Weighted Voxel: a novel voxel representation for 3D reconstruction

Haozhe Xie
Harbin Institute of Technology
hzxie@hit.edu.cn

Hongxun Yao
Harbin Institute of Technology
h.yao@hit.edu.cn

Xiaoshuai Sun
Harbin Institute of Technology
xiaoshuaisun@hit.edu.cn

Shangchen Zhou
Harbin Institute of Technology
sczhou@hit.edu.cn

Xiaojun Tong
Harbin Institute of Technology
tong_xiaojun@hit.edu.cn

ABSTRACT

3D reconstruction has been attracting increasing attention in the past few years. With the surge of deep neural networks, the performance of 3D reconstruction has been improved significantly. However, the voxel reconstructed by extant approaches usually contains lots of noise and leads to heavy computation. In this paper, we define a new voxel representation, named *Weighted Voxel*. It provides more abundant information, facilitating the subsequent learning and generalization steps. Unlike regular voxel which consists of zero-one, the proposed Weighted Voxel makes full use of the structure information of voxels. Experimental results demonstrate that Weighted Voxel not only performs better in reconstruction but also takes less time in training.

CCS CONCEPTS

• **Computing methodologies** → **Reconstruction**; *Computer vision problems*;

KEYWORDS

3D Reconstruction, Voxel, Multi-view, Convolutional Neural Network, Long Short-Term Memory

ACM Reference Format:
Haozhe Xie, Hongxun Yao, Xiaoshuai Sun, Shangchen Zhou, and Xiaojun Tong. 2018. Weighted Voxel: a novel voxel representation for 3D reconstruction . In *The 10th International Conference on Internet Multimedia Computing and Service (ICIMCS'18), August 17–19, 2018, Nanjing, China.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3240876.3240888

1 INTRODUCTION

Reliably recovering 3D shape from one or more images of an object from arbitrary viewpoints has become feasible in computer vision over the last few years due to advances in deep learning [13]. Most of the state-of-the-art methods for 3D reconstruction, including structure from motion (SFM) [5] and simultaneous localization and mapping (SLAM) [6] are subject to a number of assumptions. For example, Both [9] and [7] assume that there should be matched features across images in different viewpoints. Several studies have proved that reconstruct from views which are separated by a large baseline [14] or lack of texture on objects [1] may be difficult and problematic. These assumptions are too strong to hold in many real applications.

In order to remedy the issue related to large baselines, methods with a different philosophy have been proposed in the past few years. With the help of deep neural networks, 3D reconstruction has been achieved tremendous success. Several methods [8, 16] proposed solved the daunting problem of reconstructing the shape of an object from a single view.

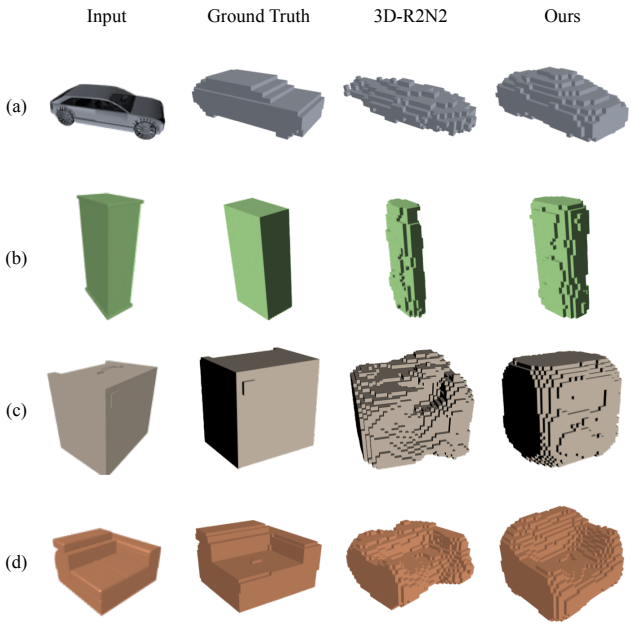

Figure 1: Reconstruction samples of (a) cars (b) cabinets (c) speakers (d) sofas on the ShapeNet testing dataset. Weighted Voxel preserves more structural details of 3D objects.

ICIMCS'18, August 17–19, 2018, Nanjing, China

ACM ISBN 978-1-4503-6520-8/18/08…$15.00
https://doi.org/10.1145/3240876.3240888

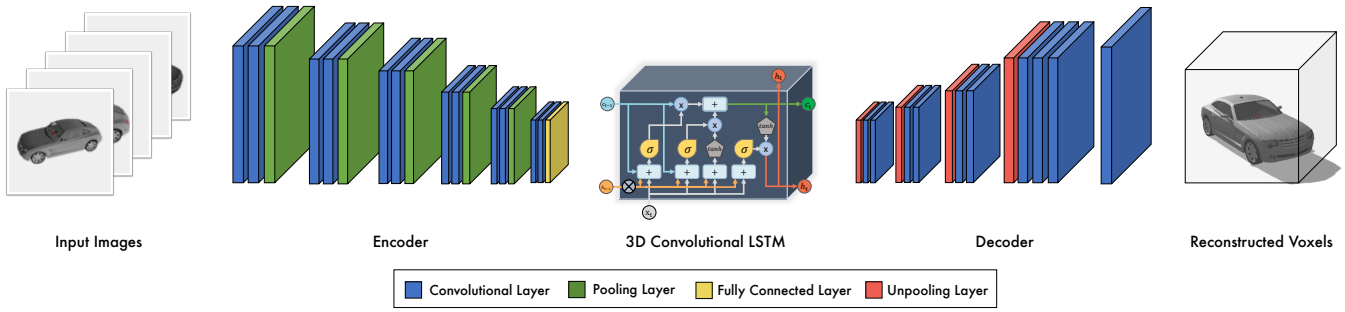

Figure 2: The network architecture for 3D reconstruction. Both of them consist of an encoder, a 3D convolutional LSTM, and a decoder. In 3D-R2N2, the reconstructed voxels are composed of zeros and ones, while in Weighted Voxel, the voxel values are filled with integers.

However, a single view cannot contain sufficient information of 3D shape and sometimes lead to high distortion [15]. 3D-R2N2 [4] takes fully use of supervision and recovers the approximated 3D shape from arbitrary viewpoints.

In this paper, we follow the same spirit as the 3D-R2N2, but with difference in voxel representation. Unlike regular voxel which is made up of zeros and ones, Weighted Voxel proposed in this paper takes structure information into account. In Weighted Voxel, the value of each voxel is weighted summed over its values of immediate neighbors. Experimental results demonstrate that the proposed Weighted Voxel outperforms 3D-R2N2 (see Figure 1).

The main contributions of this paper are summarized as follows:

- The proposed Weighted Voxel achieves higher voxel Intersection-over-Union (IoU) than 3D-R2N2 does.
- The convergence rate of our method is much higher than that of 3D-R2N2.

2 METHODS

Recurrent neural networks (RNNs) have shown great promise in many sequence learning tasks [17]. Long Short-Term Memory (LSTM) [11], as one of the most successful implementations of the hidden states of an RNN, is adopted by both baseline and our method to retain previous observations and incrementally refine the reconstruction of 3D objects as more observations are available. In the rest of this section, we provide a brief introduction of the baseline and our method.

2.1 Baseline: 3D-R2N2

As one of the state-of-art methods for 3D reconstruction, 3D-R2N2 [4] makes it possible to recover the shape of 3D objects from both single- and multi- view images. Besides, it is able to overcome past challenges of images with wide baseline viewpoints or insufficient texture.

The network of 3D-R2N2 is composed of three components: a 2D convolutional neural network (2D-CNN), a 3D convolutional LSTM (3D-LSTM), and a 3D deconvolutional neural network (3D-DCNN), which is illustrated in Figure 2.

Encoder. The input images are encoded as features by a 2D-CNN. The encoder consists of standard convolutional layers, pooling layers, and leaky rectified linear units followed by fully connected layers. Motivated by recent studies [10], residual connections are added between standard convolutional layers to speed up the optimization process. Finally, the encoder outputs a feature vector whose dimension is 1024.

3D Convolutional LSTM. As the core part of the network, the 3D-LSTM is made up of a set of structured Gated Recurrent Units (GRUs) [3] with restricted connections. There are $n \times n \times n$ 3D-LSTM units distributed in a 3D grid, where n is the spatial resolution of the 3D-LSTM grid. Each 3D-LSTM unit inside the 3D grid, indexed (i, j, k), has an independent hidden state $h_{t,(i,j,k)}$ that restricts to be affected by its neighboring 3D-LSTM units. In addition, convolution operations are applied between update gates and hidden gates.

Decoder. The decoder receives hidden state h_t from 3D-LSTM. The resolution is increased by applying 3D convolutions, non-linearities, and 3D unpooling until it reaches the target output resolution. As with the encoders, residual connections are followed by a final convolution.

2.2 Proposed method

The network structure of neural networks has evolved rapidly recent years. In general, deeper and wider networks leads to higher performance. However, it is hard to distinguish where the main benefit comes from. To clearly demonstrate the advantages of Weighted Voxel, we used a similar network as the 3D-R2N2 (see Figure 2). In the proposed method, we replace regular voxel with Weighted Voxel.

2.2.1 Voxel Structure. Unlike regular voxels whose values are zeros and ones, the values in Weighted Voxels are integers.

In regular voxels, each voxel is independent of other voxels, and the structure information is lost. Therefore, in the final reconstructed voxels, there are unexpected holes in a dense area or unexpected voxels in a sparse area. To tackle this problem, we employ a filter with size of $3 \times 3 \times 3$ that slides over regular voxels (see Figure 3). The value of each voxel in Weighted Voxel is weighted summed over voxel values of its

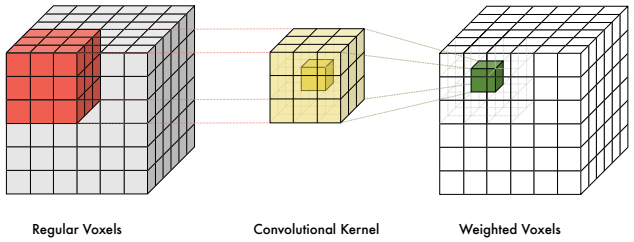

Figure 3: Generation of Weighted Voxels. The generation of Weighted Voxels can be regarded as applying a convolutional kernel on regular voxels.

immediate neighbors. More formally, the value in Weighted Voxel can be calculated as

$$y_{(i,j,k)} = -\omega(-1)^{v_{(i,j,k)}} - \sum_{m=i-1}^{i+1}\sum_{n=j-1}^{j+1}\sum_{p=k-1}^{k+1}(-1)^{v_{(m,n,p)}} \quad (1)$$

where $v_{(i,j,k)} \in \{0,1\}$ denotes the value in the regular voxel, and ω is set to 26. Specially, we define $v_{(i,j,k)} = 0$ when $i = -1$, $j = -1$ or $k = -1$.

The voxels with values of zeros in regular voxels are mapped onto negative values in Weighted Voxels. In contrast, positive values in Weighted Voxels are mapped from voxels whose values are ones in regular voxels. In addition, the larger values in Weighted Voxels correspond the denser area in the 3D object, while the smaller, the sparser. Compared to regular voxel, Weighted Voxel provides more abundant information that facilitates the subsequent reconstruction steps.

2.2.2 Loss: 3D Voxel-wise Mean Squared Error. The loss function used in the proposed method is defined as the mean of squared error. Assume the ground truth and predicted values are represented as $\mathbf{y}$ and $\widehat{\mathbf{y}}$, respectively. Therefore, the loss function can be expressed as

$$\ell(\mathbf{y}, \widehat{\mathbf{y}}) = \frac{1}{n}\sum_{i,j,k}(y_{(i,j,k)} - \widehat{y}_{(i,j,k)})^2 \quad (2)$$

3 EXPERIMENTS AND RESULTS

For envaluating the performance of Weighted Voxel, we conducted experiments on a desktop machine with an Intel Xeon E3 1230 v5 CPU (3.40 GHz) and a Nvidia GeForce 1080 Ti GPU (11 GB Memory). Our implementation uses the Theano framework and is available at https://github.com/hzxie/Weighted-Voxel.

3.1 Experimental Settings

In this section, we describe how the experiments were conducted. First, we introduce the dataset used in the experiments. Next, we give formal definition of the metrics used to evaluate the reconstruction results. Finally, we demonstrate the details of training neural networks.

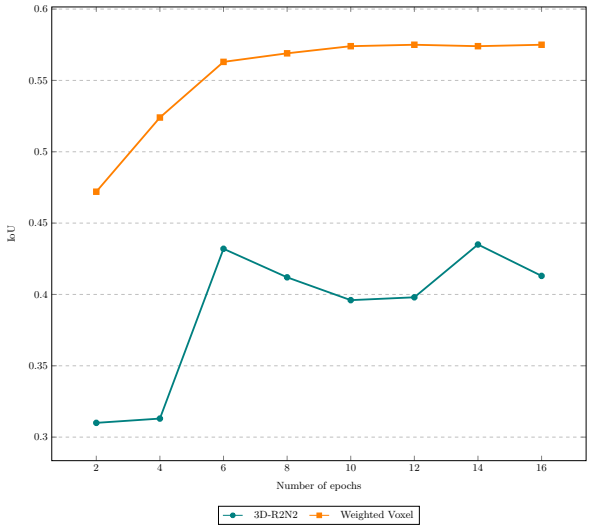

Figure 4: Comparison of reconstruction IoUs of 3D-R2N2 and our method. Our method outperforms the 3D-R2N2 in both convergence speed and IoU.

Dataset. We use a subset of ShapeNet [2] dataset for training and testing, which contains 43,783 CAD models from 13 major categories. The training and testing sets are generated from the dataset, in which 35,021 models are used for training, and 8,762 models for testing.

Metrics. We evaluated our performance of reconstruction quality by voxel IoU, which can be formulated as following

$$\text{IoU} = \frac{\sum_{i,j,k} I(pred_{(i,j,k)} > t) I(y_{(i,j,k)})}{\sum_{i,j,k} I(I(pred_{(i,j,k)} > t) + I(y_{(i,j,k)}))} \quad (3)$$

where $I(\cdot)$ is an indicator function, t represents a voxelization threshold, and $\mathbf{y}$ and **pred** are the ground truth and predicted voxel, respectively. Higher IoU values indicate better reconstructions.

Training. In training the neural networks, we used variable length of inputs ranging from one image to an arbitrary number of images. Specifically speaking, the number of views for each training sample in the same mini-batch is the same, but the input length varied randomly across different mini-batches. It makes the network can be applied to single- and multi- view reconstruction. During training, the loss is only be computed at the end of an input sequence in order to save computational resources.

Network. The size of input images is set to 127×127, while the output voxel is of size $32 \times 32 \times 32$. We optimize the networks by Adam [12] optimizer with a β_1 of 0.9, a β_2 of 0.999, a weight decay of 5×10^{-6}. The slope of the leak is set to 0.1 for LeakyReLU layers, and the initial learning rate is 10^{-5}. The optimization stops after about 10 epochs for our method and 40 epochs for the baseline. For fair comparison, we ran the experiments with the same configuration as 3D-R2N2 except that the voxel threshold t is set to 0.4 in 3D-R2N2 while set to 5×10^{-7} in Weighted Voxel.

Table 1: Per-category reconstruction of ShapeNet compared using voxel IoU.

Category	# views = 5		# views ∈ [1, 5]	
	3D-R2N2	Ours	3D-R2N2	Ours
Aero	**0.557**	0.522	**0.535**	0.510
Bench	0.378	**0.473**	0.332	**0.441**
Cabinet	0.743	**0.747**	0.709	**0.738**
Car	**0.833**	0.810	0.801	**0.802**
Chair	0.480	**0.511**	0.451	**0.491**
Monitor	0.501	**0.519**	0.349	**0.496**
Lamp	**0.373**	0.361	**0.362**	0.350
Speaker	**0.711**	0.703	**0.700**	0.688
Rifle	**0.561**	0.526	**0.515**	0.499
Sofa	0.664	**0.679**	0.613	**0.655**
Table	0.491	**0.530**	0.454	**0.511**
Phone	**0.686**	0.671	0.607	**0.648**
Watercraft	0.557	**0.589**	0.515	**0.559**

* Bold face indicates higher IoU on test data.

3.2 Results and Discussion

In this section, we report a quantitative evaluation of the proposed method in comparison with 3D-R2N2 on the ShapeNet testing set. We conducted the experiments on both fixed and variable length of inputs, and the experimental results are revealed in Table 1, Figure 1 and Figure 4.

Overall results. We first investigate the quality of reconstructed voxels under neural networks tested with 5 random views. Table 1 shows that our method outperforms the baseline method on 7 of 13 major categories. For neural networks tested with variable length of input images ranging from 1 to 5, Weighted Voxel makes a remarkable improvement in IoU compared to 3D-R2N2, where Weighted Voxel outperforms 3D-R2N2 on 9 of 13 categories. Figure 4 illustrates the trend of voxel IoU as the training progresses. The overall reconstruction quality improves as the number of epochs increases. Besides, not only does our method have much higher IoU, but also spends less time in convergence compared to 3D-R2N2. The IoU of Weighted Voxel remains unchanged after 10 epochs, however, the IoU still varies after 16 epochs in 3D-R2N2.

Per-category results. We also report the reconstruction IoUs on each of the 13 categories in the testing set in Table 1. We observed that the reconstruction quality of cabinet and watercraft is higher than that of 3D-R2N2. The shape of cabinets and watercraft is quite simple that can be well captured by Weighted Voxel. As shown in Figure 1, our recovered 3D objects contains less holes than those generated by 3D-R2N2. Weighted Voxel performs worse in reconstructing aeros, speakers and rifles. The objects in these categories have high texture level and the silhouette of the 3D object may be weaken by the filter of Weighted Voxel. In addition, both baseline and our method have poor reconstruction IoU in bench, lamp, and table categories. Compared with other classes, objects in these classes have more shape variation.

4 CONCLUSION

In this paper, we proposed a novel voxel representation named Weighted Voxel, where the value of each voxel is weighted summed over values of its immediate neighbors. To compare the reconstruction quality of 3D shape from variable or fixed length of images, we conducted experiments on the ShapeNet dataset. Experimental results demonstrate that Weighted Voxel does better in preserving the shape of 3D objects than 3D-R2N2, especially in recovering from variable length of inputs. Besides, the MSE loss function takes less time in convergence, which accelerates the training process.

ACKNOWLEDGMENTS

This work was supported by the National Natural Science Foundation of China under Project No. 61472103, 61772158, 61702136 and U1711265. We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan X Pascal GPU used for this research.

REFERENCES

[1] Dinkar N. Bhat and Shree K. Nayar. 1998. Ordinal Measures for Image Correspondence. *TPAMI* 20, 4 (1998), 415–423.
[2] Angel X. Chang, Thomas A. Funkhouser, Leonidas J. Guibas, Pat Hanrahan, Qi-Xing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. 2015. ShapeNet: An Information-Rich 3D Model Repository. *arXiv* 1512.03012 (2015).
[3] Kyunghyun Cho, Bart van Merrienboer, Çaglar Gülçehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. In *EMNLP 2014*.
[4] Christopher Bongsoo Choy, Danfei Xu, JunYoung Gwak, Kevin Chen, and Silvio Savarese. 2016. 3D-R2N2: A Unified Approach for Single and Multi-view 3D Object Reconstruction. In *ECCV 2016*.
[5] Sabine Demey, Andrew Zisserman, and Paul A. Beardsley. 1992. Affine and Projective Structure from Motion. In *BMVC 1992*.
[6] Hugh F. Durrant-Whyte and Tim Bailey. 2006. Simultaneous localization and mapping: part I. *IEEE Robot. Automat. Mag.* 13, 2 (2006), 99–110.
[7] Jakob Engel, Thomas Schöps, and Daniel Cremers. 2014. LSD-SLAM: Large-Scale Direct Monocular SLAM. In *ECCV 2014*.
[8] Haoqiang Fan, Hao Su, and Leonidas J. Guibas. 2017. A Point Set Generation Network for 3D Object Reconstruction from a Single Image. In *CVPR 2017*.
[9] Andrew W. Fitzgibbon and Andrew Zisserman. 1998. Automatic 3D model acquisition and generation of new images from video sequences. In *EUSIPCO 1998*.
[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep Residual Learning for Image Recognition. In *CVPR 2016*.
[11] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Computation* 9, 8 (1997), 1735–1780.
[12] Diederik P. Kingma and Jimmy Ba. 2014. Adam: A Method for Stochastic Optimization. *arXiv* 1412.6980 (2014).
[13] Yann LeCun, Yoshua Bengio, and Geoffrey E. Hinton. 2015. Deep learning. *Nature* 521, 7553 (2015), 436–444.
[14] David G. Lowe. 2004. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision* 60, 2 (2004), 91–110.
[15] Martin R. Oswald, Eno Töppe, Claudia Nieuwenhuis, and Daniel Cremers. 2013. A Review of Geometry Recovery from a Single Image Focusing on Curved Object Reconstruction. In *Innovations for Shape Analysis, Models and Algorithms*. 343–378.
[16] Jiajun Wu, Yifan Wang, Tianfan Xue, Xingyuan Sun, Bill Freeman, and Josh Tenenbaum. 2017. MarrNet: 3D Shape Reconstruction via 2.5D Sketches. In *NIPS 2017*.
[17] Chaoyun Zhang, Mingjun Zhong, Zongzuo Wang, Nigel H. Goddard, and Charles A. Sutton. 2018. Sequence-to-Point Learning With Neural Networks for Non-Intrusive Load Monitoring. In *AAAI 2018*.