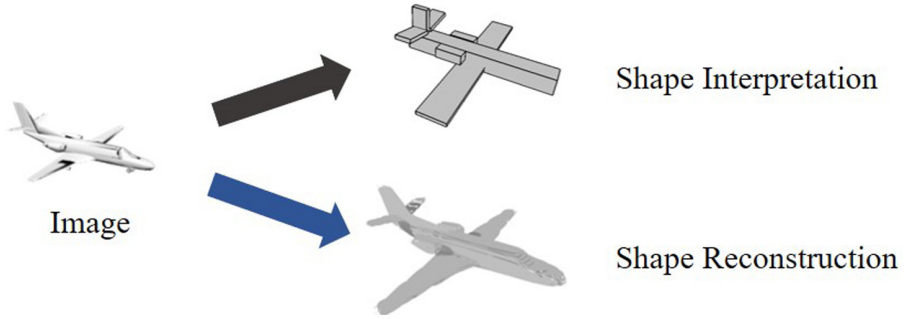
Fig. 1. We present Generalizable 3D Shape Interpretation and Reconstruction (*GSIR*) to learn 3D shape interpretation and reconstruction jointly

Many existing methods do not enforce explicit 3D representation in the model, which leads to overfitting. As a result, they suffer when reconstructing the unobservable parts of objects, especially under self-occlusions. Recently, methods that encode shapes in a function [38,40] take a step toward better generalization. In this paper, we approach the problem by enforcing explicit 3D representation in the model. Inspired by pose-guided person generation [7,34], we propose a structure-guided shape generation that explicitly uses the structure to guide shape completion and reconstruction. The key idea of our approach is to guide the reconstruction process explicitly by an appropriate representation of the object structure to enable direct control over the generation process. More specifically, we propose to condition the reconstruction network on both the observable parts of the object and a predicted structure. From the observable parts, the model obtains sufficient information about the visible surface of the object. The guidance given by the predicted structure is both explicit and flexible. There are many other interesting downstream applications. For example, we later show that we can design new objects by keeping the original surface details and manipulate the size and orientation of each part of the object by changing the guidance.

On the other hand, single image 3D structure interpretation itself is challenging and often inaccurate. Therefore, the derived structure information does not always help reconstruction. More specifically, when an image is captured from accidental views, the structure interpretation methods are not effective to predict landmarks positions [3] or primitive orientations [39]. To overcome this problem, we bring reconstructed 3D information to help the algorithm predict more accurate interpretations (cuboid position, orientation, and size in our case).

Based on the above observations, we propose to jointly reason about single image *generalizable 3D shape interpretation and reconstruction (GSIR)*. Building upon GenRe [64], we first project a predicted 2.5D sketch into a partial 3D model. We then generate geometrically interpretable representations of the partial 3D model through oriented cuboids, where symmetry, compactness, planarity, and part-to-part relations are taken into consideration. Instead of

performing shape completion in the 3D voxel grid, our method completes the shape based on spherical maps since mapping a 2D image/2.5D sketch to a 3D shape involves complex but deterministic geometric projections. Using spherical map, our neural modules only need to model object geometry, without having to learn projections, which enhances generalizability. Unlike *GenRe*, we perform the completion in a structure-guided manner. Fusing information from both the visible object surfaces and the projected spherical maps of oriented cuboids and edges, we can further complete non-visible parts of the object.

Our model consists of four learnable modules: single-view depth estimation module, structure interpretation module, structure-guided spherical map inpainting module, and voxel refinement module. In addition, geometric projections form the links between those modules. Furthermore, we propose an interpretation consistency between the predicted structure and the partial 3D reconstruction.

Our approach offers three unique advantages. First, our estimated 3D structure encodes symmetry, compactness, planarity, and part-to-part relations of the given objects explicitly, which help us understand the reconstruction in a more transparent way. Second, we reason about 3D structure from partial observable voxel grid to alleviate the burden on domain transfer in previous single image 3D structure interpretation algorithms [39,59], which enhances generalizability. Third, our interpretation consistency can be used to fine-tune the system for specific objects without any annotations, which further enables the communication between two branches (the consistency can be jointly optimized with the model).

We evaluate our method on both synthetic images of objects from the ShapeNet dataset, and real images from the PASCAL 3D+ dataset. We show that our method performs well on 3D shape reconstruction, both qualitatively and quantitatively on novel objects from unseen categories. We also show the method's capacity to generate new objects given modified shape guidance.

To summarize, this paper makes four contributions: we propose an end-to-end trainable model *(GSIR)* to jointly reason 3D shape interpretation and reconstruction; we develop a structure-guided 3D reconstruction algorithm; we develop a novel end-to-end trainable loss that ensures consistency between estimated structure and partially reconstructed model; we demonstrate that exploiting symmetry, compactness, planarity, and part-to-part relations inside object can significantly improve both shape interpretation and reconstruction accuracy and help with generalization.

2   Related Work

Single Image 3D Reconstruction. Lots of work have been done on 3D reconstruction from single images. Early works can be traced back to Hoiem *et al.* [26] and Saxena *et al.* [49]. Theoretically, recovering 3D shapes from single-view images is an ill-posed problem. To alleviate the ill-posedness, these methods rely heavily on the knowledge of shape priors, which require large amount of data.

With the releasing of IKEA [32] and ShapeNet [9], many learning-based methods begin to dominate the trend. Choy *et al.* [13] apply a CNN to the input image, then pass the resulting features to a 3D deconvolutional network, that maps them to an occupancy grid of 32^3 voxels. Girdhar *et al.* [18] and Wu *et al.* [60] proceed similarly, but pre-train a model to encode or generate 3D shapes respectively, and regress images to the latent features of the model. Instead of directly producing voxels, Arsalan Soltani *et al.* [2], Shin *et al.* [50], Wu *et al.* [58] and Richter *et al.* [44] output multiple depth-maps and/or silhouettes, which are subsequently fused for voxel reconstruction. Although we focus on reconstructing 3D voxels, there are many other works that reconstruct 3d objects using pointcloud [16,29,35], meshes [21,25,33,55,57], octrees [45,46,54], and functions [14,38,51,63]. [25] presents a general framework to learn reconstruction and generation of 3D shapes with 2D supervision, using an assembly from cuboidal primitives as a compact representation. To encode both geometry and appearance, [51] encodes a feature and RGB representation for each point and predicts the surface location with a ray marching LSTM network. [63] combines 3D point features with image features from the projected query patch and significantly improves on 3D reconstruction. [38] represents the 3D surface as continuous decision boundaries and shows robust results.

3D Structure Interpretation. Different from 3D reconstruction, 3D structure interpretation focuses on understanding structure properties instead of dense representations, which is broadly defined based on positions and relationships among semantic (the vertical part), functional (support and stability), economic (repeatable and easy to fabricate) parts. Among all ways to abstract object structures, a 3d skeleton is most common in use because of its simplicity, especially in human pose estimation [1,6,42,65]. 3D-INN [59] estimate 3D object skeletons through 2D keypoints and achieve a promising result on chairs and cars. Another way is to represent the method using volumetric primitives, which can date to the very beginnings of the computer vision. There are many attempts to represent shapes as a collection of components or primitives, such as geons [5], block world [47] and cylinders [36]. Recently, more compact and parametric representations are introduced using LSTM [66] or set of primitives [55].

Structure-Aware Shape Processing Previous studies have recognized the value of structure-guided shape processing, editing, and synthesis, mainly in computer graphics [17] and geometric modeling [19]. For shape synthesis, many approaches have been proposed based on fixed relationships such as regular structures [41], symmetries [52], probabilistic assembly-based modeling [10]. Wu *et al.* [62] encode the structure into an embedding vector. The work that is most similar to ours is probably SASS proposed by Balashova *et al.* [3]. SASS extracts landmarks from a 3D shape and adds a shape-structure consistency loss to better align shape with predicted landmarks. Our model has two advantages over SASS. First, instead of using a fixed number of landmarks, we abstract primitives of any given object. This gives more freedom to the objects that can be constructed. Second, our proposed method deeply integrates shape interpretation and reconstruction through structure-guided inpainting and the interpretation consistency other than just force the alignment.

Depth Prediction. The ability to learn depth using a deep learning framework was introduced by [15], who uses a dataset of ground truth depth and RGB image pairs to train a network to predict depth. This has been further improved through better architecture [11,31] and larger datasets [37].
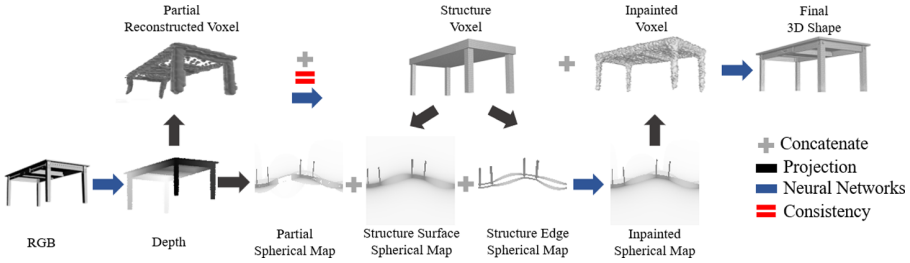
3   Approach

Fig. 2. Our model contains four learnable functions and five deterministic projection functions.

Our whole model (Fig. 2) consists of four learnable functions (f) connected by five deterministic projection functions (p). The model is summarized below and each module is discussed in details in the subsections:

1. The model begins with a **single-view depth estimation module**: with a color image (RGB) as input, the module estimates its depth map $D = f(RGB)$. We then convert the depth estimation D into partial reconstructed voxel grid $V_p = p(D)$, which reflects only visible surfaces.
2. Our second learnable function is the **structure interpretation module**: the partial voxel grid (V_p) is taken as input and parsed by the module into compact cuboid-based representations $S = f(V_p)$. We then project the resulting structure surfaces and edges into spherical maps: $M_{ss} = p(surface(S)), M_{se} = p(egde(S))$.
3. Along with projected spherical maps from depth estimation $M_p = p(D)$, the **structure-guided shape completion module** can predict the inpainted spherical map $M_i = f(M_p, M_{ss}, M_{se})$, which is then projected back into voxel space $V_i = p(M_i)$.
4. Since spherical maps only capture the outermost surface towards the sphere, they cannot handle self-occlusion along the sphere's radius. To mitigate this problem, we adopt the **voxel refinement module** that takes all predicted voxels as input and outputs the final reconstruction $V = f(V_p, V_i, S)$.

3.1 Single-View Depth Estimation Module

Since depth estimation is a class-agnostic task, we use depth as an intermediate representation like many other methods [44,58]. Previous research shows that depth estimation can be generalized well into different classes despite their distinct visual appearances and can even be applied in the wild [11]. Our module takes a color image (RGB) as input and estimates its depth map (D) through an encoder-decoder network. More details can be viewed in Sect. 3.6.

3.2 Structure Interpretation Module

To better represent the symmetry, compactness, planarity, and part-to-part relations, we adopt a recursive neural network as the 3D structure interpreter like in [28]. However, unlike [39], we encode the structure embedding from V_p to alleviate the domain adaptation. The encoder is achieved by a 3D convolutional network that encodes V_p into a bottleneck feature, then the decoder recursively decodes it into a hierarchy of part boxes.

Starting from the root feature code, the RNN recursively decodes it into a hierarchy of features until reaching the leaf nodes which each can be further decoded into a vector of box parameters. There are three types of nodes in our hierarchy: leaf node, adjacency node, and symmetry node. During the decoding, two types of part relations are recovered as the class of internal nodes: adjacency and symmetry. Thus, each node can be decoded by one of the three decoders below, based on its type (adjacency node, symmetry node or box node):

Adjacency Decoder. The adjacency decoder split a single part into two adjacent parts. Formally, it splits a parent n-D code p into two child n-D codes c_1 and c_2, using the mapping function with a weight matrix $W_{ad} \in \mathbb{R}^{2n \times n}$ and a bias vector $b_{ad} \in \mathbb{R}^{2n}$:

$$[c_1, c_2] = tanh(W_{ad} \cdot p + b_{ad}) \qquad (1)$$

Symmetry Decoder. The symmetry decoder recovers a n-D code for a symmetry group g in the form of a n-D code for the symmetry generator s and a m-D code for the symmetry parameters z. The transformation has a weight matrix $W_{sd} \in \mathbb{R}^{n \times (n+m)}$ and a bias vector $b_{sd} \in \mathbb{R}^{n+m}$:

$$[s, z] = tanh(W_{sd} \cdot g + b_{sd}) \qquad (2)$$

The symmetry parameters are represented as a 8-dim vector ($m = 8$) containing: symmetry type (1D); number of repetitions for rotation and translation symmetries (1D); and the reflection plane for reflection symmetry, rotation axis for rotation symmetry, or position and displacement for translation symmetry (6D).

Box Decoder. The box decoder converts the n-D code of a leaf node l to a 12-D box parameters defining the center, axes, and sizes of a 3D oriented box. It has a weight matrix $W_{ld} \in \mathbb{R}^{12 \times n}$ and a bias vector $b_{ld} \in \mathbb{R}^{12}$:

$$[x] = tanh(W_{ld} \cdot l + b_{ld}) \qquad (3)$$

These decoders are recursively applied during decoding. We also need to distinguish p, g and l since they require different decoders. This is achieved by learning a node classifier where the ground-truth box structure is known. The node classifier is jointly trained with the three decoders. We refer the readers to [28] for a better understanding.

3.3   Structure-Guided Shape Completion Module

The problem of 3D surface completion was first cast into 2D spherical map inpainting by *GenRe* [64], showing better performance than surface completion in the voxel space. However, the original spherical inpainting network takes only the partially observable depth map M_p as input and encode the shape prior implicitly in their neural network. We use an encoder-decoder network and concatenate M_p, M_{ss}, M_{se} channel-wise as input: structure surface map M_{ss} provides the reference depth as it shows the planar tilt; structure edges M_{se} handles self-occlusion as edges do not have volume. Thus, structure information is explicitly embedded into the network. Note both structure and depth map are viewer-centered and are automatically aligned.

3.4   Voxel Refinement Module

We adopt a voxel refinement module to recover the lost information caused by spherical projection, similar to *GenRe*. This module takes all voxels (one projected from the estimated depth map V_p and the other from the inpainted spherical map V_i) as well as the voxelized structure S as input, and predict the final reconstruction.

3.5   Interpretation Consistency

There have been works attempting to enforce the consistency between estimated 3D shape and 2D representations or 2.5D sketches [58] in a neural network. Here, we propose a consistency loss between structure interpretation S and partial reconstruction V_p.

Similar to [55], our consistency loss contains both sub loss and super loss. The former evaluates if the interpretation cuboids are completely inside the target object, the latter evaluates if the target object is completely covered by the interpretation cuboids.

Formally, sub loss L_{sub} and super loss L_{sup} are defined as

$$L_{sub} = E_{p \sim V_p} \|C(p; S)\|^2 \tag{4}$$

$$L_{sup} = E_{p \sim S} \|C(p; V_p)\|^2 \tag{5}$$

$$L = L_{sub} + L_{sup} \tag{6}$$

where the points p are sampled from either the structure interpretation or the partial reconstruction, and $C(\cdot; O)$ computes the distance to the closest point on the object and equals to zero in the object interior.

$$C(p; O) = \min_{p' \in O} \|p - p'\|^2 \tag{7}$$

Note that the reconstruction V_p only contains observable parts, so it is not reasonable to force consistency in the occluded region. Therefore, we only calculate the consistency loss of structure primitive where the volume occupied by V_p is larger than a threshold α. We fix the three decoders mentioned in Sect. 3.2 during testing and only fine-tune the node codes and parameters. During inference, our method can be self-supervised.

3.6   Technical Details

Network Parameters. Following *GenRe* [64], we use a U-Net structure [48] for both single-view depth estimation module and structure-guided shape completion module. The encoder is a ResNet-18 [23], encoding a 256×256 image into 512 feature maps of size 1×1. The decoder is a mirrored version of the encoder, replacing all convolution layers with transposed convolution layers. The decoder outputs the depth map/inpainted map in the original view at the resolution of 256×256. We use a $L2$ loss between predicted and target images. Our structure interpretation module takes the $128 \times 128 \times 128$ dimensional V_p as input and output a 128D latent vector, which is then fed into the RNN decoder. The node classifier and the decoders for both adjacency and symmetry are two-layer networks, with the hidden layer and output layer having 256 and 128 units, respectively. Our voxel refinement module is also a U-Net, which takes a three-channel $128 \times 128 \times 128$ voxel grid (V_p, V_i, S) as input, encode it into a 320D latent vector and then decode the latent vector into the $128 \times 128 \times 128$ dimensional final reconstruction.

Geometric Projections. We use five deterministic projection functions: a depth to voxel projection, a depth to spherical map projection, a structure surfaces to spherical map projection, a structure edges to spherical map projection, and a spherical map to voxel projection. We use the same method as described in *GenRe.* All projections are differentiable, thus the pipeline is end-to-end trainable.

Training. We first train each module separately with fully labelled ground truth for 250 epochs, all rendered with synthetic ShapeNet objects [9]. We then jointly fine-tune our whole model together with both 3D shape and 3D structure supervision for another 250 epochs. In practice, we fine-tuned our model using consistency loss on each image for 30 iterations. We used adam optimizer with a learning rate of 1×10^{-4}.

4   Experiments

4.1   3D Shape Interpretation

Table 1. Comparison of performance on the structure recovery task.

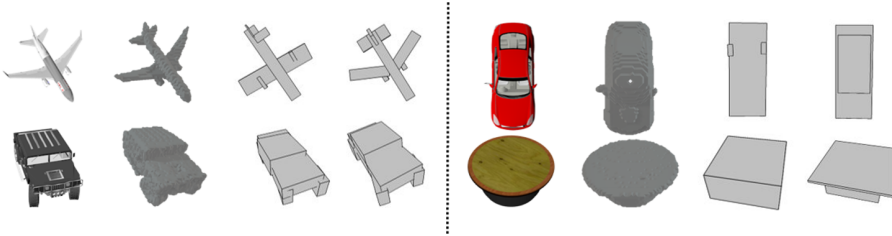| Method	Hausdorff Error	Thresholded Acc.	
		$\delta < 0.2$	$\delta < 0.1$
im2struct (Mask + VGG-19) [39]	0.1096	91.2%	66.7%
GSIR (without consistency)	0.0798	93.3%	79.6%
GSIR (With consistency)	0.0731	97.4%	84.8%

Fig. 3. Example results of 3D shape interpretation. From left to right: RGB input image, partial voxel grid, *im2struct*, Ours*(GSIR)*.

We present results on 3D shape interpretation for generalizing to novel objects unseen in training. All models are trained on cars, chairs, airplanes, tables, and motorcycles and tested on unseen objects from the same categories. Same as in *im2struct* [39], we use two measures to evaluate the performance of our 3D Shape Interpretation: *Hausdorff Error* and *Thresholded Accuracy*. The results are presented in Table 1. We compare our method with the current best method (*im2struct*). In "*GSIR* without consistency", the structure is estimated using only the structure interpretation module. In "*GSIR* with consistency", the structure is esimated using the structure interpretation module followed by a refinement using the proposed interpretation consistency. The result demonstrates that recovering structure significantly benefits from infusing information of partially reconstructed voxel grid. Figure 3 gives a visual comparison of our method and *im2struct*, which directly recover 3D shape from single-view RGB image. As can be seen, our method produces part structures that are more faithful to the input. This is because 1) we reason about 3D structure from predicted 3D voxels, which alleviates the domain adaptation, and 2) our model is end-to-end trainable, the performance of structure recovery gets better as richer information gets distilled for 3D reconstruction.
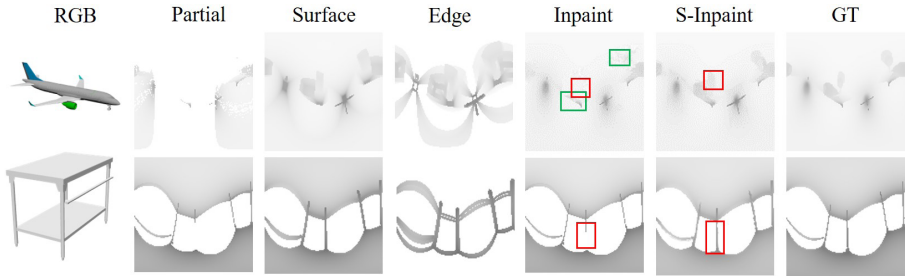
4.2   Structure-Guided Shape Completion

Fig. 4. Visualization of example spherical maps at each stage of our method, with a comparison of structure guided inpainting and normal inpainting. From left to right: RGB (original), partial map from depth estimation, surface map from structure prediction, edge map from structure prediction, inpainted map without structure guidance, inpainted map with structure guidance, ground truth. (Color figure online)

We present qualitative results on structure-guided shape completion in Fig. 4. The contribution of each element in our method is visualized in the figure. We show that with structure guidance, the missing or unobservable parts can be well completed, hence leading a more faithful reconstruction. However, without structure information, the inpainting network can only recover incomplete unobservable parts (*e.g.*, the wing of the airplane bounded by the green boxes) or even ignore the unobservable parts directly (*e.g.*, the engine of the airplane and the leg of the table bounded by the red boxes). In contrast, structure guidance enables the model to fully reconstruct unobservable parts. More quantitative results are shown in Sect. 4.3.

4.3   3D Shape Reconstruction

In Table 2, we present results on generalizing to novel objects from both training and testing classes. All models are trained on ShapeNet cars, chairs, airplanes, tables, and motorcycles while tested on novel objects from the same categories (denoted as Seen) and unseen categories (denoted as Unseen) including benches, sofa, beds and vessels. Since our model only focuses on surface voxel reconstruction, we evaluate reconstruction quality using Chamfer distance (CD) [4]. We sweep voxel thresholds from 0.3 to 0.7 with a step size of 0.05 for isosurfaces, compute CD with 1,024 points sampled from all isosurfaces, and report the best average CD for each object class. For seen categories, our method beats all other viewer-centered methods, performing on par with most object-centered methods. For unseen objects, our model outperforms all objected-centered and viewer-centered methods by a large margin, demonstrating its capacity to generalize to objects with new shapes from completely unseen classes.
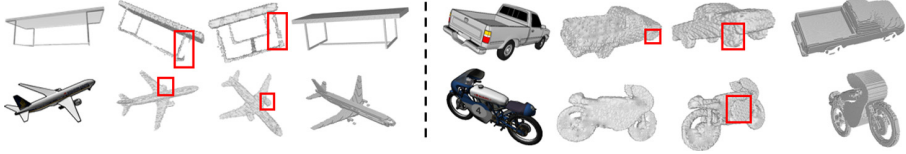
Fig. 5. Example results of 3D shape reconstruction for novel objects from training categories. From left to right: RGB image, *GenRe*, Ours*(GSIR)*, Ground Truth. The red bounding boxes surround key areas that suffer from self-occlusion/symmetry in *GenRe* but are successfully reconstructed by the proposed method. (Color figure online)
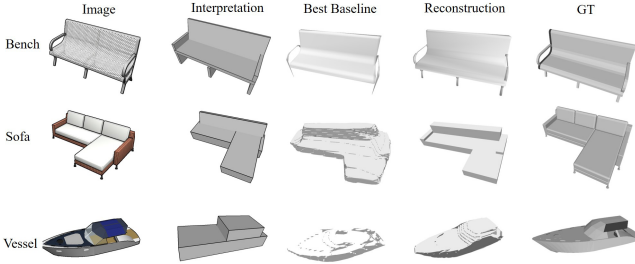

Fig. 6. Example results of 3D shape reconstruction for novel objects from testing categories. From left to right: RGB image, structural interpretation, *GenRe* (Best Baseline), Ours*(GSIR)*, Ground Truth.

We give a visual comparison of our method and the state-of-the-art method on novel objects from seen categories in Fig. 5. The red bounding boxes surround key areas that suffer from self-occlusion/symmetry in *GenRe* but are successfully reconstructed by the proposed method. These results show that our method significantly improves the reconstruction performance under self-

Table 2. Comparison of performance on the shape reconstruction task.

Method		CD	
		Seen	Unseen
Object-Centered	*IM-NET* [12]	0.055	0.119
	ONet [38]	0.060	0.128
	DeepSDF [40]	**0.053**	0.115
	AtlasNet [20]	0.063	0.126
Viewer-Centered	*DRC* [56]	0.097	0.127
	MarrNet [58]	0.081	0.116
	GenRe [64]	0.068	0.108
	Ours	0.057	**0.099**

occlusion/symmetry. We also present some visualizations on novel objects from unseen categories in Fig. 6. It can be observed that compared to the best previous method, our method better preserves the structural properties of the objects in the input images and closely reconstructs various details of the objects (*e.g.*, the middle leg of the bench, the armrest of the sofa, and the ceiling of the vessel, *etc*).

4.4   Shape Interpretation with Consistency

By reasoning the consistency between the partial voxel grid and object structure, we can obtain better structure interpretation by fine-tuning on one object while preserving good shape prior knowledge. As shown in Fig. 7, the tilt and size of each cuboid can be rectified even if the initial structure interpretation is coarse and distorted (as shown in the red boxes). Furthermore, since our structure model utilizes symmetry explicitly, the unobservable parts can also be better reconstructed through forcing consistency with observable parts.
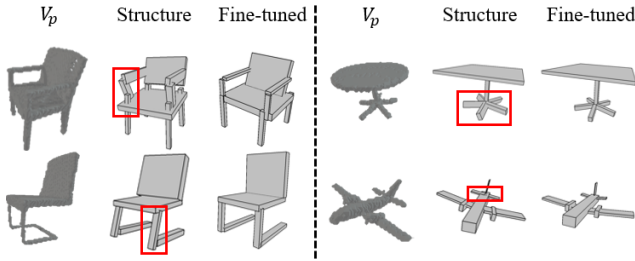

Fig. 7. Example results that demonstrates the efficacy of the proposed interpretation consistency. From left to right: partial voxel grid (V_p), coarsely reconstructed structure (Structure), fine-tuned structure with consistency (Fine-tuned).

4.5   Generalization to Real Images

In this subsection, we extend our experiments from rendered images to real images. Our experiments show that the proposed network's capability to robustly reconstruct objects of unseen classes from real images, both qualitatively and quantitatively. For example, all models are trained on rendered images of chairs, airplanes, and cars from ShapeNet, while tested on real images of beds, bookcases, desks, sofas, tables, and wardrobes from another dataset, Pix3D [53]. Quantitative results evaluated by Chamfer Distance are presented in Table 3. While *AtlasNet* achieves a smaller error on seen objects (chairs & tables), our model outperforms both other methods across all novel classes, which demonstrate its generalization abilities on cross-domain shape interpretation and reconstruction. We also present qualitative visualizations in Fig. 8. Both our interpretation network and reconstruction network produce high-fidelity results, preserving both the overall structure and fine-grained details.
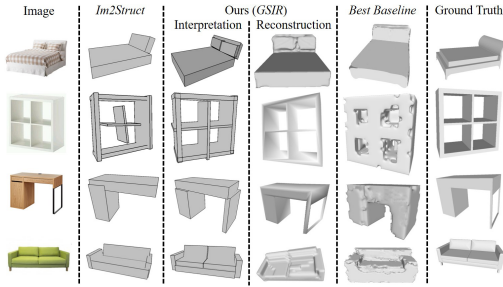
| Image | Im2Struct | Ours (GSIR) Interpretation Reconstruction | Best Baseline | Ground Truth |

Fig. 8. Example results of 3D shape interpretation and reconstruction on real images of objects from unseen classes in Pix3D (the model is trained on ShapeNet).

Table 3. Reconstruction errors (in CD) for seen (chairs, tables) and unseen classes (beds, bookcases, sofas, wardrobes) on real images from Pix3D.

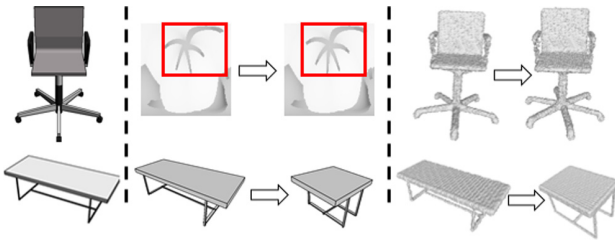|	*AtlasNet*	*GenRe*	Ours
Chair	**0.083**	0.095	0.091
Table	**0.092**	0.099	0.094
Bed	0.115	0.111	**0.107**
Bookcase	0.137	0.101	**0.095**
Desk	0.124	0.107	**0.100**
Sofa	0.096	0.085	**0.083**
Wardrobe	0.119	0.111	**0.103**

Fig. 9. Examples of structure-guided shape manipulation. We change the leg number of a swivel chair from five to six and shorten the length of a table.

4.6   Ablation Study

To investigate the effectiveness of each module in our model design, we perform an ablation study to quantify the performance of different module design configurations.

Table 4. Ablation Study. All annotations are consistent with Sect. 3.

Method	Seen	Unseen
Encoder Decoder	0.127	0.196
Depth + Decoder	0.088	0.131
Depth + Multi-view + V_i	0.075	0.123
Depth + Multi-view + Guided + V_i	0.072	0.121
Depth + Spherical Map + V_i	0.073	0.119
Depth + Spherical Map + Guided + V_i	0.069	0.113
Depth + Spherical Map + Guided + V_i + V_p	0.064	0.106
Ours (w.o. consistency loss)	0.060	0.103
Ours (w. consistency loss)	**0.057**	**0.099**

For each projection representation in our model, there could be alternative choices: instead of using spherical map, we can instead use a multi-view representation: *e.g.*, six views depth projection as proposed by MatryoshkaNet [44]. Then we can apply structure-guided depth map inpainting on all six views (denoted as Multi-view in Table 4).

In the ablation study, we gradually add more representations and more projective losses. The baseline method is a single vanilla 3D autoencoder (denoted as Encoder Decoder). Then, each module added sequentially, bearing the same name as mentioned in Sect. 3. We adopt the same experimental settings as in Sect. 4.3 and the results are shown in Table 4. Results suggest that spherical maps lead to better performance than multi-view ensemble, which justify our choice of design. This ablation study also suggests that each module in our model contributes to the improved performance. Our full model design benefits from joint learning of interpretation and reconstruction, significantly improving the baseline network performance.

4.7   Shape Manipulation

Another unique advantage of our method is that it provides explicit and flexible ways to manipulate the underlying objects while maintaining good surface details. We can modify the symmetry groups (*e.g.*, changing the number of legs of a chair from five to six) in structure-guided shape completion step (as shown in the first row of Fig. 9), and/or apply rotation, translation or scaling to the primitives (as shown in the second row of Fig. 9). As shown in Fig. 9, our model smoothly modifies the output of reconstruction according to the structure guidance.

5   Conclusion

We jointly learned single image 3D shape interpretation and reconstruction. We propose *GSIR*, an novel end-to-end trainable viewer-centered model that integrates both shape structure and surface details, for a better understanding of 3D geometry. Extensive experiments on both synthetic and real data demonstrate that with this joint structure, both interpretation and reconstruction results can be improved. We hope our work will inspire future research in this direction.

References

1. Akhter, I., Black, M.J.: Pose-conditioned joint angle limits for 3D human pose reconstruction. In: CVPR (2015)
2. Arsalan Soltani, A., Huang, H., Wu, J., Kulkarni, T.D., Tenenbaum, J.B.: Synthesizing 3D shapes via modeling multi-view depth maps and silhouettes with deep generative networks. In: CVPR (2017)
3. Balashova, E., Singh, V., Wang, J., Teixeira, B., Chen, T., Funkhouser, T.: Structure-aware shape synthesis. In: 3DV (2018)

4. Barrow, H.G., Tenenbaum, J.M., Bolles, R.C., Wolf, H.C.: Parametric correspondence and chamfer matching: two new techniques for image matching. In: IJCAI (1977)

5. Biederman, I.: Recognition-by-components: a theory of human image understanding. Psychol. Rev. **94**, 115 (1987)

6. Bogo, F., Kanazawa, A., Lassner, C., Gehler, P., Romero, J., Black, M.J.: Keep it SMPL: automatic estimation of 3D human pose and shape from a single image. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9909, pp. 561–578. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46454-1_34

7. Chan, C., Ginosar, S., Zhou, T., Efros, A.A.: Everybody dance now. arXiv preprint arXiv:1808.07371 (2018)

8. Chan, M.W., Stevenson, A.K., Li, Y., Pizlo, Z.: Binocular shape constancy from novel views: the role of a priori constraints. Percept. Psychophysics **68**(7), 1124–1139 (2006)

9. Chang, A.X., et al.: Shapenet: an information-rich 3D model repository. arXiv preprint arXiv:1512.03012 (2015)

10. Chaudhuri, S., Kalogerakis, E., Guibas, L., Koltun, V.: Probabilistic reasoning for assembly-based 3D modeling. In: ACM TOG, vol. 30, p. 35 (2011)

11. Chen, W., Fu, Z., Yang, D., Deng, J.: Single-image depth perception in the wild. In: NeurIPS (2016)

12. Chen, Z., Zhang, H.: Learning implicit fields for generative shape modeling. In: CVPR (2019)

13. Choy, C.B., Xu, D., Gwak, J.Y., Chen, K., Savarese, S.: 3D-R2N2: a unified approach for single and multi-view 3D object reconstruction. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9912, pp. 628–644. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46484-8_38

14. Deng, B., Genova, K., Yazdani, S., Bouaziz, S., Hinton, G.E., Tagliasacchi, A.: Cvxnets: Learnable convex decomposition (2020)

15. Eigen, D., Puhrsch, C., Fergus, R.: Depth map prediction from a single image using a multi-scale deep network. In: NeurIPS (2014)

16. Fan, H., Su, H., Guibas, L.J.: A point set generation network for 3D object reconstruction from a single image. In: CVPR (2017)

17. Ganapathi-Subramanian, V., Diamanti, O., Pirk, S., Tang, C., Niessner, M., Guibas, L.: Parsing geometry using structure-aware shape templates. In: 3DV (2018)

18. Girdhar, R., Fouhey, D.F., Rodriguez, M., Gupta, A.: Learning a predictable and generative vector representation for objects. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9910, pp. 484–499. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46466-4_29

19. Golovinskiy, A., Funkhouser, T.: Consistent segmentation of 3D models. Comput. Graph. **33**(3), 262–269 (2009)

20. Groueix, T., Fisher, M., Kim, V.G., Russell, B., Aubry, M.: AtlasNet: a papier-mâché approach to learning 3D surface generation. In: CVPR (2018)

21. Groueix, T., Fisher, M., Kim, V.G., Russell, B.C., Aubry, M.: A papier-mâché approach to learning 3D surface generation. In: CVPR (2018)

22. He, K., Gkioxari, G., Dollar, P., Girshick, R.: Mask R-CNN. In: ICCV (2017)

23. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR (2016)

24. He, Y., Zhu, C., Wang, J., Savvides, M., Zhang, X.: Bounding box regression with uncertainty for accurate object detection. In: CVPR (2019)

25. Henderson, P., Ferrari, V.: Learning single-image 3D reconstruction by generative modelling of shape, pose and shading. In: IJCV, October 2019
26. Hoiem, D., Efros, A.A., Hebert, M.: Recovering surface layout from an image. IJCV **75**(1), 151–172 (2007)
27. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: NeurIPS (2012)
28. Li, J., Xu, K., Chaudhuri, S., Yumer, E., Zhang, H., Guibas, L.: Grass: Generative recursive autoencoders for shape structures. In: ACM TOG (Proceedings of SIGGRAPH 2017), vol. 36(4) (2017)
29. Li, K., Pham, T., Zhan, H., Reid, I.: Efficient dense point cloud object reconstruction using deformation vector fields. In: ECCV (2018)
30. Li, Y., Pizlo, Z.: Reconstruction of 3D symmetrical shapes by using planarity and compactness constraints. J. Vis. **7**(9), 834–834 (2007)
31. Li, Z., Snavely, N.: Megadepth: learning single-view depth prediction from internet photos. In: CVPR (2018)
32. Lim, J.J., Pirsiavash, H., Torralba, A.: Parsing ikea objects: fine pose estimation. In: ICCV (2013)
33. Litany, O., Bronstein, A., Bronstein, M., Makadia, A.: Deformable shape completion with graph convolutional autoencoders. In: CVPR (2018)
34. Ma, L., Jia, X., Sun, Q., Schiele, B., Tuytelaars, T., Van Gool, L.: Pose guided person image generation. In: NeurIPS (2017)
35. Mandikal, P., Navaneet, K.L., Agarwal, M., Babu, R.V.: 3D-LMNet: latent embedding matching for accurate and diverse 3D point cloud reconstruction from a single image. In: Proceedings of the British Machine Vision Conference (BMVC) (2018)
36. Marr, D.: Vision: A Computational Investigation Into the Human Representation and Processing of Visual Information. Ph.D. thesis (1982)
37. McCormac, J., Handa, A., Leutenegger, S., Davison, A.J.: Scenenet RGB-D: Can 5m synthetic images beat generic imagenet pre-training on indoor segmentation? In: ICCV (2017)
38. Mescheder, L., Oechsle, M., Niemeyer, M., Nowozin, S., Geiger, A.: Occupancy networks: learning 3D reconstruction in function space. In: CVPR (2019)
39. Niu, C., Li, J., Xu, K.: Im2struct: Recovering 3D shape structure from a single RGB image. In: CVPR (2018)
40. Park, J.J., Florence, P., Straub, J., Newcombe, R., Lovegrove, S.: Deepsdf: learning continuous signed distance functions for shape representation. In: CVPR (2019)
41. Pauly, M., Mitra, N.J., Wallner, J., Pottmann, H., Guibas, L.J.: Discovering structural regularity in 3D geometry. In: ACM TOG, vol. 27 (2008)
42. Pavlakos, G., Zhu, L., Zhou, X., Daniilidis, K.: Learning to estimate 3D human pose and shape from a single color image. In: CVPR (2018)
43. Pizlo, Z.: 3D Shape: Its Unique Place in Visual Perception. MIT Press, Cambridge (2010)
44. Richter, S.R., Roth, S.: Matryoshka networks: predicting 3D geometry via nested shape layers. In: CVPR (2018)
45. Riegler, G., Osman Ulusoy, A., Geiger, A.: Octnet: learning deep 3D representations at high resolutions. In: CVPR (2017)
46. Riegler, G., Ulusoy, A.O., Bischof, H., Geiger, A.: Octnetfusion: learning depth fusion from data. In: 3DV (2017)
47. Roberts, L.G.: Machine perception of three-dimensional solids. Ph.D. thesis, Massachusetts Institute of Technology (1963)
48. Ronneberger, O., Fischer, P., Brox, T.: U-net: convolutional networks for biomedical image segmentation. In: MICCAI (2015)

49. Saxena, A., Sun, M., Ng, A.Y.: Make3d: learning 3D scene structure from a single still image. IEEE TPAMI **31**(5), 824–840 (2009)
50. Shin, D., Fowlkes, C.C., Hoiem, D.: Pixels, voxels, and views: a study of shape representations for single view 3D object shape prediction. In: CVPR (2018)
51. Sitzmann, V., Zollhöfer, M., Wetzstein, G.: Scene representation networks: continuous 3D-structure-aware neural scene representations. In: NeurIPS (2019)
52. Šťava, O., Beneš, B., Měch, R., Aliaga, D.G., Krištof, P.: Inverse procedural modeling by automatic generation of l-systems. Comput. Graph. Forum **29**, 665–674 (2010)
53. Sun, X., et al.: Pix3d: dataset and methods for single-image 3D shape modeling. In: CVPR (2018)
54. Tatarchenko, M., Dosovitskiy, A., Brox, T.: Octree generating networks: efficient convolutional architectures for high-resolution 3D outputs. In: ICCV (2017)
55. Tulsiani, S., Su, H., Guibas, L.J., Efros, A.A., Malik, J.: Learning shape abstractions by assembling volumetric primitives. In: CVPR (2017)
56. Tulsiani, S., Zhou, T., Efros, A.A., Malik, J.: Multi-view supervision for single-view reconstruction via differentiable ray consistency. In: CVPR (2017)
57. 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)
58. Wu, J., Wang, Y., Xue, T., Sun, X., Freeman, B., Tenenbaum, J.: Marrnet: 3D shape reconstruction via 2.5 d sketches. In: NeurIPS (2017)
59. Wu, J., et al.: 3D interpreter networks for viewer-centered wireframe modeling. In: IJCV (2018)
60. 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)
61. Wu, J., Zhang, C., Zhang, X., Zhang, Z., Freeman, W.T., Tenenbaum, J.B.: Learning shape priors for single-view 3D completion and reconstruction. In: ECCV (2018)
62. Wu, Z., Wang, X., Lin, D., Lischinski, D., Cohen-Or, D., Huang, H.: Structure-aware generative network for 3D-shape modeling. arXiv preprint arXiv:1808.03981 (2018)
63. Xu, Q., Wang, W., Ceylan, D., Mech, R., Neumann, U.: Disn: deep implicit surface network for high-quality single-view 3D reconstruction (2019)
64. Zhang, X., Zhang, Z., Zhang, C., Tenenbaum, J., Freeman, B., Wu, J.: Learning to reconstruct shapes from unseen classes. In: NeurIPS (2018)
65. Zhou, X., Zhu, M., Leonardos, S., Derpanis, K.G., Daniilidis, K.: Sparseness meets deepness: 3D human pose estimation from monocular video. In: CVPR (2016)
66. Zou, C., Yumer, E., Yang, J., Ceylan, D., Hoiem, D.: 3D-PRNN: generating shape primitives with recurrent neural networks. In: ICCV (2017)