3D Scene Reconstruction from RGB Images Using Dynamic Graph Convolution for Augmented Reality

Tzu-Hsuan Weng* Robin Fischer† Li-Chen Fu‡

Department of Computer Science and Information Engineering
National Taiwan University

Abstract

The 3D scene reconstruction task aims to reconstruct the object shape, object pose, and the 3D layout of the scene. In the field of augmented reality, this information is required for interactions with the surroundings. In this paper, we develop a holistic end-to-end scene reconstruction system using only RGB images. We further designed an architecture that can adapt to different types of objects through our graph convolution network during object surface generation. Moreover, a scene-merging strategy is proposed to alleviate the occlusion problem by merging different views continuously. This also allows our system to reconstruct the complete surroundings in a room.

Index Terms: 3D Scene Understanding—Dynamic Graph Convolution—Augmented Reality—;

1 Introduction

In the past few years, virtual reality (VR) and augmented reality (AR) have experienced a rise in popularity due to the spread of affordable devices. Especially for AR applications devices need to be able to detect the geometry of the environment, in order to dynamically add virtual objects to the scene. Originally the 3D scene reconstruction task to only considered individual parts part of the scene, like reconstructing the surface of individual objects in the scene, while not considering the relationships among objects and the layout of the whole environment. Thereby lacking important scene information. In this paper, we propose a holistic 3D scene reconstruction system that can bridge the gap between the real world and the virtual world.

2 Related Work

In previous works object reconstruction has been explored in detail. However it is not enough to only interact with a single object in the virtual space. Therefor the holistic scene reconstruction task is introduced. A remaining challenge in this task is to run it in real time. Pan *et al.* [3] solve this problem by utilizing the coarse-to-fine function to deform a template mesh to the final result. In each step, they proposed a topology modification to prune the face that is far from the ground truth by a binary classifier. Nie *et al.* [1] enhance [3] to perform the face pruning process depending on the local density of the ground truth points, rather than a predefined threshold. These methods do not need a lot of post-processing and are suitable for real-time applications.

3 Methodology

Our scene reconstruction system contains a single-scene reconstruction system and a multi-scene merging system. For applications in which users are moving around the same environment, maintaining scenes reconstructed from different views is non-trivial. We continuously merge any newly obtained reconstructed scenes into the previous one.

*e-mail:r08922144@ntu.edu.tw
†e-mail:d08922027@ntu.edu.tw
‡e-mail:lichen@ntu.edu.tw

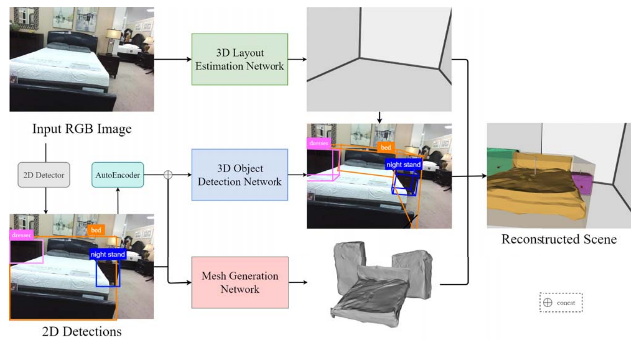


Figure 1: The overview of the Single Scene Reconstruction System.

3.1 Single-Scene Reconstruction System

Our single-scene reconstruction system comprises three modules: Layout Estimation module, 3D Object Detection module, Mesh Generation module. The overview of the system is shown in Figure 1. The input of the system is a single RGB image, which is is used to generate the 3D camera pose and 3D Layout bounding box, object bounding boxes and 2D offsets, reconstructed surfaces of the objects. Each of the individual module outputs is combined for the final output.

3.1.1 3D Object Detection Network

In the architecture of the detection network we adopt ResNet34 as our feature extractor. For the input RGB image of the scene, we use a pre-trained Auto-Decoder [2] to calculate the depth map of it, then concatenate the depth map to the RGB map.

3.1.2 Mesh Generation Network

In the mesh generation module, we design a Deformation network based on GCN taking advantage of the property that it can aggregate the information of all nodes on the graph. **Deformation network.** For most indoor objects, there is a standard pattern of what they should look like. Therefore we concatenate the initial template mesh only to the class code of the object as the input of GCN1 for it to learn the deformation between the two. As shown in Figure 2, the intermediate mesh of the 'bed' class is nearly an ellipse due to the characteristic of beds. Next, GCN2 is used to extract the feature for the final deformation. After processing the intermediate mesh, the local feature and global feature block are merged to generate the final feature of each node. The MLP blocks then predict the deformation of each node of the final object shape. **Chamfer distance loss.** The Chamfer distance is the main loss we used to guide our generation

2022 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW) | 978-1-6654-8402-2/22/$31.00 ©2022 IEEE | DOI: 10.1109/VRW55335.2022.00170

978-1-6654-8402-2/22/$31.00 ©2022 IEEE
DOI 10.1109/VRW55335.2022.00170

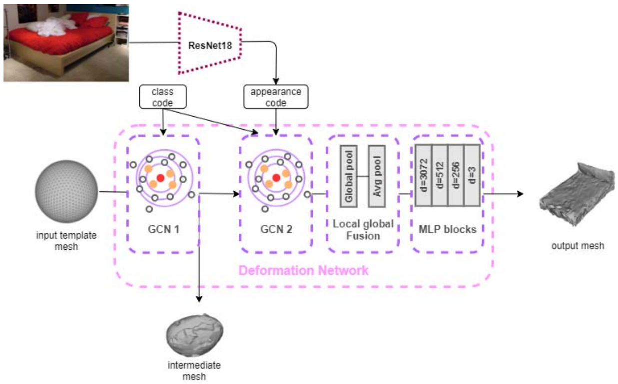
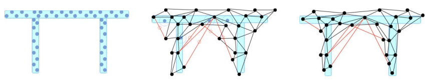


Figure 2: The architecture of the Mesh Generation Network.



(a) Ground truth (b) Before (c) After

Figure 3: Illustration of edge loss. (c) predicted mesh after applied edge loss.

module. It measures the nearest neighbor distance between the ground truth set $\mathscr{S}$ and generated point set $\mathscr{M}$. The CD loss is defined as follows:

$$\mathscr{L}_{cd} = \sum_{x\in\mathscr{M}} \min_{y\in\mathscr{S}} \|x-y\|_2^2 + \sum_{y\in\mathscr{S}} \min_{x\in\mathscr{M}} \|x-y\|_2^2 \tag{1}$$

where x and y are points belong to generated mesh $\mathscr{M}$ and the ground truth mesh $\mathscr{S}$, respectively.

Template loss. We replace the generated point set $\mathscr{M}$ with the intermediate point set $\mathscr{T}$ in CD loss to guide the intermediate mesh as well.

Edge loss. In addition, to make the points in the mesh more compact, we apply edge loss on edges E. The edge loss $\mathscr{L}_e$ is defined as follows:

$$\mathscr{L}_e = \frac{1}{|E_p|}\sum_{e\in E_p}|e|, \quad E_p = \{e \mid distance_{m_e} < density_{N(m_e)},\, e \in E_M\} \tag{2}$$

where $E_p \in E_M$ stands for plausible edges in the predicted mesh M. m_e is the middle point of edge e, $N(m_e)$ is the top-1 (nearest) neighbor on the ground truth S of m_e, $distance_{m_e}$ is the distance between m_e and $N(m_e)$, and $density_{N(m_e)}$ is the density of $N(m_e)$. The measurement of $density_{N(m_e)}$ follows the principle of Nie *et al.* [1]. The concept of edge loss is depicted in Figure 3.

3.2 Multi-Scene Merging Algorithm

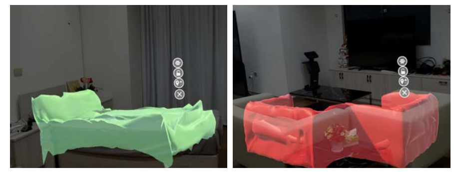
Our system maintains a single global scene, which is a set of objects with set parameters like rotation, size ect. We generate each local scene from the single-scene reconstruction system by the RGB image, which is packaged including it's metadata as inputs of the scene merging system. Each object is transformed it into the world coordinate system and aligned to the world position. Checking for overlaps we combine those objects from different views at different time stamps. We use a Intersection over Union (IoU) threshold of the bounding boxes to distinguish identical object.

4 Experiments

The SUN RGB-D dataset and Pix3D dataset are used to evaluate our system. For object reconstruction, the comparison is shows that our method outperforms previous state-of-the-arts by 20%. Especially in the misc class, our method shows the capability of being adaptive to various object shapes. Which is also shown by the variance of the loss of each class on the Pix3D dataset in Table 1. The low variance means our model does not simply fit the shape that appears in the dataset, but can be adaptive to each type of this set. Additional information about our setup, the ablation study, and quantitative results can be found in the appendix to this poster.

Category	bed	bookcase	chair	sofa	table	wardrobe	misc	mean
Total3D [1]	1.396	0.125	0.225	**0.047**	1.368	0.083	2.098	0.606
Ours	**0.708**	**0.024**	**0.189**	0.054	**1.197**	**0.058**	**0.351**	**0.303**

Table 1: Comparisons of object reconstruction on Pix3D dataset using the variance of the Chamfer distance. The values are in units of 10^3.



(a) Reconstructed bed (b) Reconstructed coaches

Figure 4: Result on Magic Leap One.

4.1 Real-World AR Application

We demonstrate our scene reconstruction system by implementing an augmented reality application for a real-world scenario.Figure 4 shows the result on Magic Leap One in combination with a personal computer running the server to process the video frame from the Magic Leap into the reconstructed scene.

5 Conclusion

We propose a holistic scene reconstruction system which incorporates a custom merging strategy to update the scene in real-time by merging the scenes from different timestamps. We design an architecture that can be adaptive to very different types of objects based on the property of our deformation network. The template loss can make learning progress easier, while the edge loss can punish the flying points and make the nodes more compact. In addition, the scene-merging strategy can alleviate the occlusion problem utilizing the different views from different timestamps. Given that the object reconstruction result is not smooth enough, balancing the visual quality and the object shape is what we consider for further improvements.

Acknowledgments

This research was supported by the Ministry of Science and Technology of Taiwan, and Center for Artificial Intelligence Advanced Robotics, National Taiwan University, under the grant numbers MOST 110-2634-F-002-049 MOST 110-2221-E-002-166-MY3.

References

[1] Y. Nie, X. Han, S. Guo, Y. Zheng, J. Chang, and J. J. Zhang. Total3dunderstanding: Joint layout, object pose and mesh reconstruction for indoor scenes from a single image. In *CVPR*, pp. 55–64, 2020.

[2] S. Niklaus, L. Mai, J. Yang, and F. Liu. 3d ken burns effect from a single image. *ACM TOG*, 38(6):1–15, 2019.

[3] J. Pan, X. Han, W. Chen, J. Tang, and K. Jia. Deep mesh reconstruction from single rgb images via topology modification networks. In *ICCV*, October 2019.