Im2Struct: Recovering 3D Shape Structure from a Single RGB Image

Chengjie Niu   Jun Li   Kai Xu*
National University of Defense Technology

Abstract

We propose to recover 3D shape structures from single RGB images, where structure refers to shape parts represented by cuboids and part relations encompassing connectivity and symmetry. Given a single 2D image with an object depicted, our goal is automatically recover a cuboid structure of the object parts as well as their mutual relations. We develop a convolutional-recursive auto-encoder comprised of structure parsing of a 2D image followed by structure recovering of a cuboid hierarchy. The encoder is achieved by a multi-scale convolutional network trained with the task of shape contour estimation, thereby learning to discern object structures in various forms and scales. The decoder fuses the features of the structure parsing network and the original image, and recursively decodes a hierarchy of cuboids. Since the decoder network is learned to recover part relations including connectivity and symmetry explicitly, the plausibility and generality of part structure recovery can be ensured. The two networks are jointly trained using the training data of contour-mask and cuboid-structure pairs. Such pairs are generated by rendering stock 3D CAD models coming with part segmentation. Our method achieves unprecedentedly faithful and detailed recovery of diverse 3D part structures from single-view 2D images. We demonstrate two applications of our method including structure-guided completion of 3D volumes reconstructed from single-view images and structure-aware interactive editing of 2D images.

1. Introduction

The last few years have witnessed a continued interest in single-view image-based 3D modeling [2, 4, 5]. The performance of this task has been dramatically boosted, due to the tremendous success of deep convolutional neural networks (CNN) on image-based learning tasks [10]. The existing deep models, however, have so far been mainly targeting the output of volumetric representation of 3D shapes [2]. Such models are essentially learned to map an input 2D image to a 3D image (voxel occupancy of a 3D shape in a 3D

*Corresponding author: kevin.kai.xu@gmail.com

Figure 1: 3D shape structures (2nd column) recovered from photos of household objects (1st column). Top row: The inferred 3D shape structure can be used to complete and refine the volumetric shape estimated from the image using existing methods [22]. Bottom row: The structure is used to assist structure-aware image editing, where our cuboid structure is used as an editing proxy [25].

volume). Some compelling results have been demonstrated.

While enjoying the high capacity of deep models in learning the image-to-image mapping, the 3D volumes reconstructed by these methods lose an important information of 3D shapes – shape topology or part structure. Once a 3D shape is converted into a volumetric representation, it would be hard to recover its topology and structure, especially when there exist topological defects in the reconstructed volume. Shape structure, encompassing part composition and part relations, has been found highly important to semantic 3D shape understanding and editing [15]. Inferring a part segmentation for a 3D shape (surface or volumetric model) is known to be difficult [8]. Even if a segmentation is given, it is still challenging to reason about part relations such as connection, symmetry, parallelism, etc.

We advocate learning a deep neural network that directly recovers 3D shape structure of an object, from a single RGB image. The extracted structure can be used for enhancing the volumetric reconstruction obtained by existing methods, facilitating structure-aware editing of the reconstructed 3D shapes, and even enabling high-level editing of the input images (see Fig. 1). However, directly mapping an image
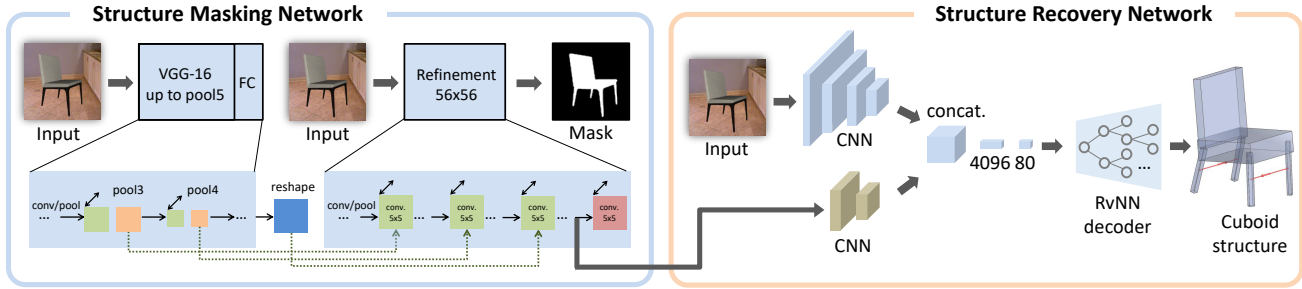
Figure 2: An overview of our network architecture. The structure masking network is a two-scale CNN which is trained to produce a contour mask for the object of interest. The structure recovery network first fuses the feature map of the masking network and the CNN feature of the original image, and decode the fused feature recursively into a box structure. The red arrows in the resultant chair structure (right most) indicate recovered reflectional symmetries between chair legs.

to a part structure seems a dunting task. Tulsiani et al. [19] proposed a deep architecture to map a 3D volume to a set of cuboid primitives. Their method, however, cannot be adapted to our problem setting since the output primitive set does not possess any structural information (mutual relations between primitives are not recovered).

Our problem involves the reasoning not only about shape geometry, but also for higher level information of part composition and relations. It poses several special challenges. 1) Different from shape geometry, part decomposition and relations do not manifest explicitly in 2D images. Mapping from pixels to part structure is highly ill-posed, as compared to pixel-to-voxel mapping studied in many existing 2D-to-3D reconstruction works. 2) Many 3D CAD models of man-made objects contain diverse, fine-grained substructures. A faithful recovery of those complicated 3D structures goes far beyond shape synthesis modulated by a shape classification. 3) Natural images always contain cluttered background and the imaged objects have large variations of appearance due to different textures and lighting conditions.

Human brains do well both in shape inference based on low-level visual stimulus and structural reasoning with the help of prior knowledge about 3D shape compositions. The strength of human perception is to integrate the two ends of processing and reasoning to form a capable vision system for high-level 3D shape understanding. Motivated by this, we propose to learn and integrate two networks, a *structure masking network* for accentuating multi-scale object structures in an input 2D image, followed by a *structure recovery network* to recursively recover a hierarchy of object parts abstracted by cuboids (see Figure 2).

The structure masking network produces a multi-scale attentional mask for the object of interest, thereby decerning its shape structures in various forms and scales. It designed as a multi-scale convolutional neural networks (CNN) augmented with jump connections to retain shape details while screening out the structure-irrelevant information such as

background and textures in the output mask image. The structure recovery network fuses the features extracted in the structure masking network and the CNN features of the original input image and feed them into a recursive neural network (RvNN) for 3D structure decoding [13]. The RvNN decoder, which is trained to explicitly model part relations, expands the fused image features recursively into a tree organization of 3D cuboids with plausible spatial configuration and reasonable mutual relations.

The two networks are jointly trained, with the training data of image-mask and cuboid-structure pairs. Such pairs can be generated by rendering 3D CAD models and extracting the box structure based on the given parts of the shape. Several mechanisms are devised to avoid overfitting in training this model. Experiments show that our method is able to faithfully recover diverse and detailed part structures of 3D objects from single 2D natural images. Our paper makes the following contributions:

- We propose to directly recover 3D shape structures from single RGB images. The faithful and detailed recovery of 3D structural information of an object, such as part connectivity and symmetries, from 2D images has never been seen before, to our knowledge.

- We present an architecture to tackle the hard learning task, via integrating a convolutional structure masking network and a recursive structure recovery network.

- We develop two prototype applications where we use the recovered box structures 1) to refine the 3D shapes reconstructed from single images by existing methods and 2) to assist structure-aware editing of 2D images.

2. Related work

Reconstructing 3D shapes from a single image has been a long-standing pursue in both vision and graphics fields. Due to its ill-posedness, many priors and assumptions have

been attempted, until the proliferation of high-capacity deep neural networks. We will focus only on those deep learning based models and categorize the fast-growing literature in three different dimensions.

Depth estimation vs. 3D reconstruction. Depth estimation is perhaps the most straightforward solution for recovering 3D information from single images. Deep learning has been shown to be highly effective for depth estimation [3, 11]. Compared to depth estimation, reconstructing a full 3D model is much more challenging due to the requirement of reasoning about the unseen parts. The latter has to resort to shape or structure priors. Using deep neural networks to encode shape priors of a specific category has received much attention lately, under the background of fast growing large 3D shape repositories [1]. Choy et al. [2] develop a 3D Recurrent Reconstruction Neural Network to generate 3D shapes in volumetric representation, given a single image as input. A point set generation network is proposed for generating from a 2D image a 3D shape in point cloud [4]. We are not aware of any previous works that can generate part-based structures directly from a single image.

Discriminative vs. Generative. For the task of 3D modeling from 2D images, discriminative models are learned to map an input image directly to the output 3D representation, either by a deep CNN for one-shot generation or a recurrent model for progressive generation [2]. The advantages of such approach include ease of training and high-quality results. With the recent development of deep generative models such as variational auto-encoder (VAE) [9], generative adversarial nets (GAN) [6] and their variants. Learning a generative model for 3D shape generation has gained extensive research [22, 5, 13]. For generative models, the input image can be used to condition the sampling from the predefined parameter space or learned latent space [22, 5]. Generative models are known hard to train. For the task of cross-modality mapping, we opt to train a discriminative model with a moderate size of training data.

Geometry reconstruction vs. Structure recovery. The existing works based on deep learning models mostly utilize volumetric 3D shape representation [22, 5]. Some notable exceptions include generating shapes in point clouds [4], cuboid primitives [19] and manifold surfaces [14]. However, none of these representations contains structural information of parts and part relations. Interestingly, structure recovery is only studied with non-deep-learning approaches [24, 17, 7]. This is largely because of the lack of a structural 3D shape representation suitable for deep neural networks. Recently, Li et al. [13] propose to use recursive neural networks for structural representation learning,

nicely addressing the encoding/decoding of arbitrary number of shape parts and various types of part relations. Our method takes the advantage of this and integrate it into a cross-modality mapping architecture for structure recovery from an RGB image.

3. Method

We introduce our architecture for learning 3D shape structures from single images. It is an auto-encoder composed of two sub-networks: a *structure masking network* for decerning the object structures from the input 2D image and a *structure recovery network* for recursive inference of a hierarchy of 3D boxes along with their mutual relations.

3.1. Network architecture

Our network is shown in Fig. 2, which is composed of two modules: a two-scale convolutional structure masking network and a recursive structure recovery network. The structure masking network is trained to estimate the contour of the object of interest. This is motivated by the observation that object contours provides strong cues for understanding shape structures in 2D images [16, 18]. Instead of utilizing the extracted contour mask, we feed the feature map of the last layer of the structure masking network into the structure recovery network. To retain more information in the original image, this feature is fused with the CNN feature of the input image via concatenation and fully connected layers, resulting in a 80D feature code. An RvNN decoder then recursively unfolds the feature code into a hierarchical organization of boxes, with plausible spatial configuration and mutual relations, as the recovered structure.

3.2. Structure Masking Network.

Our structure masking network is inspired by the recently proposed multi-scale network for detailed depth estimation [12]. Given an input RGB image rescaled to 224×224, we design a two-scale structure masking network to output a binary contour mask with a quarter of the input resolution (56×56). The first scale captures the information of the whole image while the second produces a detailed mask map at a quarter of the input resolution. As our prediction target is a binary mask, we use the SoftMax Loss as our training loss.

We employ VGG-16 to initialize the convolutional layers (up to pool5) of the first scale network, followed by two fully connected layers. The feature maps and outputs of the first scale network are fed into various layers of the second scale one for refined structure decerning. The second scale network, as a refinement block, starts from one 9×9 convolution and one pooling over the original input image, followed by nine successive 5×5 convolutions without pooling. The feature maps from the pool3, pool4 and the output of last fully connected layer of first scale network are fused

into the second, the fourth and the sixth convolutional layer of the second scale network, respectively. All the feature fusions get through a jump connections layer, which has a 5×5 convolutional layer and a 2x or 4x up-sampling to match the 56×56 feature map size in the second scale; the jump connection from the fully connected layer is a simple concatenation. It is shown that jump connections help extracting detailed structures from images effectively [12].

3.3. Structure Recovery Network

The structure recovery network integrates the features extracted from the structure masking network and for the input image into a bottleneck feature and recursively decodes it into a hierarchy of part boxes.

Feature fusion. We fuse features from two convolutional channels. One channel takes as input the feature map of the structure masking network (the last feature map before the mask prediction layer), followed by two convolutions and poolings. Another channel is the CNN feature of the original image extracted by a VGG-16. The output feature maps from the two channels are then concatenated with size 7×7, and further encoded into a 80D code after two fully connected layers, capturing the object structure information from the input image. We found through experiments such fused features not only improve the accuracy of structure recovery, but also attain good domain-adaption from rendered images to real ones. We believe the reason is that the extracted features for mask prediction task retain shape details through factoring them out of background clutters, texture variations and lighting conditions. Since it is hard for the masking network to produce perfect mask prediction, the CNN feature of the original image provides complimentary information via retaining more object information.

Structure decoding. We adopt a recursive neural network (RvNN) as box structure decoder like in [13]. Starting from a root feature code, RvNN recursively decodes its 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. Decoder ADJDEC splits a parent code p into two child codes c_1 and c_2, using the mapping function:

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

where $W_{ad} \in \mathbb{R}^{2n \times n}$ and $b_{ad} \in \mathbb{R}^{2n}$. $n = 80$ is the dimension of a non-leaf node.
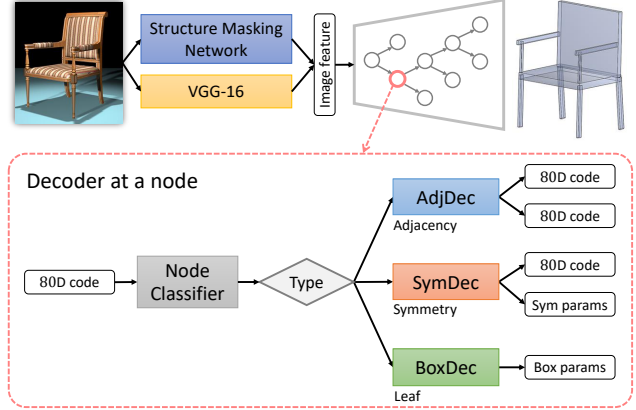

Figure 3: Recursive decoding of a 3D box structure from a 2D image feature (top) and an illustration of decoder network at a given node (bottom).

Symmetry decoder. Decoder SYMDEC recovers a symmetry group in the form of a symmetry generator (a node code c) and a vector of symmetry parameters s:

$$[c \, s] = \tanh(W_{sd} \cdot p + b_{sd})$$

where $W_{sd} \in \mathbb{R}^{(n+m) \times n}$, and $b_{sd} \in \mathbb{R}^{m+n}$. We use $m = 8$ for symmetry parameters consisting of: symmetry type (1D); number of repetitions for rotational and translational symmetries (1D); and the reflectional plane for reflective symmetry, rotation axis for rotational symmetry, or position and displacement for translational symmetry (6D).

Box decoder. Decoder BOXDEC converts the code of a leaf node to a 12D box parameters defining the center, axes and dimensions of a 3D oriented box, similar to [13].

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

where $W_{ld} \in \mathbb{R}^{12 \times n}$, and $b_{ld} \in \mathbb{R}^{12}$.

The decoders are recursively applied during decoding. The key is how to determine the type of a node so that the corresponding decoder can be used at the node. This is achieved by learning a node classifier based on the training task of structure recovery where the ground-truth box structure is known for a given training pair of image and shape structure. The node classifier is jointly trained with the three decoders. The process of structure decoding is illustrated in Fig. 3. In our implementation, the node classifier and the decoders for both adjacency and symmetry are two-layer networks, with the hidden layer and output layer being 200D and 80D vectors, respectively.

3.4. Training details

There are two stages in the training. First, we train the structure masking network to estimate a binary object mask for the input image. The first and the second scale of the structure masking network are trained jointly. In the next, we jointly refine the structure masking network and train the structure recovery network, during which a low learning rate for structure masking network is used. The structure recovery loss is computed as the sum of the box reconstruction error and the cross entropy loss for node classification. The reconstruction error is calculated as the sum of squared differences between the input and output parameters for each box and symmetry node. Prior to training, all 3D shapes are resized into a unit bounding box to make the reconstruction error comparable across different shapes. In Fig. 4 (top), we plot the training and testing losses for box reconstruction, symmetry recovery and node classification, respectively, demonstrating the convergence of our structure recovery network.

We use the Stochastic Gradient Descent (SGD) to optimize our structure recovery network with back-propagation through structure (BPTT) for the RvNN decoder training. The convolutional layers of VGG-16 are initialized with the parameters pre-trained over ImageNet; all the other convolutional layers, the fully connected layers and the structure recovery network are randomly initialized. The learn-

ing rate of the structure masking network is 10^{-4} for pre-training and 10^{-5} for fine-tuning. During joint training, the learning rate is 10^{-3} for structure masking network, 0.2 for RvNN decoder and 0.5 for RvNN node classifier. These learning rates are decreased by a factor of 10 for every 50 epochs. Our network is implemented with Matlab based on the MatConvNet toolbox [20]. The details on generating training data is provided in Section 4.1.

4. Experiments

We collected a dataset containing 800 3D shapes from three categories in ShapeNet: chairs (500), tables (200), aeroplanes (100). The dataset is split into two subsets for training(70%) and testing (30%), respectively. With these 3D shapes, we generate training and testing pairs of image mask and shape structure to train the network and evaluate our method quantitatively. We also evaluate our methods qualitatively with a Google image search challenge. Both quantitative and qualitative evaluations demonstrate the capability of our method in recovering 3D shape structures from single RGB images faithfully and accurately.

4.1. Training data generation

Image-structure pair generation. For each 3D shape, we create 36 rendered views around the shape for every $30°$ rotation and with 3 elevations. Plus another 24 randomly generated views, we create 60 rendered RGB images in total for each shape. The 3D shapes are rendered with randomly selected backgrounds from NYU v2 dataset. For each RGB image, the ground-truth object mask can be easily extracted using the depth buffer for rendering.

All 3D shapes in our dataset are pre-segmented based on their original mesh components or using the symmetry-aware segmentation proposed in [21]. We utilize symmetry hierarchy [21] to represent the shape structure, which defines how parts in a shape are recursively assembled by connectivity or grouped by symmetry. We adopt the method in [13] to infer consistent hierarchy trees for the shapes of each category. Specifically, we train a unsupervised auto-encoder with the task of self-reconstruction for all shapes. During testing, we use this auto-encoder to perform a greedy search of grouping hierarchy for each shape. For more details on this process, please refer to the original work. Consequently, we generate 60 image-structure pairs for each 3D shape.
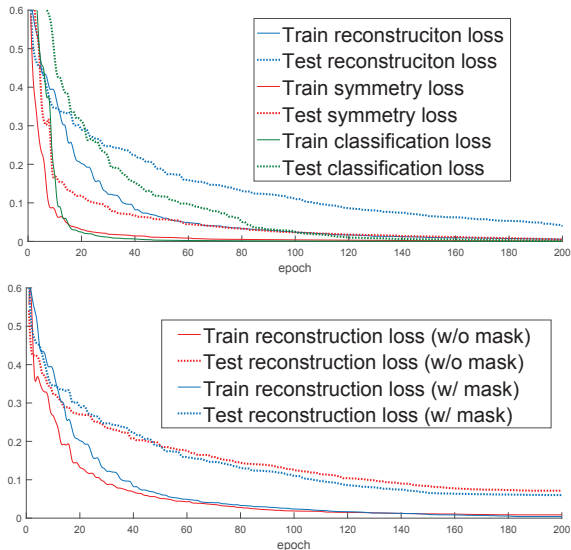

Figure 4: The convergence of the structure recovery network. We show in the top plot the training and testing losses for cuboid reconstruction, symmetry parameter recovery and node classification, respectively. The bottom plot shows that our method with structure masking network yields lower reconstruction loss than without it.

Data processing and augmentation. To further enhance our dataset and alleviate overfitting, we conduct on each training 3D shape structure-aware deformation [23] based on component-wise controllers [26] to generate a set of structurally plausible variations for the training shape. Such structure-aware deformation preserves the connection and

Figure 5: Object mask predictions for six sample images from the Internet. The probability of object mask is indicated by the brightness in the grey-scale output.

symmetry relations between shape parts, while maintaining the shape texture for each part. This step is fully automatic and the parameters for each variation generation is randomly set within a given range. In our implementation, we randomly generate 20 new variations for each 3D shape, thus enlarging our database to 16K 3D shapes. For the input images (and the corresponding object masks), we employ the common operations for image data augmentation [12] such as color perturbation, contrast adjustment, image flip and transformation, etc.

4.2. Results and evaluation

We first show in Fig. 5 some results of object mask prediction by our structure masking network. As can be seen in the output, the background clutters are successfully filtered out and some detailed structures of the objects are captured.

Google image challenge for structure recovery. We first perform a qualitative evaluation on the capability and versatility of our structure recovery. In order for a more objective study, instead of cherry-picking a few test images, we opt to conduct a small-scale stress test with a Google image challenge [24]. During the test, we perform text-based image search on Google using the keywords of "chair", "table" and "airplane", respectively. For each search, we try to recover a 3D cuboid structure for each of the top 8 returned images using our method.

The results are shown in Fig. 6. From the results, we can see that our method is able to recover 3D shape structures from real images in a detailed and accurate way. More importantly, our method can recover the connection and symmetry relations of the shape parts from single view inputs,

leading to high quality results with coherent and plausible structure. Examples of symmetry recovery include the reflectional symmetry of chair legs or airplane wings, the rotational symmetry of legs in a swivel chair or a table.

There are some failure cases (marked with red boxes in Fig. 6). The marked chair example is not even composed of multiple parts and hence may not admit a part structure. When the structure of the object of interest is unseen from our training dataset of 3D shapes, such as the marked table example, our method fails to recover a reasonable structure.

Quantitative evaluation. We quantitatively evaluate our algorithm with our test dataset. For the structure masking network, we evaluate the mask accuracy by the overall pixel accuracy and per-class accuracy against the ground-truth mask (see table 1). We provide a simple ablation study by comparing our method with two baselines: single-scale (without refinement network) and two-scale (without jump connection). The results demonstrate the effectiveness of our multi-scale masking network.

Method	Overall Pixel	Per-Class
single-scale	0.953	0.917
two-scale (w/o jump)	0.982	0.964
two-scale (with jump)	0.988	0.983

Table 1: An ablation study of structure masking network.

For 3D shape structure recovery, we develop two measures to evaluate the accuracy:

- *Hausdorff Error*: $\frac{1}{2T}\sum_i^T (D(S_i, S_i^{\text{gt}}) + D(S_i^{\text{gt}}, S_i))$, where S_i is a recovered shape structure (represented by a set of boxes) and S_i^{gt} its corresponding ground-truth. T is the number of models in the test dataset. $D(S_1, S_2) = \frac{1}{n}\sum_{B_j^1 \in S_1} \min_{B_k^2 \in S_2} H(B_j^1, B_k^2)$ measures the averaged minimum Hausdorff distance from the boxes in structure S_1 to those in S_2, where B_j^1 and B_k^2 represent the boxes in S_1 and S_2, respectively. $H(B^1, B^2) = \max_{p \in B^1} \min_{q \in B^2} ||p - q||$ is the Hausdorff distance between two boxes, with p and q being the corner points of box. Since Hausdorff is asymmetric, the distance is computed for both directions and averaged.

- *Thresholded Accuracy*: The percentage of boxes B_i such that $\delta = H(B_i, B_i^*)/L(B_i^*) < threshold$, where B_i is the i-th box in recovered shape structure S and B_i^* its nearest box in the ground-truth S^{gt}. H is the Hausdorff distance between two boxes as defined above. L is the diagonal length of a box.

We consider as our baseline where the structure masking network is simply a vanilla VGG-16 network. In table 2, we compare the accuracy of structure recovery, based

Figure 6: Google image search challenge for 3D shape structure recovery. We perform text-based image search on Google with keywords "chair", "table" and "airplane", respectively. For each search, we run our method on the top 8 returned images (duplicate or very similar images are removed). Failure cases are marked with red boxes.

on the above two measures, for our method and the baseline. We also compare the two methods where VGG-16 is replaced with VGG-19. The results demonstrate the significant effect of our structure masking network in helping the structure decoding. This can also be observed from the reconstruction error plotted in Figure 4 (bottom). A deeper structure masking network (with VGG-19) also boosts the performance to a certain degree.

Method	Hausdorff Error	Thresholded Acc.	
		$\delta < 0.2$	$\delta < 0.1$
Vanilla VGG-16	0.0980	96.8%	67.8%
Structure masking (VGG-16)	0.0894	97.8%	75.3%
Vanilla VGG-19	0.0922	96.4%	72.2%
Structure masking (VGG-19)	0.0846	97.6%	78.5%

Table 2: Comparison of structure recovery accuracy over different methods.

Comparison. In Fig. 7, we give a visual comparison of 3D shape reconstruction from single-view images between our method and two state-of-the-art methods, [7] and [19]. Both the two alternatives produce part-based representation of 3D shapes, making them comparable to our method. The method by Huang et al. [7] recovers 3D shapes through assembling parts from database shapes while preserving their symmetry relations. The method of Tulsiani et al. [19] generates cuboid representation similar to ours, but does not produce symmetry relations. As can be seen, our method produces part structures which are more faithful to the input, due to the integration of the structure masking network, and meanwhile structurally more plausible, benefiting from our part relation recovery.

5. Applications

We develop two prototype applications to demonstrate the utility of the recovered shape structure in structure-aware shape editing and processing.

Figure 7: Comparing single-view, part-based 3D shape reconstruction between our Im2Struct and two alternatives.

Figure 8: Examples of structure-aware image editing. Given an input image (the left column), we show the edited structures of 3D cuboids along with the edited images.

Structure-aware image editing. In [25], a structure-aware image editing is proposed, where a cuboid structure is manually created for the object in the image to assist a plausible shape deformation. With our method, this cuboid structure can be automatically constructed. Moreover, the part relations are also recovered which can be used to achieve structure-aware editing. Given an RGB image, we first recover the 3D shape structure of the object of interest using our method. To align the inferred cuboid structure with the input image, we train another network to estimate the camera view. The 3D cuboids are then projected to the image space according to the estimated view. The object in the image is segmented with a CRF-based method constrained with the cuboid projections [24]. Each segment is assigned to a 3D cuboid based their image-space overlapping. At this point, the image editing method in [25] can be employed to deform the object of interest. Fig. 1 and 8 show a few examples of structure-aware image editing based on our 3D shape structure recovery.

Structure-assisted 3D volume refinement. A common issue with 3D reconstruction with volumetric shape repre-
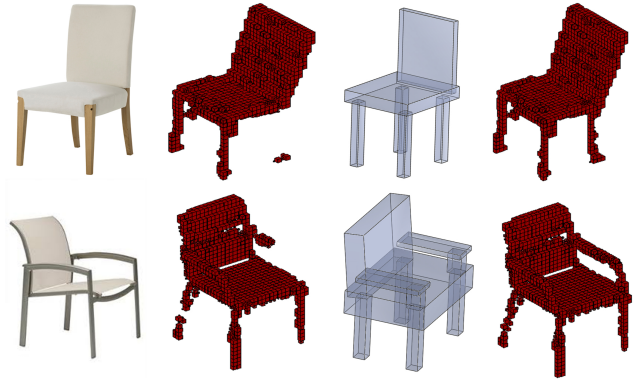
Figure 9: Part symmetry induced volume refinement. Given a 2D image, a volumetric shape is recovered using 3D-GAN [22] where the missing voxels break the symmetry of the object. Our recovered 3D structure helps complete the volume based on part symmetry relation.

sentation is that the resolution of volume is greatly limited due to the high computational cost. This usually results in missing parts and hence broken structure in the reconstructed volume. Our recovered 3D structures can be used to refine the 3D volumes estimated by existing approaches such as 3D-GAN [22]. Given a 2D image, a 3D volume is estimated with 3D-GAN and a cuboid structure recovered by our method. They can be easily aligned with the help of camera view estimation (as have been done above). Each voxel is assigned to the closest cuboid, leading to a part-based segmentation of the volume. We then utilize the part symmetry relation in our recovered structure to complete the missing voxels; see results in Fig. 9.

6. Conclusion

We have proposed a deep learning framework that directly recovers 3D shape structures from single 2D images. Our network joins a structure masking network for decerning the object structure and a structure recovery network for inferring 3D cuboid structure. The recovered 3D structures achieve both fidelity with respect to the input image and plausibility as a 3D shape structure. To the best of our knowledge, our work is the first that recovers detailed 3D shape structures from single 2D images.

Our method fails to recover structures for object categories unseen from the training set. For such cases, it would be interesting to learn an incremental part assembler. Our method currently recovers 3D cuboids only but not the underlying part geometry. A worthy direction is to synthesize detailed part geometry matching the visual appearance of the input image. Another interesting topic is to study the profound correlation between 2D features and 3D structure, so as to achieve a more explainable 3D structure decoding.

References

[1] A. X. Chang, T. Funkhouser, L. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015. 3

[2] C. B. Choy, D. Xu, J. Gwak, K. Chen, and S. Savarese. 3d-r2n2: A unified approach for single and multi-view 3d object reconstruction. In *European Conference on Computer Vision*, pages 628–644. Springer, 2016. 1, 3

[3] D. Eigen and R. Fergus. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2650–2658, 2015. 3

[4] H. Fan, H. Su, and L. Guibas. A point set generation network for 3d object reconstruction from a single image. *arXiv preprint arXiv:1612.00603*, 2016. 1, 3

[5] R. Girdhar, D. F. Fouhey, M. Rodriguez, and A. Gupta. Learning a predictable and generative vector representation for objects. In *European Conference on Computer Vision*, pages 484–499. Springer, 2016. 1, 3

[6] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014. 3

[7] Q. Huang, H. Wang, and V. Koltun. Single-view reconstruction via joint analysis of image and shape collections. *ACM Transactions on Graphics (TOG)*, 34(4):87, 2015. 3, 7

[8] E. Kalogerakis, M. Averkiou, S. Maji, and S. Chaudhuri. 3d shape segmentation with projective convolutional networks. *arXiv preprint arXiv:1612.02808*, 2016. 1

[9] D. P. Kingma and M. Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 3

[10] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012. 1

[11] I. Laina, C. Rupprecht, V. Belagiannis, F. Tombari, and N. Navab. Deeper depth prediction with fully convolutional residual networks. In *3D Vision (3DV), 2016 Fourth International Conference on*, pages 239–248. IEEE, 2016. 3

[12] J. Li, R. Klein, and A. Yao. A two-streamed network for estimating fine-scaled depth maps from single rgb images. In *ICCV*, 2017. 3, 4, 6

[13] J. Li, K. Xu, S. Chaudhuri, E. Yumer, H. Zhang, and L. Guibas. Grass: Generative recursive autoencoders for shape structures. *arXiv preprint arXiv:1705.02090*, 2017. 2, 3, 4, 5

[14] Z. Lun, M. Gadelha, E. Kalogerakis, S. Maji, and R. Wang. 3d shape reconstruction from sketches via multi-view convolutional networks. *arXiv preprint arXiv:1707.06375*, 2017. 3

[15] N. Mitra, M. Wand, H. R. Zhang, D. Cohen-Or, V. Kim, and Q.-X. Huang. Structure-aware shape processing. In *SIGGRAPH Asia 2013 Courses*, page 1. ACM, 2013. 1

[16] J. Shotton, A. Blake, and R. Cipolla. Contour-based learning for object detection. In *Proc. ICCV*, volume 1, pages 503–510. IEEE, 2005. 3

[17] H. Su, Q. Huang, N. J. Mitra, Y. Li, and L. Guibas. Estimating image depth using shape collections. *ACM Transactions on Graphics (TOG)*, 33(4):37, 2014. 3

[18] A. Toshev, B. Taskar, and K. Daniilidis. Shape-based object detection via boundary structure segmentation. *International journal of computer vision*, 99(2):123–146, 2012. 3

[19] S. Tulsiani, H. Su, L. J. Guibas, A. A. Efros, and J. Malik. Learning shape abstractions by assembling volumetric primitives. *arXiv preprint arXiv:1612.00404*, 2016. 2, 3, 7

[20] A. Vedaldi and K. Lenc. Matconvnet: Convolutional neural networks for matlab. In *Proceedings of the 23rd ACM international conference on Multimedia*, pages 689–692. ACM, 2015. 5

[21] Y. Wang, K. Xu, J. Li, H. Zhang, A. Shamir, L. Liu, Z. Cheng, and Y. Xiong. Symmetry hierarchy of man-made objects. *Computer Graphics Forum*, 30(2):287–296, 2011. 5

[22] J. Wu, C. Zhang, T. Xue, B. Freeman, and J. Tenenbaum. Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling. In *Advances in Neural Information Processing Systems*, pages 82–90, 2016. 1, 3, 8

[23] K. Xu, H. Zhang, D. Cohen-Or, and B. Chen. Fit and diverse: set evolution for inspiring 3d shape galleries. *ACM Transactions on Graphics (TOG)*, 31(4):57, 2012. 5

[24] K. Xu, H. Zheng, H. Zhang, D. Cohen-Or, L. Liu, and Y. Xiong. Photo-inspired model-driven 3d object modeling. *ACM Transactions on Graphics (TOG)*, 30(4):80, 2011. 3, 6, 8

[25] Y. Zheng, X. Chen, M.-M. Cheng, K. Zhou, S.-M. Hu, and N. J. Mitra. Interactive images: Cuboid proxies for smart image manipulation. *ACM Transactions on Graphics*, 31(4):99:1–99:11, 2012. 1, 8

[26] Y. Zheng, H. Fu, D. Cohen-Or, O. K.-C. Au, and C.-L. Tai. Component-wise controllers for structure-preserving shape manipulation. In *Computer Graphics Forum*, volume 30, pages 563–572. Wiley Online Library, 2011. 5