SLICENET: SLICE-WISE 3D SHAPES RECONSTRUCTION FROM SINGLE IMAGE

Yunjie Wu, Zhengxing Sun, Youcheng Song, Yunhan Sun, Jinlong Shi

[1,2,3]State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing, P R China
[4,5]Department of Computer, Jiangsu University of Science and Technology, Zhengjiang, P R China
[2]szx@nju.edu.cn, [5]jlshifudan@gmail.com

ABSTRACT

3D object reconstruction from a single image is a highly ill-posed problem, requiring strong prior knowledge of 3D shapes. Deep learning methods are popular for this task. Especially, most works utilized 3D deconvolution to generate 3D shapes. However, the resolution of results is limited by the high resource consumption of 3D deconvolution. In this paper, we propose SliceNet, sequentially generating 2D slices of 3D shapes with shared 2D deconvolution parameters. To capture relations between slices, the RNN is also introduced. Our model has three main advantages: First, the introduction of RNN allows the CNN to focus more on local geometry details,improving the results' fine-grained plausibility. Second, replacing 3D deconvolution with 2D deconvolution reducs much consumption of memory, enabling higher resolution of final results. Third, an slice-aware attention mechanism is designed to provide dynamic information for each slice's generation, which helps modeling the difference between multiple slices, making the learning process easier. Experiments on both synthesized data and real data illustrate the effectiveness of our method.

Index Terms— Image-based 3D Reconstruction, RNN, Slice-Aware Attention Model

1. INTRODUCTION

Recovering 3D shapes from single 2D image is one of the central problems of computer vision. To eliminate the ambiguity of 2D image, some priors are necessary during the 3D reconstruction. Recently, researchers tried to tackle this problem with deep learning. Deep neural networks are employed to learn the priors from corresponding 2D images and 3D shapes. Most works relies the encoder-decoder architecture [1, 2, 3, 4, 5] to predict a 3D shape from input image. Especially, the voxel representation is commonly used as it's convolutional friendly. To generate a 3D voxel with neural network, 3D deconvolution layer is a natural choice [1, 6, 7, 8]. However, there exist some problems with the 3D deconvolution. First of all, 3D deconvolution consumes high resource ($O\left(n^3\right)$), leading to a limitation of resolution for the reconstructed 3D voxel. What's more, to generating a globally plausible 3D shape demands the deconvolution block a large receptive filed, which may reduce the network's concentration on the local geometry details. There have been some works attempting to solve the problem of 3D deconvolution. A common idea is employing 3D deconvolution in an octree form, predicting coarse-to-fine 3D voxels [5, 9]. While in this paper, we propose a different method, performing repeated forward process of same 2D deconvolution to produce multiple 2D slices, which could be used to make up the final 3D voxel.

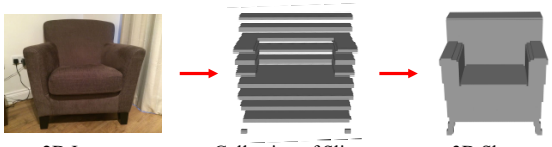

Fig. 1: An example of 3D shape and 2D slices. It could be seen that, 1) the symmetry maintains among each slice as in 3D shape 2) most adjacent slices show smooth variety.

Our key observation is that, the 2D slices of 3D shape usually share many common patterns. As shown in Fig.1, if a 3D shape is symmetrical, the symmetry would maintain among all 2D slices of this shape. If a 3D shape is plausible, then adjacent slices would be similar and the varieties between them ought to be smooth. So, it's possible to model these 2D slices with shared-weight 2D deconvolution block, avoiding the use of 3D deconvolution with much more parameters. Based on this observation, we propose the SliceNet. With single input image, SliceNet predicts 2D slices sequentially. Apart from individual patterns of slices, the relations between them are also necessary to make them consistent and final 3D shape plausible. To capture such global dependencies, the LSTM is introduced, enabling our model to remember previous slices and keep the coherence of the slice sequence. We also design a slice-aware attention module to dynmically produce specific input for each specific slice's generation, making it more eas-

This work was supported by National High Technology Research and Development Program of China (No. 2007AA01Z334), National Natural Science Foundation of China (Nos. 61321491 and 61272219),National Key Research and Development Program of China(Nos. 2018YFC0309100, 2018YFC0309104), the China Postdoctoral Science Foundation (Grant No. 2017M621700) and Innovation Fund of State Key Laboratory for Novel Software Technology (Nos. ZZKT2018A09).

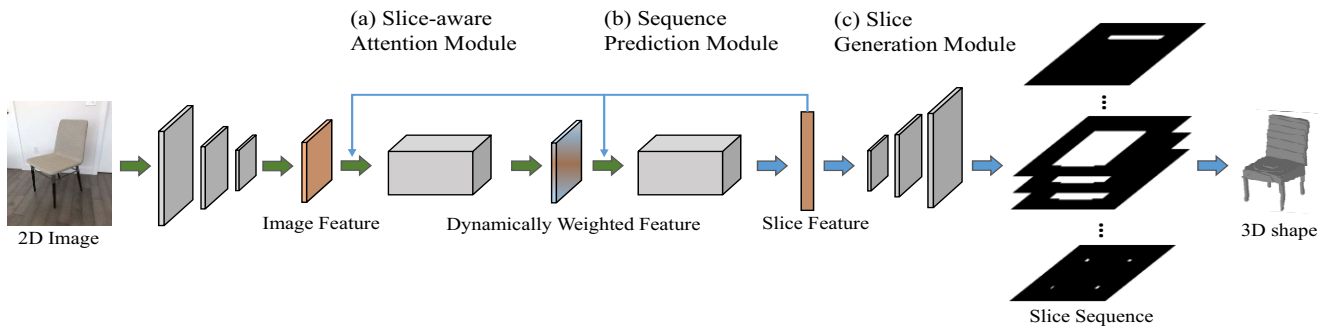

Fig. 2: Our model (SliceNet) has three major components: (a) slice-aware attention module (SAM) (b) sequence prediction module (SPM) and (c) slice generation module (SGM).

ier to model the difference between slices.

Actually there have already been some works attemped to employ both CNN and RNN [10, 11, 12, 13, 14]. While most of these works foucs on discriminant tasks such as sentence classification or emotion recognition, while our work foucs on 3D shape reconstruction, a generative task. The most relevant method to ours is 3D-R2N2 [1], which introduces LSTM to fusion multiple input images for 3D reconstruction. The main difference between our work and theirs is the role LSTM plays. We introduce the LSTM for sequential predictions, while their LSTM is designed to fusion multiple inputs.

This paper's contributions are three-fold. Firstly, we present a novel network sequentially outputs 2D slices with shared-weight 2D deconvolution for 3D reconstruction; Secondly, we design a slice-aware attention mechanism, enabling our model to extract relevant and meaningful information for each slice. Thirdly, we demonstrate our work's effectiveness on both synthesized and real data.

2. METHOD

Our method takes an encoder-decoder architecture. The encoder is a Resnet-18 [15]. The decoder consists of three main components, as illustrated in Fig.2: (a) A slice-aware attention module to produce dynamic feature map according to current slice. (b) A sequence prediction module to predict feature vector of slice one by one. (c) A slice generation module to generate difference slice at each step.

2.1. Slice-Aware Attention Module

To perform a slice-wise reconstruction, it's necessary for the model to capture relevant information at each slice step. It's obvious that, 1) not all pixels of the input image are meaningful for reconstruction, e.g. the background regions. 2) for different individual slice, the distribution of their highly-relevant features may vary.

Based on these observations, we introduce our slice-aware attention module (SAM) to produce customized features for each slice (Fig.2a). Inspired by SCA-CNN [16], the SAM contains two branchs: spatial-wise attention and channel-wise

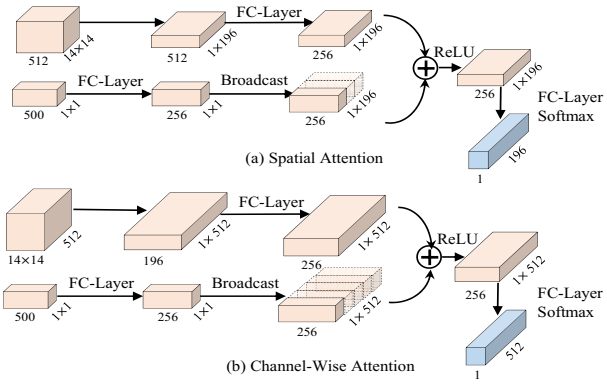

Fig. 3: The proposed SAM.

attention. Details of SAM are shown in Fig.3. The SAM yeilds a slice-aware feature map for further generation.

2.2. Sequence Prediction Module

To make the final results plausible, it's necessary to learn the dependencies between slices. That requires the model to remember slices that already generated when it perform current slice's prediction. Therefore, we propose the sequence prediction module (SPM). As illustrated in Fig.2b, it works in a recurrent way, taking features of both input image and previous slice as input, then predicts the feature of the next slice. The sequence prediction module is made up of two LSTM layers, each LSTM layer has 500 units.

2.3. Slice Generation Module

The final part of our model is slice generation module(SGM). It is responsible for producing slices' contents according to their features provided by SPM. As the SPM predict the features with taking all previous slices into consideration, the sequence consistency is maintained and the SGM could focus on slice's local details.

The architecture of SGM is inspired by the DCGAN [17]. It contains five 2D deconvolution layers with batch normalization and ReLU added in between. The kernel size of them is 4x4 and the stride is 2. The channels of them are

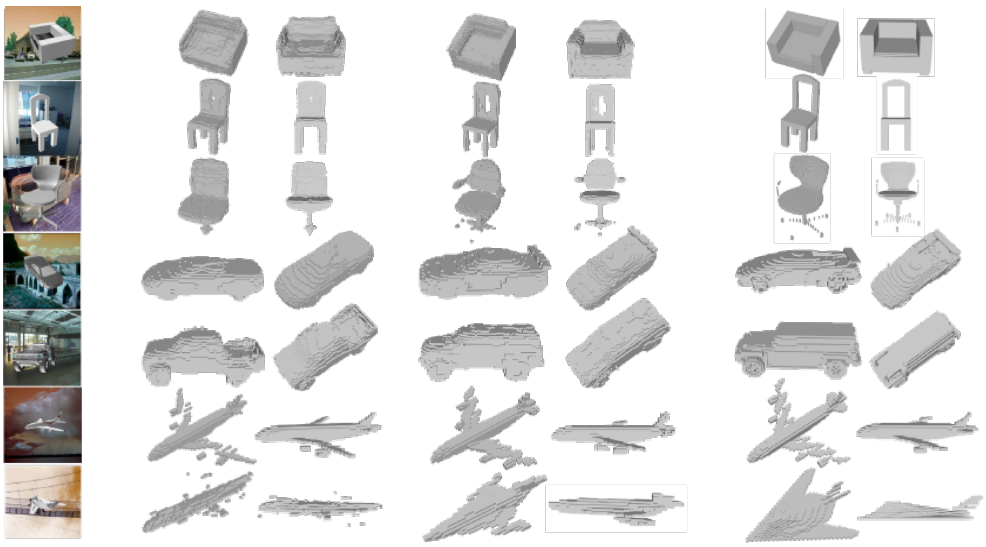

Fig. 4: Our results on single-image 3D reconstruction, compared with the MarrNet [6]. Our results contain more details than MarrNet (e.g. the the clear contour of the sofa in row 1, the tail of the sports car in row 4 and the wing of the airplane in row 7).

{512,256,128,64,32}. A Sigmoid layer is added after the final deconvolution.

3. EXPERIMENTS

In this section, we present both qualitative and quantitative results on single-image 3D reconstruction using our frameworks. We also compare our models with state-of-the-art and make some experimental analysis.

3.1. Setup

For training data, we select the ShapeNet [18]. We use the train, test split provided by [1]. Each shape is rendered from 24 different viewpoints. A random background from the SUN database [19] is applied. Two different voxel resolutions (64, 256) are selected for experiments. The 64 resolution is commonly used, while 256 is beyond most methods' reach. For comparison, we select a simple encoder-decoder with normal 3D deconvolution and two SOTA voxel-based methods 3D-R2N2 [1] and MarrNet [6].

3.2. Results on synthesized data

Qualitative results. First, we present the results of single-image 3D reconstruction performed on the ShapeNet synthesized images. Some examples are shown in Fig.4. It could be seen that most of our results show more geometry details, making the final results high-quality and plausible.

Quantitative comparisons. The quantitative results are shown in Table 1. From the table, we can conclude that our slice-wise reconstruction outperforms state-of-the-arts 3D-deconvolution based methods.

	chair		airplane		car	
	Voxel Resolution					
Methods	64	256	64	256	64	256
3D Deconv	.298	-	.358	-	.430	-
3D-R2N2	.328	-	.343	-	.519	-
MarrNet	.363	-	.640	-	.648	-
Ours	**.370**	**.120**	**.773**	**.627**	**.661**	**.591**

Table 1: Results for single-image 3D reconstruct tion on ShapeNet synthesized data. The Intersection over Union is computed for two voxel resolution: 64 and 256.

	Chair	Airplane	Car
No-attention	.178	.242	.371
S-attention	.306	.748	.649
C-attention	.213	.549	.438
GRU-1-layer	.321	.747	.651
GRU-2-layer	.327	.756	.653
LSTM-1-layer	.352	.761	.658
LSTM-2-layer	.370	.773	.661

Table 2: Reconstruction performance of SliceNet variations according to mean IoU.

Ablation study. To evaluate the effectiveness of our model's module and the impact of different parameters, an ablation study is conducted. Seven variations of our models are tested. Table 2 shows the results. We observe that with two-layer LSTM and both spatial and channel attention module, the model could achieve the best performance.

High-resolution reconstruction results. To show the benefits of being able to perform higher-resolutions reconstruction, we present some results of both 64 and 256 resolution in Figure.5. It could be observed that the results show more geometry details and plausibility at a higher resolution.

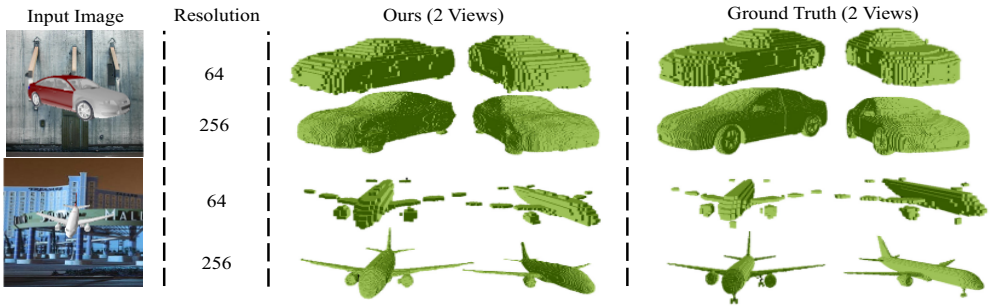

Fig. 5: Examples of our different resolution reconstruction results. From left to right: input, the size of voxel resolution, two views of our predictions and ground truth. It's clear the 256 resolution results outperform the 64 resolution results.

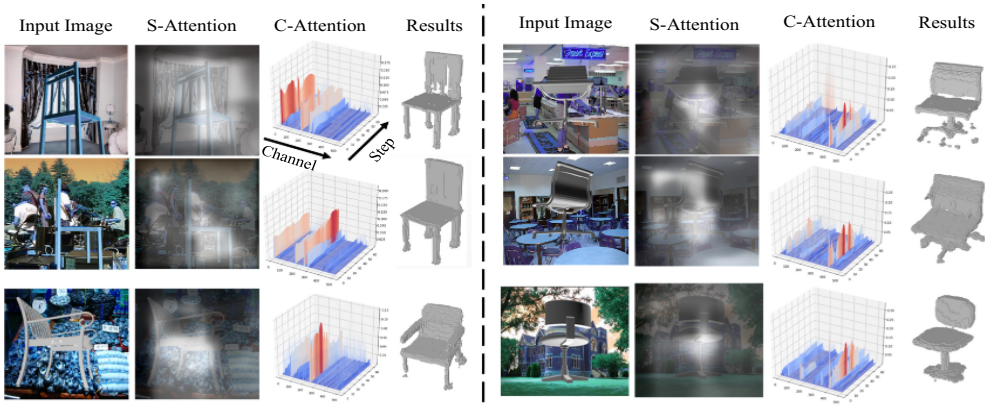

Fig. 6: Visualization of slice-aware attention result. From left to right: input, the spatial attention, the channel-wise attention and final reconstructed results. It shows our model can focus on the target shapes, ignoring background regions. The channel-wise attention is visualized with a 3D histogram. The value of channel-wise weight in each time step is represented in both color and height. Red color and high bar mean high attention weight.

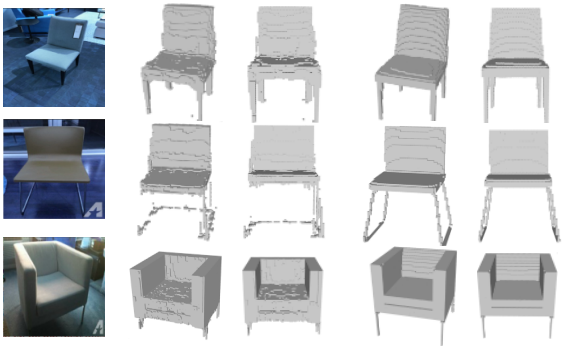

Fig. 7: Some reconstruction results of chairs on real image. From left to right: input, two views of SliceNet predictions and two views of ground truth.

Visualization of Attention. We visualize the learned attention to get a better understanding of it. From Fig.6, we can see that our model focus on the target objects in the image, ignoring background regions. The visualization of channel-wise attention shows that only a small part of channels are activated, implying the extracted image feature is redundant. What's more, it's worthy to notice that similar shapes with different views or background corresponds to similar activation distribution (e.g., the chairs in left column are all equipped with four legs, and all pay more attention to the front part of channels). It means the image encoder represented specific shapes with regular channels.

Results on real data. We also test our model in the real-image dataset. Considering the huge difference between synthesized and real image, after training the model, we fine-tune it with PASCAL 3D+ [37] and then test it in the Pix3D dataset [38]. Some results are shown in Fig.7. From the results we can conclude that our method is also applicable to real images.

4. CONCLUSION

In this paper, we present the SliceNet to perform slice-wise 3D reconstruction. The commonly used 3D deconvolution is replaced by reusable 2D deconvolution, sparing lots of memory space. The LSTM is also introduced to capture global relations between slices. We also design a slice-aware attention module to provide dynamic information for each slice's generation. The model is able to produce more plausible results with higher resolution. Experiments on both synthesized data and real data verify the effectiveness of our proposed method.

5. REFERENCES

[1] Christopher B Choy, Danfei Xu, JunYoung Gwak, Kevin Chen, and Silvio Savarese, "3d-r2n2: A unified approach for single and multi-view 3d object reconstruction," in *European conference on computer vision*. Springer, 2016, pp. 628–644.

[2] Haoqiang Fan, Hao Su, and Leonidas J Guibas, "A point set generation network for 3d object reconstruction from a single image," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 605–613.

[3] Chuhang Zou, Ersin Yumer, Jimei Yang, Duygu Ceylan, and Derek Hoiem, "3d-prnn: Generating shape primitives with recurrent neural networks," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 900–909.

[4] Thibault Groueix, Matthew Fisher, Vladimir G Kim, Bryan C Russell, and Mathieu Aubry, "A papier-mâché approach to learning 3d surface generation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 216–224.

[5] Maxim Tatarchenko, Alexey Dosovitskiy, and Thomas Brox, "Octree generating networks: Efficient convolutional architectures for high-resolution 3d outputs," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 2088–2096.

[6] Jiajun Wu, Yifan Wang, Tianfan Xue, Xingyuan Sun, Bill Freeman, and Josh Tenenbaum, "Marrnet: 3d shape reconstruction via 2.5 d sketches," in *Advances in neural information processing systems*, 2017, pp. 540–550.

[7] Jiajun Wu, Chengkai Zhang, Xiuming Zhang, Zhoutong Zhang, William T Freeman, and Joshua B Tenenbaum, "Learning shape priors for single-view 3d completion and reconstruction," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 646–662.

[8] Xinchen Yan, Jimei Yang, Ersin Yumer, Yijie Guo, and Honglak Lee, "Perspective transformer nets: Learning single-view 3d object reconstruction without 3d supervision," in *Advances in Neural Information Processing Systems*, 2016, pp. 1696–1704.

[9] Christian Häne, Shubham Tulsiani, and Jitendra Malik, "Hierarchical surface prediction for 3d object reconstruction," in *2017 International Conference on 3D Vision (3DV)*. IEEE, 2017, pp. 412–420.

[10] Jiang Wang, Yi Yang, Junhua Mao, Zhiheng Huang, Chang Huang, and Wei Xu, "Cnn-rnn: A unified framework for multi-label image classification," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2285–2294.

[11] Yin Fan, Xiangju Lu, Dian Li, and Yuanliu Liu, "Video-based emotion recognition using cnn-rnn and c3d hybrid networks," in *Proceedings of the 18th ACM International Conference on Multimodal Interaction*. ACM, 2016, pp. 445–450.

[12] Jin Wang, Liang-Chih Yu, K Robert Lai, and Xuejie Zhang, "Dimensional sentiment analysis using a regional cnn-lstm model," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 2016, pp. 225–230.

[13] Shiou Tian Hsu, Changsung Moon, Paul Jones, and Nagiza Samatova, "A hybrid cnn-rnn alignment model for phrase-aware sentence classification," in *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, 2017, pp. 443–449.

[14] Zhen Zuo, Bing Shuai, Gang Wang, Xiao Liu, Xingxing Wang, Bing Wang, and Yushi Chen, "Convolutional recurrent neural networks: Learning spatial dependencies for image representation," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2015, pp. 18–26.

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[16] Long Chen, Hanwang Zhang, Jun Xiao, Liqiang Nie, Jian Shao, Wei Liu, and Tat-Seng Chua, "Sca-cnn: Spatial and channel-wise attention in convolutional networks for image captioning," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 5659–5667.

[17] Alec Radford, Luke Metz, and Soumith Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," *arXiv preprint arXiv:1511.06434*, 2015.

[18] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al., "Shapenet: An information-rich 3d model repository," *arXiv preprint arXiv:1512.03012*, 2015.

[19] Jianxiong Xiao, James Hays, Krista A Ehinger, Aude Oliva, and Antonio Torralba, "Sun database: Large-scale scene recognition from abbey to zoo," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE, 2010, pp. 3485–3492.