Neural Processing Letters (2021) 53:653–670
https://doi.org/10.1007/s11063-020-10399-1

Object Reconstruction Based on Attentive Recurrent Network from Single and Multiple Images

Zishu Gao[1,2] · En Li[2] · Zhe Wang[1,2] · Guodong Yang[2] · Jiwu Lu[3] · Bo Ouyang[3] · Dawei Xu[1] · Zize Liang[2]

Accepted: 30 November 2020 / Published online: 5 January 2021

Abstract

The application of traditional 3D reconstruction methods such as structure-from-motion and simultaneous localization and mapping are typically limited by illumination conditions, surface textures, and wide baseline viewpoints in the field of robotics. To solve this problem, many researchers have applied learning-based methods with convolutional neural network architectures. However, simply utilizing convolutional neural networks without taking other measures into account is computationally intensive, and the results are not satisfying. In this study, to obtain the most informative images for reconstruction, we introduce a residual block to a 2D encoder for improved feature extraction, and propose an attentive latent unit that makes it possible to select the most informative image being fed into the network rather than choosing one at random. The recurrent visual attentive network is injected into the auto-encoder network using reinforcement learning. The recurrent visual attentive network pays more attention to useful images, and the agent will quickly predict the 3D volume. This model is evaluated based on both single- and multi-view reconstructions. The experiment results show that the recurrent visual attentive network increases prediction performance in a way that is superior to other alternative methods, and our model has desirable capacity for generalization.

Keywords Object reconstruction · Convolutional LSTM · Visual attention · Robotic application

✉ En Li
en.li@ia.ac.cn

Zishu Gao
gaozishu2016@ia.ac.cn

Jiwu Lu
jiwu_lu@hnu.edu.cn

[1] University of Chinese Academy of Sciences, Beijing 100049, China

[2] The State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China

[3] College of Electrical and Information Engineering, Hunan University, Changsha 410082, China

1 Introduction

Recently, understanding the 3D world has been demanded of various robotic applications, such as navigation [1] and object manipulation. Many object reconstruction methods have been proposed using RGB or depth views [2]. Because the depth of images is not sufficient to explain the original input, it can only be used as additional information for reconstruction. Therefore, automatic and effective 3D object reconstruction is more inclined to leverage RGB images, which are much easier to acquire.

Traditional 3D object reconstruction methods such as structured light, structure-from-motion (SfM) [3], and simultaneous localization and mapping (SLAM) [4,5] use the correspondence of geometric features to complete scene reconstruction effectively. However, they only work effectively under certain assumptions. Their limitations include the following: 1) images are expected to have rich texture and high brightness for successful feature extraction, 2) one perfect view or a number of dense views with small baselines are required, 3) object with specular reflection error, and 4) cumulative error.

Many generated methods are inspired by prior knowledge [6,7], overcoming the drawbacks of feature extraction and correspondences failure caused by a lack of texture. Shape prior-based methods can repair and make up for missing parts for 3D reconstruction [8–10], while semantic prior-based methods learn the prior category-level, which is used with weighted warping and refinement mechanisms to complete high-quality 3D shape representations [11]. These approaches lead to reconstruction using fewer images, which have fewer restrictions in illumination condition and surface texture. However, they operate under the strong premise that the input contains a large amount of information, which is not the case in many applications.

With the growing development of learning methods in many applications [12–16], many studies have been devoted to achieving 3D representation through learning-based methods. Inspired by this philosophy, this work leverages an end-to-end deep recurrent convolutional network based on auto-encoder architecture to learn 2D-to-3D mapping and recover the shape of object categories from one or multiple images.

To improve efficiency in completing high-precision reconstruction, rather than taking images randomly, we utilize a convolutional Long Short Term Memory (LSTM) encoder to obtain an attention latent unit with attentive information and to implement a recurrent attentional model for actively selecting those images that can be helpful. The main contributions of this study are summarized as follows: (1) The recurrent visual attentional model and selection policy are established to achieve more accurate results of representation with fewer input images. (2) A 2D convolutional LSTM encoder is utilized to gain an attentive latent unit with beneficial properties, such as providing attractive information and smoothness. (3) Extensive experiments show that this method is superior to state-of-the-art learning-based methods.

This paper is structured as follows. Section 2 discusses related research on 3D reconstruction based on learning methods and recurrent attention models. The pipeline of our object reconstruction network architecture and the detail of the proposed method are explained in Section 3. Section 4 presents the experimental results and discussions while Sect. 5 states the conclusions.

2 Related Work

2.1 3D Reconstruction Based on Learning Methods

Researchers focus on learning-based methods to predict single- and multi-view 3D shapes [17,18]. For single-view shapes, [19] introduces the deep belief network to learn the probability of the voxel, and the author uses the image with 2.5D information to fill in the unknown voxel to complete the reconstruction. Several pioneers have utilized the auto-encoder convolutional network to predict 3D shapes when only one image of the object is given [20–23]. [24] generates point cloud coordinates rather than volumetric grids in a fully-supervised manner, which avoids obscuring the natural invariance of 3D shapes under geometric transformations. [25] proposes a 3D interpreter network to predict 3D object structure and 2D keypoint heat maps sequentially. It is worth mentioning that this work simultaneously completed 2D keypoint estimation and 3D structure and viewpoint recovery. Related to our work, [26] introduces perspective transformations leading to inferring 3D shapes without using the ground truth 3D volumetric data for training, which proposes a novel perspective transformer model for 3D reconstruction.

Single-view-based methods suffer from self-occlusion and a lack of enough information from other viewpoints; hence, many researchers have increasingly focused on multi-view reconstruction. [27] uses a projective generative adversarial network where projections of 3D models match the distributions of the input images, which assisted in shape predictions. [28] presents a 3D Recurrent Neural Network (3D-R2N2) with a LSTM unit to make a prediction using one or random multi-view images, which is a state-of-the-art method for 3D reconstruction.

2.2 Recurrent Attention Model

Classic auto-encoder networks have benefitted from the attention mechanism [29]. [30] proposes a recurrent network and novel attention model to extract features from images by adaptively selecting a sequence of locations and regions. [31] addresses an improved model and trains an attention-based deep recurrent neural network (RNN) with reinforcement learning to find the most relevant regions of the input image for multiple object recognition. [32] attempts to train an attention-based model in a deterministic manner to generate image captions automatically. [23] proposes a method of target tracking combining soft attention and an LSTM loop structure.

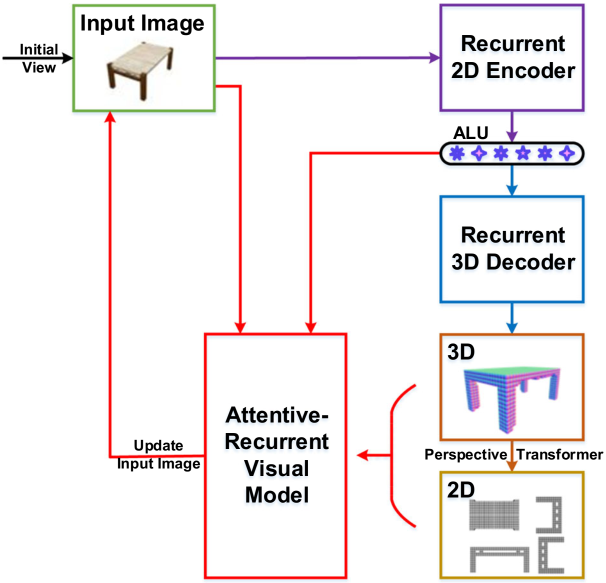
Inspired by these applications, our work develops auto-encoder networks combined with a recurrent visual attentional model to build 3D volume to minimize reconstruction error. The recurrent attention network is applied to induce the network to focus on more helpful views, which allows reconstruction to use as few images as possible while maintaining quality.

3 Recurrent Convolutional Auto-Encoder Network

In this section, we introduce the overall description of our proposed network. There are three main components to our network: a 2D convolutional LSTM encoder, a 3D convolutional LSTM decoder, and an attentive-recurrent visual model, as shown in Fig. 1.

Given one or multiple images of an object, our objective is to reconstruct its 3D model. In the training stage, beginning with an arbitrary view, we single out the rendered RGB images

Fig. 1 Object reconstruction model. Starting with an arbitrary view, the rendered RGB image corresponding to this view is feed into the 2D convolutional LSTM encoder to obtain an attentive latent unit and then decode the attentive latent unit to attain the 3D volume by utilizing a 3D convolutional LSTM decoder. At the same time, the attentive-recurrent visual model continuously regress the next informative image parameterized as camera azimuth. The new parameterized image is then fed into the 2D convolutional LSTM encoder to begin the next iteration



corresponding to this view and feed them into a 2D convolutional LSTM encoder. The main goal of the 2D convolutional LSTM network is to encode the input images into attentive latent units, which play an important role in spreading attentive information when new images are fed into encoder. Then the network is split into two branches, and the attentive latent units act as inputs for both branches. One branch is the 3D convolutional LSTM decoder, which decodes the attentive latent unit and generates a 3D voxel prediction. The other branch is the attentive-recurrent visual model, which takes the hidden states and converts them into new parameterized images which contribute to the high-quality reconstruction in the next training iteration. This allows the network to focus on views with rich information. The new parameterized images are then fed into the 2D convolutional LSTM encoder to begin the next training iteration. In the testing stage, when targeting an object, our network generates the view of the object under the control of the attentive-recurrent visual model and predicts the 3D volume by utilizing the 2D convolutional LSTM encoder and 3D convolutional LSTM decoder. We call this process the 2D-3D-attention network for convenience.

3.1 2D Convolutional LSTM Encoder

Figures. 2 and 3 illustrate the detailed network architecture, 2D convolutional LSTM encoder, 3D convolutional LSTM encoder, and attentive-recurrent visual model.

The 2D convolutional LSTM encoder is leveraged to extract compressed features. It is composed of three layers of residual blocks [33], which help capture features from inputs followed by the convolutional LSTM. The convolutional LSTM replaces the fully connected operation in the traditional LSTM with a convolution operation, which considers the spatial and temporal connections and improves feature extraction. Therefore, the convolutional LSTM decreases the loss of detailed information and captures long-term dependencies in

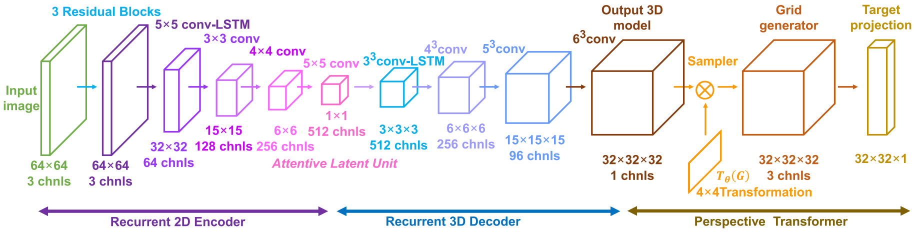


Fig. 2 Recurrent convolutional auto-encoder network architecture. Our network is composed of a 2D convolutional LSTM encoder and a 3D convolutional LSTM decoder combined with a perspective transformer. Taking an image as the input, the 2D convolutional LSTM encoder generates an attentive latent unit, and then the 3D convolutional LSTM decoder predicts the 3D shapes, using a perspective transformer map of the 3D shapes to screen coordinates and project 2D silhouettes for back propagation

sequences of images. The convolutional LSTM formulas are as shown below:

$$\begin{cases} f_t = \sigma(W_{xf} * X_t + W_{hf} * H_{t-1} + W_{cf} * C_{t-1} + b_f) \\ \mathrm{i}_t = \sigma(W_{xi} * X_t + W_{hi} * H_{t-1} + W_{ci} * C_{t-1} + b_i) \\ C_t = f_t \circ C_{t-1} + \mathrm{i}_t \circ \tanh(W_{xc} * X_t + W_{hc} * H_{t-1} + b_c) \\ o_t = \sigma(W_{xo} * X_t + W_{ho} * H_{t-1} + W_{co} * C_t + b_o) \\ H_t = o_t \circ \tanh(C_t) \end{cases}$$

where f_t, i_t, and o_t refer to the forget gate, input gate, and output gate, respectively. C_t presents the memory cell, which is fed into the next LSTM, and H_t refers to the hidden state, where output features are generated. We use $*$ and $\circ$ to represent convolutional and element-wise multiplication operators, respectively, and X_t refers to the input features generated by Residual blocks. $W_{(x\cdot)}$, $W_{(h\cdot)}$, and $W_{(c\cdot)}$ are weights that transform the current input X_t, the previous hidden state H_{t-1}, and the memory cell, respectively, and $b(\cdot)$refers to the biases. Tanh function is a saturated activation, which means that when the input reaches a certain value, the output will not change significantly. If we use an unsaturated activation function, such as ReLU, it will be difficult to achieve the effect of gating.

Next, three convolutional layers generate an attentive latent unit by taking the previous convolutional LSTM output features as their inputs, giving us A_t.

$$A_t = f_{convLSTM}(I^t_{image}, H^{t-1}_{convLSTM})$$

where I^t_{image} refers to the input views, and $H^{t-1}_{convLSTM}$ are all past states of hidden layers at $t-1$ time step.

In the next stage, the attentive latent unit A_t will be fed into two parts: the decoder network and attentive-recurrent visual model.

3.2 3D Convolutional LSTM Decoder

As shown in Fig. 2, the 3D convolutional LSTM decoder consists of two subnetworks: a convolutional LSTM model and a perspective transformer network. The first part is composed of one convolutional LSTM and three convolutional layers to establish the map from the attentive latent unit for 3D reconstruction. The convolutional LSTM considers the spatial connection and guarantees the validity of the 3D reconstruction. The other part is the perspective transformer network, which includes a perspective grid generator and a bilinear sampler perspective. The perspective grid generator normalizes the voxel grid by transforming the

3D world coordinates into screen coordinates.

$$P_i^s \sim T_{4\times 4} P_i^w$$

where P_i^s are screen coordinates in volume V, and P_i^w are 3D world coordinates in volume W. $T_{4\times 4}$ is a 4 x 4 transformation matrix:

$$T_{4\times 4} = \begin{bmatrix} K & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} R & t \\ 0 & 1 \end{bmatrix}$$

where K refers to the internal parameter matrix, and R and t are external parameters.

Bilinear sampler perspective maps pixels are converted to 3D mesh coordinates using bilinear interpolation. This can be written as

$$\begin{aligned} &V_i = \sum_n^H \sum_m^W \sum_l^D W_{nml} \max(1 - \left| x_i^s - m \right|, 0) \\ &\max(1 - \left| y_i^s - n \right|, 0) \max(1 - \left| z_i^s - l \right|, 0) \\ &\forall i \in [1 \ldots H^{'} W^{'} D^{'}] \end{aligned}$$

where (H, W, D) and $(H^{'}, W^{'}, D^{'})$ are the height, width, and depth of input volume W and output volume V. (x_i^s, y_i^s, z_i^s) is the coordinate of input volume W, and $n, m,$ and l represent the nth, mth, and lth pixel, respectively. Finally, the projection is implemented by max operation to generate a 2D silhouette S. We utilize 3D volume V and 2D silhouette S to compute the loss. There are two loss functions in our networks: 3D volume loss and 2D silhouette loss. We define the loss function as follows:

$$L_{MSE}(m) = \left\| m - m_{groundtruth} \right\|^2$$

then the 3D volume loss can be written as $L_{MSE}(V)$, and 2D silhouette loss is shown as follows:

$$L_{MSE}(S) = \frac{1}{n} \sum_{i=1}^{n} L_{MSE}(S_i)$$

where n refers to the number of images. Overall, the losses of our recurrent convolutional auto-encoder network are shown as follows:

$$L_{MSE} = \lambda_V L_{MSE}(V) + \lambda_S L_{MSE}(S)$$

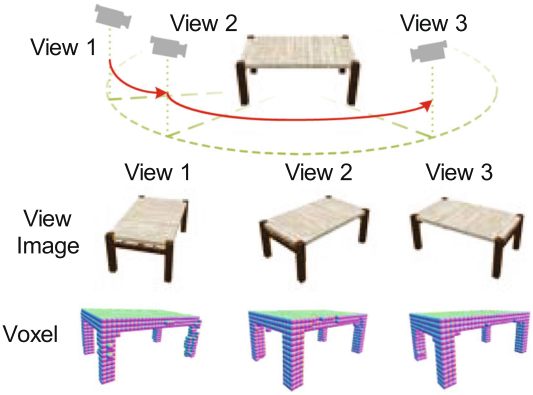
where λ_V and λ_S are the weights of the 3D volume and 2D silhouettes losses, respectively.

3.3 Attentive-Recurrent Visual Model

In the 3D shape prediction process, some images have less of an effect on reconstruction, while others are extremely helpful for reconstruction. If we can discover which images have an important impact on reconstruction, then we may be able to improve the accuracy of reconstruction significantly. In some computer vision applications, the attention mechanism is mainly used to find a piece of interest in a picture. However, we use the attentive-recurrent visual model to find an image of interest in a series of images at each step. The overview of the selection of the attentive-recurrent visual model is shown as Fig. 3.

This method can be considered the sequential decision process of an agent in a 3D visual environment where the agent is built around a recurrent model, as shown in Fig. 4.

Fig. 3 The selection of the attentive-recurrent visual model



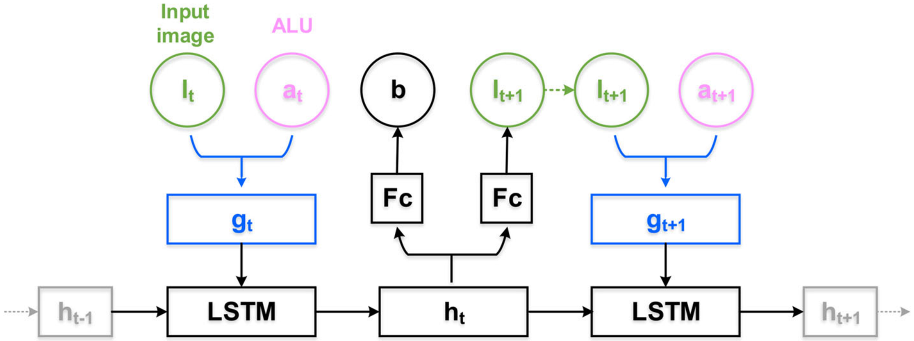


Fig. 4 The attentive-recurrent visual model, where the LSTM receives the parameterized input image and attentive latent unit from the 2D convolutional LSTM encoder. The output hidden state is fed into a fully connected convolutional layer to regress the parameter of the new image and baseline

The model regresses images which are parameterized as camera azimuth. At each step, the parameterized image l_t and attentive latent unit A_t constitute the glimpse feature vector g_t,

$$g_t = A_t * l_t$$

With an LSTM network, our model aggregates the glimpse feature vector g_t and the hidden states. The hidden state h_{t-1} is used to remember and store information from the past states, and it can be written as follows:

$$h_t = f_{LSTM}(g_t, h_{t-1})$$

Then the hidden state h_t is fed into the fully connected layer to predict the image parameters which the agent feeds into the 2D–3D-attention network and the attentive-recurrent visual model at next time step.

In this study, we calculate the cumulative reward for an entire episode to update the policy. The reward is measure by intersection-over-union (IoU). The IoU formula is as follows:

$$IoU(\mathrm{PR}) = \frac{\Pr edict\, \mathrm{Result} \cap Ground\, Truth}{\Pr edict\, \mathrm{Result} \cup Ground\, Truth}$$

The selection at each step in the attentive recurrent visual model is a regress process. The standard of the selection includes as follows: when the regressed view gets higher 3D IoU reward and 2D IoU reward, it can be seen that the regressed view is better. Besides, the regressed view has a low overlap rate with the previous view, it can be seen that the regressed view is informative. Therefore, the reward should be considered from three perspectives. First, we compare the 3D volume predicted by the 2D-3D-Attention network with the 3D ground truth, and obtain the first part of the reward:

$$r_{3D}^{t} = IoU(V_t) - IoU(V_{t-1})$$

where V_t refers to the 3D shapes predicted by the 2D-3D attention network. Next, the perspective transformer is used to compute the 2D mapping of the predicted 3D model in the decoder network. The 2D mapping can be used as the second part of the reward:

$$r_{\mathrm{2D}}^{t} = \frac{1}{n}\sum_{i=1}^{n} IoU(P_i^t) - IoU(P_i^{t-1})$$

where P_i^t is the 2D projection sampled by the perspective transformer model at t step.

Consequently, the higher the content overlap between the selected and previous images, the less information is available for the agent. Therefore, it is valuable to select an image with a relatively low overlap rate with the previous input image to calculate the distance between the current and previous images to measure the reward. The smaller the distance, the greater the penalty for the action performed by the agent. The distance can be formulated as follows:

$$r_{dis\tan ce}^{t} = \frac{1}{(current_{image} - past_{image})^2}$$

where $current_{image}$ is the value parameterized as the camera azimuth, and the value is set at $[0, 2\pi]$. Finally, the reward can be expressed as the sum of the above three parts as the formula:

$$r_t = r_{\mathrm{3D}}^{t} + r_{\mathrm{2D}}^{t} + r_{dis\tan ce}^{t}$$

In this research, we use the gradient policy algorithm used in the partially observable Markov decision processes (POMDPs) to control the policy. If an action makes the reward larger, then it should be associated with a higher probability in the strategy. In contrast, if the reward obtained by an action is small, then the probability of its occurrence in the policy should be lower. The strategy formula is as follows:

$$J(\theta) = \sum_{t=1}^{T} -\log \pi(u_t|\theta; s_t)(r_t - b_t)$$

where $\pi(u_t|\theta; s_t)$ is represented by a normal distribution function with the following formula:

$$\pi(u_t|\theta; s_t) = \frac{1}{\sqrt{2\pi}\sigma} e^{\frac{-(loc-\mu_{loc})^2}{2\sigma^2}}$$

where $(r_t - b_t)$ refers the evaluation index, and r_t is the cumulative reward for executing an action. To prevent a high variance, b_t is used as a baseline [23], which is obtained from the hidden layers in the LSTM network.

4 Experiments and Analysis

In this section, we carry out several experiments to demonstrate and validate the goals of the study. First, we describe the datasets and implementation of our methods (Sect. 4.1). Next, we demonstrate the improvement in reconstruction accuracy using the attentive-recurrent visual model (Sect. 4.2). Third, the hyper-parameter influence is tested and analyzed (Sect. 4.3). Then we compare the capacity of our network with state-of-the-art methods for single-view reconstruction (Sect. 4.4). In addition, we validate the ability of our approach to perform multi-view reconstruction predictions (Sect. 4.5). We also analyze the computational time for each step of the proposed network (Sect. 4.6). Finally, we conduct extended experiments to evaluate the generalization ability of our method (Sect. 4.7).

4.1 Dataset and Training Details

Experiments are conducted using two datasets: ShapeNet and PASCAL 3D+. The ShapeNet dataset is composed of 55 object categories with more than 51,300 3D models. We use 13 major classes as the 3D-R2N2 set for the single- and multi-view reconstruction comparison. It is composed of rendered images taken from 24 azimuth angles and representative models with sizes of 32 × 32 × 32 based on their canonical orientations. In this experiment, the rendered images are cropped and rescaled according to their centering region to 64 × 64 × 3 [26]. The PASCAL 3D+ dataset consists of 12 rigid categories of the PASCAL VOC 2012 [34] data with a 3D CAD model. For the convenience of training and testing, we used the rendered images in the ShapeNet dataset format [26].

We implement the system using the public Pytorch framework. We use a batch size of 16 to fit in an NVIDIA Titan X GPU. During training, we set Adam optimizer with $\beta_1 = 0.9, \beta_2 = 0.99$ and learning rate of 0.0001.

4.2 Network Structure Variance Comparison

To demonstrate that incorporating the attentive-recurrent visual model improves performance, we compared the 2D-3D network, with the 2D-3D-attention network, which combines the encoder-decoder network and the attentive-recurrent visual model. We trained both models on the couch category using different views in the experiments. Table 1 shows the quantitative results of loss and IoU, and Fig. 5 describes the trend of loss and IoU when taking in multiple images. We observe that the 2D-3D-attention network outperforms the 2D-3D network. For each network, reconstruction results improve as the number of views increases.

Table 1 Multi-view reconstruction using our variant model

	Loss					IoU				
view	1	2	3	4	5	1	2	3	4	5
2D–3D	0.138	0.129	0.104	0.090	0.088	0.635	0.639	0.642	0.678	0.688
2D–3D-attention	0.108	0.091	0.088	0.072	0.064	0.698	0.699	0.704	0.720	0.724

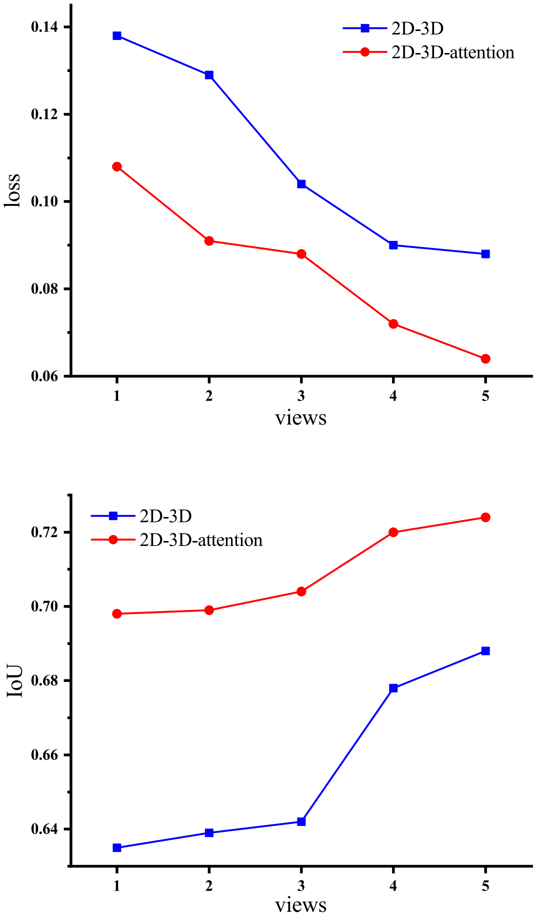


Fig. 5 Reconstruction performance of 2D–3D network variations. The loss and IoU for the couch category

4.3 Hyper-Parameter Influence

To compare the influence of hyper-parameters λ_V and λ_S of the loss function, the network was trained and tested using the motorbike classes of the PASCAL 3D+ dataset. In this experiment, we set up several sets of parameters and compared the IoUs of training and testing under different hyper-parameter combinations. When λ_V is equal to 0 or 1, we only use 2D silhouette loss or 3D volume loss, respectively.

The results of this experiment are presented in Table 2. The network performs better when 2D silhouette loss accounts for a larger weight. We can conclude that the effect of 2D silhouette loss is greater than that of 3D volume loss. The proposed network performs best when λ_V and λ_S are 0.4 and 0.6, respectively. Therefore, this hyper-parameter set is introduced in all other experiments.

4.4 Single-View Reconstruction

To validate the performance of our 2D-3D-attention network, we compared the results with several different reconstruction networks including the 3D-R2N2, Perspective Transformer

Table 2 IoU predictions using different hyper-parameter combinations

λ_V	λ_S	Mean IoU	
		Training	Test
0	1	0.592	0.523
0.2	0.8	0.604	0.530
0.4	0.6	0.606	0.533
0.5	0.5	0.604	0.529
0.6	0.4	0.546	0.485
0.8	0.2	0.505	0.415
1	0	0.488	0.462

Table 3 Per-category average IoU using single view

	3D-R2N2	OGN	PTN	Ours
Plane	0.513	0.587	0.553	**0.602**
Bench	0.421	0.481	0.482	**0.508**
Cabinet	0.716	0.729	0.711	**0.757**
Chair	0.466	**0.483**	0.458	0.468
Car	0.798	**0.816**	0.712	0.788
Monitor	0.468	0.502	0.535	**0.566**
Lamp	0.381	0.398	0.354	**0.398**
Speaker	0.662	0.637	0.586	**0.700**
Firearm	0.544	0.593	0.582	**0.598**
Couch	0.628	0.646	0.643	**0.698**
Table	0.513	0.536	0.471	**0.624**
Cellphone	0.661	0.702	0.728	**0.758**
Watercraft	0.513	**0.632**	0.536	0.532

Nets (PTN), and Octree Generating Network (OGN) [35]. The 3D-R2N2 takes in one or more images and predicts the voxel using a 3D RNN. The PTN uses proposed perspective transformations to reconstruct the volumetric structure from a single view. The OGN explores octree representation to generate high-resolution 3D outputs. Because the 3D reconstruction networks that PTN and OGN designed are suitable for inputting an image, we carried out single-view reconstruction experiments with them on the ShapNet datasets. To ensure a fair comparison, we trained and tested the proposed 2D-3D attention network on the same datasets and settings as with the 3D-R2N2 and OGN, which is split by Choy et al. In addition, we retrained the released the PTN model using these categories because they only conducted experiments using the chair category. The results are presented in Table 3.

Table 3 shows that performance varied depending on the category in the single-view tests. Our model has the best performance for most categories due to the presence of the attentive-recurrent visual model. The OGN outperforms our model in the chair, car, and watercraft categories because they have greater differences in shape variance between different types than other classes. For example, a chair can be an armchair or a rocking chair, which have different frameworks, and it is difficult to reconstruct 3D outputs from a random single view due to self-occlusion. However, our proposed network makes up for this deficiency when producing multiple-view reconstructions. The trends of mean reconstruction IoU of

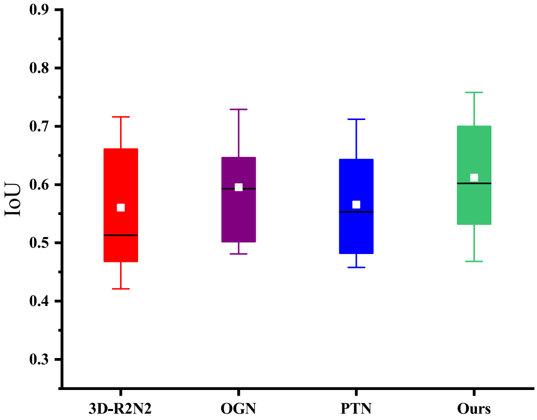


Fig. 6 The performance is reported in median(black line) and mean (white square) intersection over union (IoU) values

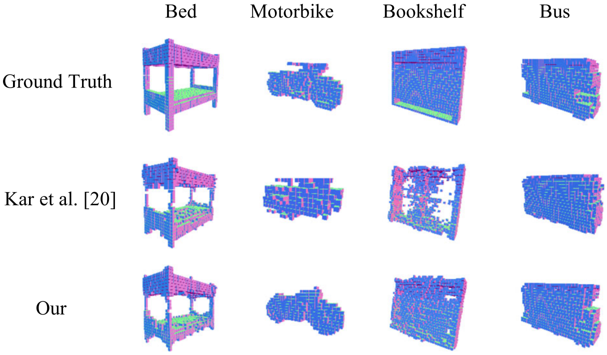


Fig. 7 3D prediction examples on PASCAL 3D+ dataset

the compared networks were plotted in Fig. 6. The results indicate that the proposed network structure is superior to that of the other methods overall.

In addition, to further evaluate the performance of the proposed network, the single-view reconstruction results on PASCAL 3D+ dataset are tested comparing with the method by Kar et al. [20]. This method predicts the 3D surfaces learning from single image using ground truth segmentations and keypoints. A network trained on the ShapeNet dataset was fine-tuned for training and testing on the PASCAL 3D+ dataset. Fig. 7 shows the visible reconstruction results. It can be seen that our 3D prediction results are better than the results of the Kar et al. in each class. Our network can predicts more details such as the legs of bed.

4.5 Multi-View Reconstruction

In this section, we compared our 2D-3D-attention network to the 3D-R2N2 in multi-view reconstruction. The experiments used five views to make a prediction. Based on the visualization results shown in Fig. 8, our method is more advantageous than the 3D-R2N2 approach in reconstructing thin parts of an object, such as the legs of tables and speakers. This shows that our 2D-3D-attention model can find images more effectively and boost the performance under certain views.

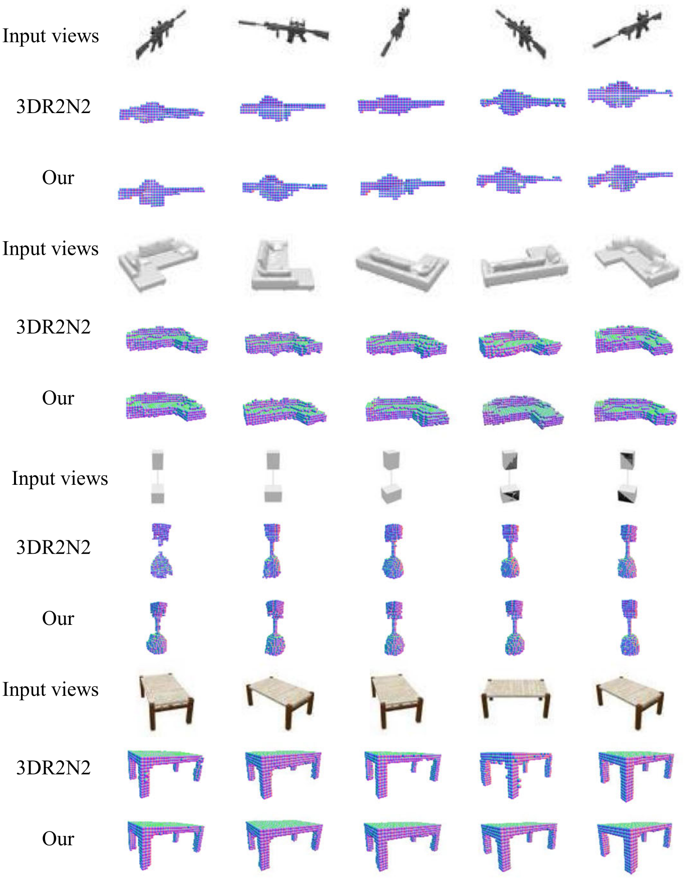


Fig. 8 Multi-view reconstruction comparison

4.6 Shannon Entropy Evaluation

To evaluate the performance of view prediction, we compared our method with random selection. The random selection means the module selects random view as the next view. We use the table category to do this experiment and we employ Shannon Entropy as the measure. Shannon Entropy formula is as follows:

$$\mathrm{H}(P_t^{pre}) = -\sum_{i=1}^{32}\sum_{j=1}^{32}\sum_{k=1}^{32} P_t^{pre}(i,j,k)log(P_t^{pre}(i,j,k))$$

where $P_t^{pre}(i,j,k)$ refers to the predicted value of each voxel. When a new view is taken into the model, the more the Shannon entropy decreases, the more information the newly added image has. The Fig. 9 shows the Shannon Entropy over the number of views. It can be seen that our method reduce Shannon entropy more than random selection, and it means predicted view is informative than random view.

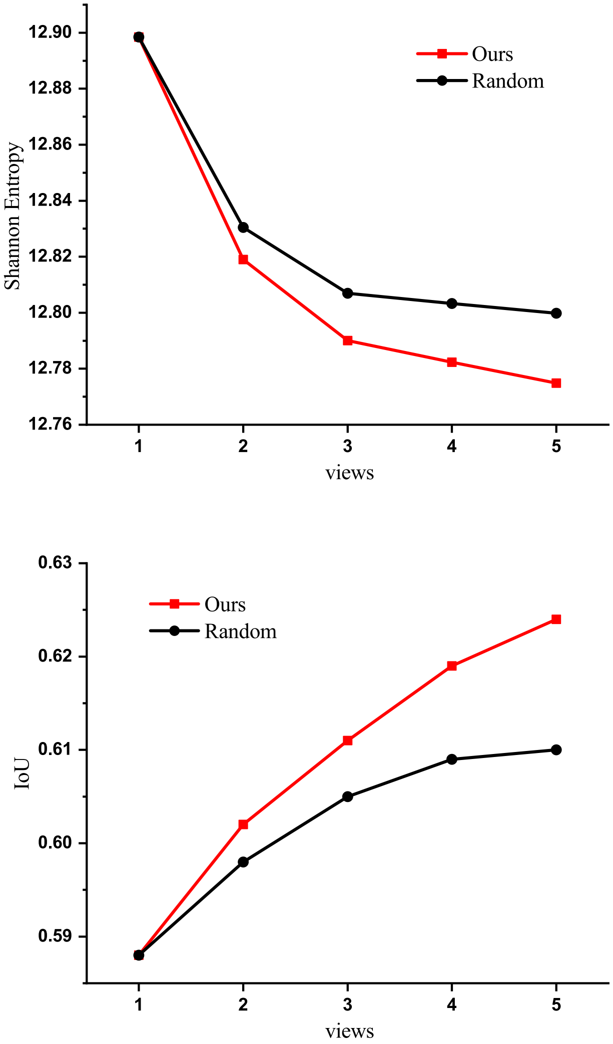


Fig. 9 The Shannon Entropy and IoU for the table category

4.7 Evaluation on computational time

In this experiment, we validated the computational efficiency when the network takes in different views. The proposed network ran on the Ubuntu16.04 platform equipped with a 3.6GHz Intel i7-7700 CPU, 32GB system memory, and NVIDIA Titan X GPU with 11 GB frame buffer memory. The reconstruction process can be divided into four steps: pre-processing data, loading models, making prediction, and preserving outputs. We recorded the average computational time for each step. Table 4 shows the average processing time for single-view reconstruction. Loading models costs the most time for the network followed by making prediction, pre-processing data, and preserving outputs. There is still room for improvement regarding the computational efficiency.

Table 4 Average processing time in each step for single-view reconstruction

Step	Time (s)	Percentage (%)
Pre-processing data	0.455	11.34
Loading models	2.774	69.13
Making prediction	0.516	12.85
Preserving outputs	0.268	6.68
Total	4.013	

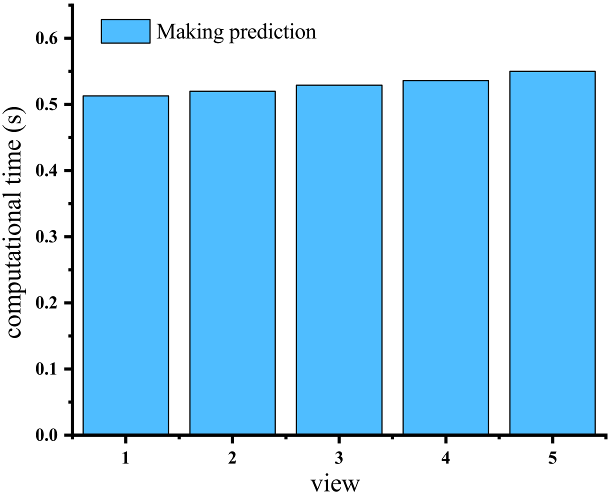


Fig. 10 Computational time in making prediction step for multi-view reconstruction

Moreover, we tested the timing of making prediction for the multi-view reconstruction. As shown in Fig. 10, the prediction time increases slightly as the number of input images increases, and it has no serious impact on the total time.

4.8 Generalization Evaluation

In robotics applications, it is important for intelligent robots to have robust learning capabilities. We chose the bed, motorbike, and bookshelf categories belonging to the PASCAL 3D+ dataset to be fed into our network that was trained by the ShapeNet dataset to evaluate prediction performance. The bed category was predicted to be the most difficult because it is challenging to find categories that are similar in shape to the training set. Table 5 shows that the proposed method obtains sufficient IoU values, and our network predicted volumes reasonably well for all three categories. The reconstruction performance improves when we increase the input views. As shown in Fig. 11, the proposed approach has sufficient generalization abilities for the bookshelf and motorbike categories. As we expected, the bed category is reconstructed the most poorly. This experiment demonstrates that our 2D-3D-attention network model has excellent generalization capabilities for unseen categories and can infer more representations in robotic applications.

Table 5 Predicted IoU values using the 2D-3D-attention tested on bookshelf, bed, and motorbike categories

Input views	1	3	5
Bookshelf	0.369	0.417	0.515
Bed	0.140	0.217	0.228
Motorbike	0.281	0.362	0.514

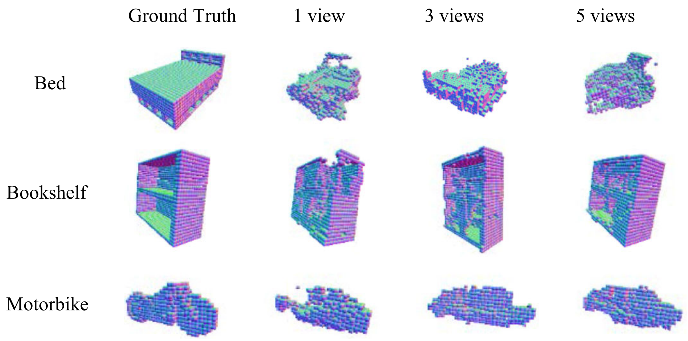


Fig. 11 Results of generalization

5 Conclusions

In this paper, we introduced a learning-based 3D object reconstruction approach. The approach leverages a recurrent convolutional auto-encoder network with the attentive-recurrent visual model. The 2D convolutional LSTM encoder of the network learns long-range dependencies and contains stable learning dynamics, which are effective in extracting features, and the attentive-recurrent visual model adaptively selects useful images for volume prediction. Our experiments show that our model can improve both the accuracy and efficiency of object reconstruction. For robotic applications, the proposed extension model may be promising in achieving scene reconstruction.

Acknowledgements This work is partially supported by the National Natural Science Foundation of China (61873267, U1713224). The authors would also like to express their sincere gratitude to the reviewers and editors for their valuable suggestions.

References

1. Li C, Lu B, Zhang Y et al (2018) 3d reconstruction of indoor scenes via image registration. Neural Process Lett 48(3):1281–1304
2. Orts-Escolano S, Garcia-Rodriguez J, Morell V et al (2016) 3d surface reconstruction of noisy point clouds using growing neural gas: 3d object/scene reconstruction. Neural Process Lett 43(2):401–423

3. Snavely N, Seitz SM, Szeliski R (2006) Photo tourism: exploring photo collections in 3D. In: ACM Siggraph 2006 Papers, pp 835–846
4. Newcombe RA, Izadi S, Hilliges O et al (2011) Kinectfusion: real-time dense surface mapping and tracking. ISMAR 11(2011):127–136
5. Fuentes-Pacheco J, Ruiz-Ascencio J, Rendón-Mancha JM (2015) Visual simultaneous localization and mapping: a survey. Artif Intell Rev 43(1):55–81
6. Soubies E, Blanc-Féraud L, Schaub S, *et al.* (2014) "A 3d model with shape prior information for biological structures reconstruction using multiple-angle total internal reflection fluorescence microscopy," in *2014 IEEE 11th International Symposium on Biomedical Imaging (ISBI)*, 608–611, IEEE
7. Dame A, Prisacariu VA, Ren CY (2013) *et al.*, "Dense reconstruction using 3d object shape priors," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1288–1295
8. Han F, Zhu S-C, (2003) "Bayesian reconstruction of 3d shapes and scenes from a single image," in *First IEEE International Workshop on Higher-Level Knowledge in 3D Modeling and Motion Analysis*, 12–20
9. Chen Y, Cipolla R (2011) Single and sparse view 3d reconstruction by learning shape priors. Computer Vis Image Underst 115(5):586–602
10. Tao L (2014) *3D Non-Rigid Reconstruction with Prior Shape Constraints*. PhD thesis, University of Central Lancashire
11. Wu J, Zhang C, Zhang X, *et al.* (2018) "Learning shape priors for single-view 3d completion and reconstruction," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 646–662
12. Yu J, Tan M, Zhang H et al (2019) Hierarchical deep click feature prediction for fine-grained image recognition. IEEE transactions on pattern analysis and machine intelligence
13. Hong C, Yu J, Chen X (2013) "Image-based 3d human pose recovery with locality sensitive sparse retrieval," in *2013 IEEE international conference on systems, man, and cybernetics*, 2103–2108
14. Yu J, Zhu C, Zhang J et al (2019) Spatial pyramid-enhanced netvlad with weighted triplet loss for place recognition. IEEE Trans Neural Netw Learn Sys 31(2):661–674
15. Yu J, Kuang Z, Zhang B et al (2018) Leveraging content sensitiveness and user trustworthiness to recommend fine-grained privacy settings for social image sharing. IEEE Trans Inf Forensics Secur 13(5):1317–1332
16. Yu J, Zhang B, Kuang Z et al (2016) iprivacy: image privacy protection by identifying sensitive objects via deep multi-task learning. IEEE Trans Inf Forensics Sec 12(5):1005–1016
17. Saxena A, Sun M, Ng AY (2008) Make3d: Learning 3d scene structure from a single still image. IEEE Trans Pattern Anal Mach Intell 31(5):824–840
18. Chang AX, Funkhouser T, Guibas L, *et al.* (2015) "Shapenet: An information-rich 3d model repository," arXiv:1512.03012
19. Lee H, Grosse R, Ranganath R et al (2009) "Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations," in Proceedings of the 26th annual international conference on machine learning, 609–616
20. Kar A, Tulsiani S, Carreira J, *et al.* (2015) "Category-specific object reconstruction from a single image," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1966–1974
21. Su H, Qi CR, Li Y, *et al.* (2015) "Render for cnn: viewpoint estimation in images using cnns trained with rendered 3d model views," in *Proceedings of the IEEE international conference on computer vision*, 2686–2694
22. Girdhar R, Fouhey DF, Rodriguez M et al (2016) "Learning a predictable and generative vector representation for objects," in European conference on computer vision, Springer, 484–499
23. Tatarchenko M, Dosovitskiy A, Brox T (2016) "Multi-view 3d models from single images with a convolutional network," in European conference on computer vision, Springer, 322–337
24. Fan H, Su H, Guibas LJ (2017) "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*, 605–613
25. Wu J, Xue T, Lim JJ et al (2016) "Single image 3d interpreter network," in European conference on computer vision, Springer, 365–382
26. Yan X, Yang J, Yumer E et al (2016) Perspective transformer nets: learning single-view 3d object reconstruction without 3d supervision. Adv Neural Inf Process Sys 29:1696–1704
27. Gadelha M, Maji S, Wang R (2017) "3d shape induction from 2d views of multiple objects," 2017 international conference on 3d vision (3DV), 402–411
28. Choy CB, Xu D, Gwak J et al (2016) "3d-r2n2: a unified approach for single and multi-view 3d object reconstruction," in European conference on computer vision, Springer, 628–644
29. Wang F, Tax DM (2016) "Survey on the attention based rnn model and its applications in computer vision," arXiv:1601.06823
30. Mnih V, Heess N, Graves A (2014) Recurrent models of visual attention. In: Advances in neural information processing systems, pp 2204–2212

31. Ba J, Mnih V, Kavukcuoglu K (2014) "Multiple object recognition with visual attention," arXiv:1412.7755
32. Xu K, Ba J, Kiros R, *et al.* (2015) "Show, attend and tell: neural image caption generation with visual attention," in *International conference on machine learning*, 2048–2057
33. He K, Zhang X, Ren S, *et al.* (2016) "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778
34. Xiang Y, Mottaghi R, Savarese S (2014) "Beyond pascal: A benchmark for 3d object detection in the wild," in *IEEE winter conference on applications of computer vision*, 75–82
35. Tatarchenko M, Dosovitskiy A, Brox T (2017) "Octree generating networks: efficient convolutional architectures for high-resolution 3d outputs," in *Proceedings of the IEEE international conference on computer vision*, 2088–2096