

Meta3D: Single-View 3D Object Reconstruction from Shape Priors in Memory

Shuo Yang^{1,2}, Min Xu², and Hongxun Yao¹

¹ Harbin Institute of Technology, Harbin, China

² University of Technology Sydney, Sydney, Australia
`{syang, h.yao}@hit.edu.cn min.xu@uts.edu.au`

Abstract 3D shape reconstruction from a single-view RGB image is an ill-posed problem due to the invisible parts of the object to be reconstructed. Most of the existing methods rely on large-scale data to obtain shape priors through tuning parameters of reconstruction models. These methods might not be able to deal with the cases with heavy object occlusions and noisy background (Fig. 1) since prior information can not be retained completely or applied efficiently. In this paper, we are the first to develop a memory-based meta-learning framework for single-view 3D reconstruction. A write controller is designed to extract shape-discriminative features from images and store image features and their corresponding volumes into external memory. A read controller is proposed to sequentially encode shape priors related to the input image and predict a shape-specific refiner. Experimental results demonstrate that our Meta3D outperforms state-of-the-art methods with a large margin through retaining shape priors explicitly, especially for the extremely difficult cases.

Keywords: 3D Reconstruction, Meta-learning, Memory Network

1 Introduction

Reconstructing 3D shape from a single-view RGB image is a very vital but challenging computer vision task in the areas of robotics, CAD, virtual and augmented reality. Humans can easily infer the 3D shape from a single image due to sufficient prior knowledge and ability of visual understanding, while it is an extremely difficult and ill-posed problem for a machine vision system because a single-view image can not deliver sufficient information of an object to be reconstructed.

Most of the existing methods use prior shapes implicitly and employ conventional encoder-decoder architectures to predict 3D voxel (volume pixel) grid from a single RGB image. However, implicitly encoding prior knowledge from all available shape priors into the latent parameter space of reconstruction network cannot retain shape information completely or consider different shape prior requirements of various objects differently. In general, only a few shape priors can be very helpful when reconstructing a particular object. Utilizing irrelevant



(a) The occlusion issue

(b) The background issue

Figure 1. Two tricky problems of shape reconstruction from a single-view image. (a) Some parts of object to be reconstructed are invisible due to occlusions. (b) The target object is difficult to be differentiated from its surroundings.

shape priors can possibly introduce noisy and thus make reconstruction worse. The encoder-decoder based approaches seriously suffer from the issues of object occlusions, rough surface and noisy background, which commonly exist in a single-view image. Unsatisfactory performance of these methods are shown in Fig. 1.

Humans can infer a reasonable 3D shape from a single image with incomplete visual cues because humans can retrieve similar shape priors from their memory and apply these the shape priors to repair the invisible parts of the object. Inspired by memory-based meta-learning, which uses an external memory to store prior knowledge and achieves fast domain adaptation by exploiting the relationship between the current input and items in memory, we propose a framework named Meta3D. Meta3D transforms the usage of prior shapes from implicit to explicit by storing the shape priors into external memory, as shown in Fig. 2. The Meta3D can store existing ‘image-3D shape’ pairs into the memory slots with a novel write controller. Later, for each input image, a read controller is carefully designed to retrieve the top few relevant 3D shapes to the input from memory and followed by an LSTM, which is introduced to contextually encode the knowledge useful for reconstruction among retrieved shape priors to synthesis the parameters of a shape-specific refiner. Our main contributions are summarized as follows:

- We are the first to propose a memory-based meta-learning approach, *i.e.*, Meta3D, for single-view 3D reconstruction. The proposed Meta3D can reconstruct reasonable 3D shape even for the object with invisible parts and complex background by explicitly utilizing shape priors and fully exploiting the relationship between a single-view image and shapes priors in memory.
- A novel write controller is devised to treat the memory update as an image-shape feature aggregation problem. Our write controller makes images with similar volumes (3D shapes) closer to each other while pulling away from images with different volumes. Moreover, a novel read controller with a parameter prediction network is proposed to encode the most input-relevant

prior shapes sequentially in the memory and form a *shape-specific-refiner* for each object to be reconstructed.

- Experimental results on three popular benchmarks demonstrate that Meta3D outperforms state-of-the-art methods with a large margin. Through select input-relevant prior shapes and apply priors efficiently, the proposed Meta3D can deal with extremely difficult cases, *e.g.*, objects with truncated and occluded parts or with very complex background, which cannot be handled by other methods.

2 Related Work

2.1 Single-image 3D Reconstruction

Recently, 3D shape reconstruction from a single-view image has attracted increasing research efforts because of its wide applications in the real world. Recovering object shape from a single-view image is an ill-posed problem due to the limitation of visual clues. Existing works use the representation of silhouettes [7], shading [24], and texture [38] to recover 3D shape. With the success of deep learning, especially the generative adversarial networks [12] and variational autoencoders [15], the deep neural network based encoder-decoder has become the main-stream architecture, such as 3D-VAE-GAN [40]. MarrNet [39] reconstructs 3D objects by estimating depth, surface normals, and silhouettes. PSGN [8] and 3DLMNet [17] generate point representations from single-view images. 3D-R2N2 [6] applies a 2D CNN to encode the input single-view image into a feature map. Then a 3D convolutional neural network was used to decode the feature representation into a 3D shape. Tulsiani *et al.* [33] adopt an unsupervised solution for 3D object reconstruction. However, all these works suffer from the issues of object occlusion and indistinguishable foreground and background [1]. Different from existing works using shape priors in an implicit way by tuning parameters of reconstruction networks, we propose a framework, so-called Meta3D that can explicitly select and use the most relevant prior shapes to guide the reconstruction and has superiority of reconstructing invisible parts of the object.

2.2 Memory-based Meta-learning

Meta-learning aims to research how to distill prior knowledge from the past experience and enable fast adaptation to new, even unseen, tasks with only a limited amount of samples. The main-stream approaches of meta-learning can be broadly categorized into three groups: optimization-based [22][10][11][25], metric-based [34][28][23][2][31] and memory-based [26]. The MANN [26] is the very first work to demonstrate the ability of a memory-augmented neural network to rapidly assimilate new data and leverage them to make accurate predictions after only a few samples.

Memory Network is first proposed in [36], which augmented neural networks with an external memory module that enables the neural network to store long-term memory. Then later works [29][16] improve the Memory Networks to be trained in an end-to-end manner. Hierarchical Memory Networks [4] is proposed to allow read controller to efficiently access large scale memories. Key-Value Memory Networks [18] stores prior knowledge in a key-value structured memory, keys are used to address relevant memories whose corresponding values are returned. We introduce and improve the memory-based approach in our Meta3D since the memory module has the ability to store critical information over long periods which is close to human perception. In this paper, we utilize memory network to store shape priors so that prior information can be retained completely. Novel write and read strategies are devised for our specific task.

2.3 Parameter Prediction

The parameter prediction is one of the meta-learning strategies. Parameter prediction refers to evolve one network to generate parameters for another network, which is an effective way to encode relational information into the network and make the network adaptive for novel samples. Early work in [1] is to explore the prediction of the weight parameters in deep neural network, which trains a multi-layer perceptron to predict a binary classifier for class-specific description in text. The authors of [27] suggest the fast weights in which one network can produce the changes of context-dependent weights for second networks. A few subsequent works study practical applications with the fast parameters prediction, *e.g.*, objects detection [35] and image super-resolution [14]. We devise a shape-specific parameter predictor to achieve fast shape domain adaption so that our framework can refine the reconstructed volumes shape by shape.

3 Method Overview

The basic idea of our Meta3D is to refine generated coarse volume with relevant prior shapes stored in memory during the training stage. To store and utilize shape priors explicitly, we adopt a Key-Value Memory Network [19] as our memory module to store shape-discriminative feature of input image as ‘Key’ and it’s corresponding ground-truth volume as ‘Value’.

As illustrated in Fig. 2, our framework is composed of the following modules: an external memory module (which including many memory slots, an shape-discriminative feature extractor, a write controller and a read controller), a shape-agnostic volume generator, and a shape-specific refiner. The feature extractor is used to extract shape-discriminative features from input images with the help of novel devised shape threshold triplet loss. The write controller takes extracted shape-discriminative features and their corresponding ground-truth volumes as input, and determines how to store the ‘image-volume’ pairs into the memory. The read controller can retrieve the most relevant volumes of input image by computing similarities between extracted features of input image and

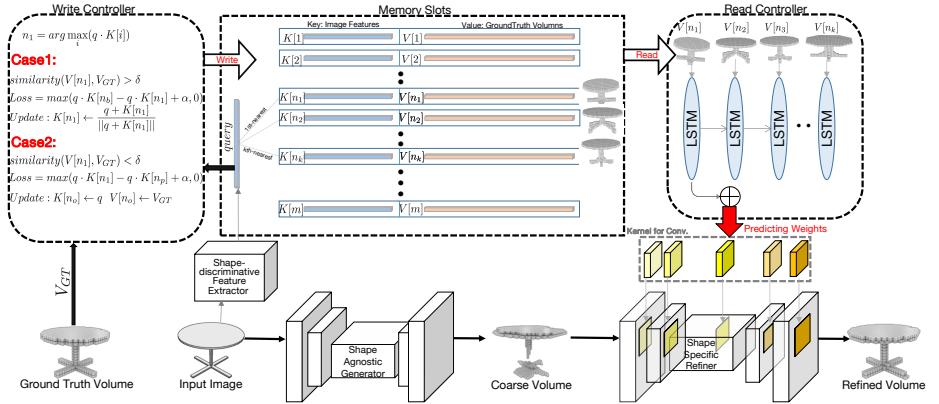


Figure 2. An illustration of Meta3D. In the training stage, the shape-discriminative feature and the ground-truth volume of the input image are stored in the memory slots in a Key-Value structure according to the write strategy. When inferring, the read controller reads shape priors from the memory according to shape-discriminative features extracted from input image and predict parameters of shape-specific refiner, which is used to refine the coarse volume generated by the shape-agnostic generator, using these shape priors.

the ‘Key’s stored in the memory. After several volumes are retrieved, the read controller contextually encodes all the retrieved volumes to synthesis parameters of the *shape-specific refiner* which is adaptive for each input image.

4 The Proposed Method

In this section, we will introduce the external memory module, the shape-agnostic generator module and the shape-specific refiner module individually.

4.1 External Memory Module

Employing RNN based models is a natural way to encode contextual relationships and similarities among shape priors. The latent parameter space of RNN is treated as memory, which may be too small to completely remember prior shape information. Reconstructing a particular object may only rely on a subset of shape priors. However, the latent parameter space of RNN is fixed after training without considering different shape prior requirements of various objects. Inspired by the Memory Network [37] and the Key-Value Memory Network [19], we design an external memory module for 3D reconstruction that can manipulate a large external memory module and store shape priors in a flexible Key-Value mode. By explicitly storing and utilizing shape priors, our Meta3D can construct a relevant shape prior subset to guide the reconstruction for a particular object.

We construct the memory items as: $[key, value, age]$, which is denoted as $M = ((K[i], V[i], A[i]))_{i=1}^m$, m means that the memory has m slots to store. A

‘key’ memory slot stores a D_m -dimensional shape-discriminative feature of the input image which is denoted as $K[i] \in \mathbb{R}^{D_m}$, and the keys will be used to compute the cosine similarities with the queries. A ‘value’ memory stores the ground truth 3D volume $V[i] \in \mathbb{R}^{32^3}$ of its corresponding key. The age $A[i]$ keeps track of the age of items not being used so that we can choose rarely used slots to overwrite.

Shape-discriminative Feature Extractor and Write Controller. It is worth noting that the memory slots M^m is a set of constructed data without any trainable parameters. The construction of memory slots relies on the extracted image features and the write strategy. The write controller takes both extracted features (query) q^{D_m} and ground truth volumes v^{n^3} as input, while the write controller determines how to update the memory slots M and how to optimize the shape-discriminative feature extractor. Different from feature extractor for image retrieval [44], which aims to make images of the same semantic or visual classes closer to each other while making images of different semantic or visual classes further away, our shape-discriminative feature extractor needs to have the ability to retrieve shape-similar images with the input image q^{D_m} instead of appearance-similar images. Because images with similar semantic or visual classes may not have similar shapes. Thus we devise an objective function to treat the feature extraction as an image-volume feature aggregation problem to make images with similar volumes closer to each other while pulling away from those images with different volumes. We define the positive neighbor n_p of the input ground truth volume v as the memory slot with the smallest index where the similarity between $V[n_p]$ and v is over a threshold δ , i.e.,

$$\text{Similarity}(V[n_p], v) > \delta, \quad (1)$$

$$\text{Similarity}(V[n_p], v) = 1 - \frac{1}{n^3} \sum_{i=1}^{n^3} (V[n_p]^i - v^i)^2, \quad (2)$$

Similarly, the negative neighbor n_b is the memory slot with the smallest index where the similarity between $V[n_b]$ and v is less than a threshold δ , i.e.,

$$\text{Similarity}(V[n_b], v) < \delta, \quad (3)$$

$$\text{Similarity}(V[n_b], v) = 1 - \frac{1}{n^3} \sum_{i=1}^{n^3} (V[n_b]^i - v^i)^2, \quad (4)$$

$$V[n_p] \in \mathbb{R}^{n^3}, \quad V[n_b] \in \mathbb{R}^{n^3}, \quad v \in \mathbb{R}^{n^3}, \quad (5)$$

Where n^3 denotes the dimension of the 3D volume is $n \times n \times n$, i.e. the n equals to 32 in the ShapeNet dataset.

The triplet loss is constructed as

$$L(q, M, \delta) = \max(q \cdot K[n_b] - q \cdot K[n_p] + \alpha, 0), \quad (6)$$

Which can minimize the distance among images with similar 3D volumes and maximize the distance among images with different 3D volumes.

After extracting the shape-discriminative features of the input RGB image as query q , the write controller takes the q and it's corresponding ground truth volume v as input. To update the memory slots, two write strategies are considered: 1) Aggregating the similar samples into one memory slot by updating the corresponding memory key. 2) Storing the distinctive samples into a new memory slot. It depends on the similarity between the q and it's 1-st nearest neighbor $V[n_1]$ in the memory M .

$$n_1 = \arg \max_i (q \cdot K[i]), \quad (7)$$

Case 1: If the similarity between $V[n_1]$ and v is over the threshold δ , we update the key $K[n_1]$ and set the age of the slot $A[n_1]$ with zero, but keep the $V[n_1]$ unchanged.

Case 2: If the similarity between $V[n_1]$ and v is less than the threshold δ , it means that there is no memory slot that storing a similar volume with the current input v . So we need to seek an oldest memory slot $M[n_o]$ (with the largest age $A[n_o]$) to store the new input $[q, v]$.

$$\begin{cases} K[n_1] = \frac{q + K[n_1]}{\|q + K[n_1]\|}, A[n_1] = 0, & \text{if } \text{Similarity}(V[n_1], v) > \delta, \\ K[n_o] = q, V[n_o] = v, A[n_o] = 0, & \text{otherwise}. \end{cases} \quad (8)$$

It is worth noting that the write controller only works when performing the training process because it takes the ground truth volumes as inputs that are invisible in testing.

Read Controller with Shape-Specific Parameter Prediction. The ultimate aim of our Meta3D is to read and construct a object-relevant prior shape subset from the memory to guide the particular object reconstruction. Inspired by the success in fast domain adaption and few-shot learning on the fast parameters generation[3][27][20], we utilize an LSTM network [13] to sequentially encode the relationships and contextual information of the k specific shape priors $V[n_1], V[n_2], \dots, V[n_k]$ read from the memory and generate the *shape-specific-parameter* W_q^r for the refiner r to guide the reconstruction of the specific object q . The procedure is formulated as

$$(n_1, n_2, \dots, n_k) = \text{NearestNeighbor}_k(q, K), \quad (9)$$

$$W_q^r = \mathbf{X} \cdot \text{lstm}(V[n_1], V[n_2], \dots, V[n_k]), \mathbf{X} \in \mathbb{R}^{D_w \times D_{lstm}}, \quad (10)$$

Where q is the query which is extracted shape-discriminative features, K is Key slots in the memory. We compute the cosine similarities between q and all the

keys $K[1], K[2], \dots, K[m]$ and choose k volumes $V[n_1], V[n_2], \dots, V[n_k]$ with the top- k similarities as the k *specific shape priors* of query q . The $\mathbf{X} \in \mathbb{R}^{D_w \times D_{lstm}}$ is a weight matrix of the full-connected layer FC which transform the D_{lstm} -dimension outputs of the LSTM to the D_w -dimension of the refiner weights. The generated parameter W_q^r will be applied to the volume refiner to make it a *shape-specific volume refiner*.

4.2 Shape-Agnostic Generator Module

This module is used to generate coarse 3D shapes from input RGB images. The 3D shape of an object is represented by a 3D voxel grid, where 0 is an empty cell and 1 denotes an occupied cell. The term of ‘shape agnostic’ indicates that this generator is trained using all kinds of objects, therefore it is a generic solution to reconstruct coarse 3D models. To make a fair comparison to prove the effectiveness of our memory module, we adopt the same network architecture of the current state-of-the-art work [43]. The generator is composed of an encoder and a decoder, the encoder takes images as input and computes 2D feature maps of images. The decoder is responsible for transforming the 2D feature maps into 3D volumes. The details of the network architecture will be discussed in the experiment section 5.2. The reconstruction loss function of generator is formulated as

$$L_{rec}(GT, v_g) = \frac{1}{n^3} \sum_{i=1}^{n^3} [GT^i \log(v_g^i) + (1 - GT^i) \log(1 - v_g^i)], \quad (11)$$

where n^3 is the dimension of the 3D volume, v_g and GT represents the predicted volume of the generator and the corresponding ground truth.

4.3 Shape-Specific Refiner Module

This module is used to correct and refine coarse volumes v_g . ‘Shape-specific’ is used to emphasize that the refiner is able to adapt different input images, i.e. different shapes, through applying *shape-specific parameters*. The *shape-specific parameters* W_q^r are produced by the *LSTM* and *FC* networks mentioned above. It takes v_g as input and produces a refined volume v_r .

$$v_r = Refiner_q(v_g | W_q^r), \quad (12)$$

The parameter prediction loss of the *LSTM* and *FC* networks is computed as

$$\begin{aligned} L_{pred} &= L_{rec}(GT, v_r) \\ &= L_{rec}(GT, Refiner_q(v_g | W_q^r)) \\ &= L_{rec}(GT, Refiner_q(v_g | \mathbf{X} \cdot lstm(V[NN_k[q, K]]))) , \end{aligned} \quad (13)$$

Then the parameter prediction network *LSTM* and *FC* performs backward propagation using the L_{pred} to minimize the prediction error.

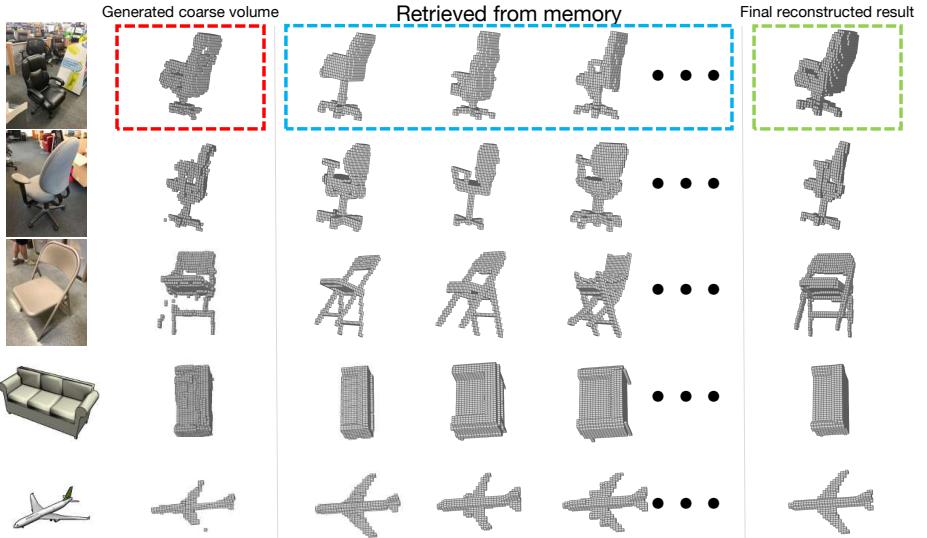


Figure 3. The left images are inputs, the coarse volume in red box is produced by the shape-agnostic generator, the volumes in blue box are shapes retrieved from external memory according to the input image, and the right volumes are the final reconstruction results refined by considering the retrieved shape priors.

5 Experiments

5.1 Datasets and Metrics

Datasets ShapeNet [5], Pix3D [30] and PASCAL 3D+ [42] are used to evaluate the performance of our Meta3D, which are the most commonly used public datasets in single-view image 3D reconstruction. The ShapeNet, which is used to learn shape priors, is composed of synthetic images and corresponding 3D volumes. The evaluate datasets, Pix3D and the PASCAL 3D+, are much more challenging because the images are from the real-world and contain noisy background and occlusions, as shown in Fig. 1. We evaluate our Meta3D on both synthetic and real-world images to demonstrate the effectiveness of our model to handle the complicated self-occlusion, noisy background, and truncation issues.

Evaluation Metrics We apply the intersection over union (IOU) and Chamfer Distance (CD) evaluation metrics widely used by existing works. The IOU measures the similarity between ground-truth and reconstructed voxels. Which can be formulated as

$$IoU = \frac{\sum_{i,j,k} \mathbb{I}(p(i,j,k) > t) \mathbb{I}(gt(i,j,k))}{\sum_{i,j,k} \mathbb{I}[\mathbb{I}(p(i,j,k) > t) + \mathbb{I}(gt(i,j,k))]}, \quad (14)$$

where $p(i, j, k)$ and $gt(i, j, k)$ indicate predicted occupancy probability and ground-truth at (i, j, k) , respectively. \mathbb{I} is the indication function which will equal to one when the requirements are satisfied. The t denotes a threshold. The Chamfer distance between two point clouds $P_1, P_2 \in \mathbb{R}^3$ is defined as

$$CD(P_1, P_2) = \frac{1}{|P_1|} \sum_{x \in P_1} \min_{y \in P_2} \|x - y\|_2 + \frac{1}{|P_2|} \sum_{x \in P_2} \min_{y \in P_1} \|x - y\|_2, \quad (15)$$

For each point in each set, CD finds the closest point in the other set and average the distances. We sample points in the voxel isosurface to compute the CD for voxel occupancy as same as [21].

5.2 Implementation Details

Most of our settings follow previous works to make a fair comparison. Specifically, we resize input images into 224x224 and downsample the voxels provided by the official repository into 32^3 . We train all the modules with Adam optimizer with $\beta_1 = 0.9$ and $\beta_2 = 0.999$. The initial learning rate is set to 0.0001 and decayed by 2 every 100 epochs. The modules and their hyperparameters will be introduced below.

Memory Module In the memory module, we use a ResNet pre-trained on Imagenet as our shape-discriminative feature extractor. We use the feature extracted by the pool5 layer of the ResNet18, the dimension of the feature is 512, the volume similarity threshold δ is set as 0.7, 0.9, and 0.9 in ShapeNet, Pix3D and PASCAL 3D+, respectively. The margin α in the triplet loss is 0.1. We set the size of the memory as 20,000, which means the memory module has storage capacity to store 20,000 items. One item is $[key, value, age]$. The k in the read controller is 256 and the size of the hidden layer in LSTM is 512.

Generator Module We follow the same generator architecture as previous work [43] to make comparison fair enough. Specifically, the generator is composed of an encoder and a decoder. The encoder adopts the first nine convolutional layers of VGG pre-trained on Imagenet and followed by three sets of 2D convolutional layers, batch normalization layers, and ELU layers. The kernel sizes are 3^2 , 3^2 and 1^2 , respectively. The decoder is composed of five 3D transposed convolutional layers. The first four transposed convolutional layers are of a kernel size of 4^3 , with a stride of 2 and padding of 1. The fifth transposed convolutional layer with a bank of 1^3 filter. All the transposed layers are followed by a batch normalization layer and a RELU except for the last layer followed by a sigmoid function. More details can be found in [43].

Refiner Module The encoder of the refiner has three 3D convolutional layers, each of which has a bank of 4^3 filters with padding of 2, followed by a batch

normalization layer, a leaky ReLU and a max-pooling layer with a kernel size of 2^3 . The encoder is followed by two fully connected layers with the dimension of 2048 and 8192. The decoder consists of three transposed convolutional layers, with a bank of 4^3 filters with padding of 2 and stride of 1. And all the layers except for the last one followed by a batch normalization layer and a ReLU activation. The last layer is followed by a sigmoid function.

5.3 Reconstruction on ShapeNet

Same as previous works [6], we use a subset of ShapeNet consisting of 13 major categories and 43,783 3D models, in which the voxel resolution is 32^3 . In memory training stage, we train a shape-discriminative feature extractor and apply a write controller to fill the memory slots. Training of the generator and the parameter predictor is performed in the next stage. The experimental results are shown in Table. 1. As shown in Table. 1, although several methods take extra information as input (The PSGN needs the objects masks and uses 220k 3D CAD models), our Meta3D over performs all other method with a large margin cross all categories. Our Meta3D benefits from the external memory module which can explicitly retain complete shape priors and apply them according to object's individual needs through an efficient fast domain adaption.

Table 1. Single-view reconstruction results on ShapeNet comparison using Intersection-over-Union (IoU). DRC is trained/tested category by category. PSGN takes ground-truth objects masks as an additional input. PSGN uses 220k 3D CAD models while others use 44k 3D CAD models during training.

Category	3D-R2N2 [6]	OGN [32]	DRC [33]	PSGN [9]	Pix2Vox [43]	Meta3D
airplane	0.513	0.587	0.571	0.601	0.684	0.732
bench	0.421	0.481	0.453	0.550	0.616	0.667
cabinet	0.716	0.729	0.635	0.771	0.792	0.823
car	0.798	0.828	0.755	0.831	0.854	0.880
chair	0.466	0.483	0.469	0.544	0.567	0.622
display	0.468	0.502	0.419	0.552	0.537	0.591
lamp	0.381	0.398	0.415	0.462	0.443	0.501
speaker	0.662	0.637	0.609	0.737	0.714	0.761
rifle	0.544	0.593	0.608	0.604	0.615	0.643
sofa	0.628	0.646	0.606	0.708	0.709	0.755
table	0.513	0.536	0.424	0.606	0.601	0.627
cellphone	0.661	0.702	0.413	0.749	0.776	0.810
watercraft	0.513	0.632	0.556	0.611	0.594	0.633
overall	0.560	0.596	0.545	0.640	0.661	0.696

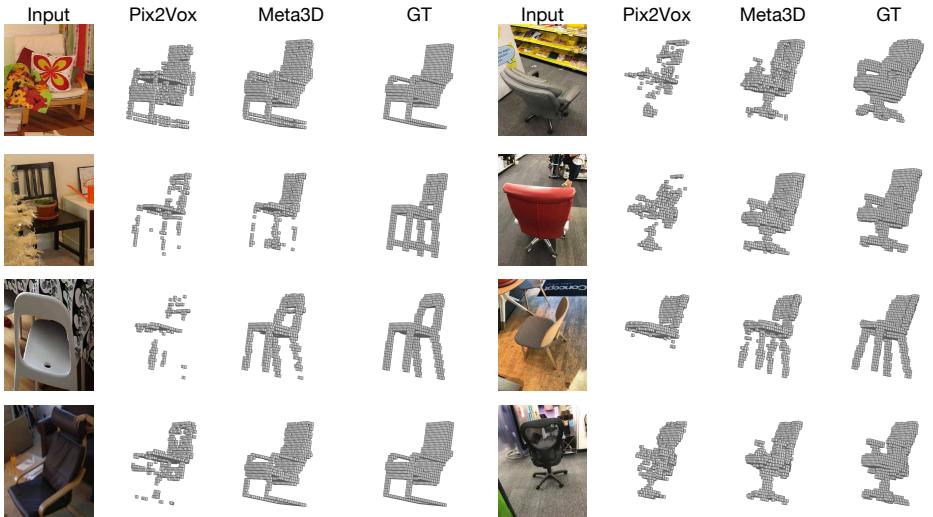


Figure 4. Single-view reconstruction results on Pix3D. GT represents the ground-truth volumes of the 3D objects. The results are compared with the state-of-the-art method Pix2Vox[43].

5.4 Reconstruction on Pix3D

Pix3D is a large-scale benchmark of diverse image-shape pairs. The most significant category in this dataset is chairs. Most of the previous works [43][30][41] evaluate their approaches using the hand-selected 2894 untruncated and unoccluded ‘chair’ images. In this work, we evaluate our Meta3D on both 2895 untruncated and unoccluded ‘chair’ images and the truncated and occluded ‘chair’ set to demonstrate the ability of our Meta3D to handle the challenges of having invisible parts and noisy background.

The training procedure is performed on the ShapeNet-Core[5] dataset which contains over 50k object instances of 55 categories. In the first training stage, we use ShapeNet-Core[5] dataset to help the shape-agnostic generator learn shape priors and optimize the shape-discriminative feature extractor and parameter predictor. At the next training stage, we empty the memory slots and train with a higher volume similarity threshold $\delta = 0.95$ on the subset only containing ‘chairs’ to store more diverse ‘chair’ shape priors. We follow the previous work and use the same rendered view images in [21] and [6].

First of all, we evaluate our Meta3D at 2894 untruncated and unoccluded chair images same as most of the previous works and compared with the state-of-the-art methods, as shown in Table. 2. Note that these methods use different types of extra information. For instance, MarrNet [39], DRC [33] and ShapeHD [41] use extra depth, surface normals and silhouettes information and PSGN [8] takes objects masks as input. Although our Meta3D uses only an RGB image as input, Meta3D still surpasses the state-of-the-art method by a large

Table 2. Single-view reconstruction on Pix3D compared using Intersection-over-Union (IoU) and Chamfer Distance (CD). The best performance is highlighted in bold.

Method	IoU	CD
3D-R2N2[6]	0.136	0.239
3D-VAE-GAN [40]	0.171	0.182
MarrNet [39]	0.231	0.144
DRC [33]	0.265	0.160
ShapeHD [41]	0.284	0.123
DAREC [21]	0.241	0.140
Pix3D [30]	0.282	0.119
Meta3D (w/o replacing slots)	0.331	0.101
Meta3D (w/ replacing slots)	0.339	0.094

margin. The result also shows that replacing all the shapes in memory slots into shapes of ‘chair’ performs better. This may be caused by the limitation of the memory capacity to store all shapes of ‘chair’.

We make another comparison on the truncated and occluded ‘chair’ images of the Pix3D dataset, which is a extremely challenging task. The experimental results are shown in Table. 3 and Fig. 4. The performance of all other methods is decreased significantly, while our Meta3D can handle the hardest samples better than other methods. The external memory module transforms the usage of shape priors from implicit to explicit and construct a relevant shape priors subset according to different object’s needs. Our Meta3D extracts the feature of the input and retrieves relevant shape priors from the memory to guide the reconstruction process. Using retrieved 3D volumes, which are clean and complete, to guide the reconstruction can significantly filter out the background noise and refine the generated coarse shapes. Fig. 3 shows some examples of retrieved volumes from memory and refined results.

Table 3. Single-view reconstruction on truncated and occluded Pix3D comparison using Intersection-over-Union (IoU) and Chamfer Distance (CD). The best performance is highlighted in bold.

Method	IoU	CD
3D-R2N2[6]	0.055	0.497
3D-VAE-GAN [40]	0.091	0.401
MarrNet [39]	0.138	0.356
DRC [33]	0.151	0.331
ShapeHD [41]	0.183	0.269
Pix3D [30]	0.175	0.251
Meta3D (w/o replacing slots)	0.289	0.193
Meta3D (w/ replacing slots)	0.293	0.189

5.5 Reconstruction on Pascal 3D+

Similar to previous works, we use PASCAL 3D+ to evaluate instead of training. Our model learns the shape priors from ShapeNet. We train our model using the categories that are present in both PASCAL 3D+ and ShapeNet renderings: ‘aeroplane’, ‘car’, ‘chair’, ‘table’, and ‘tv’. Comparison results of three categories reported in [21][33][41] are shown in Table.4. Note that DRC and ShapeHD use depth/normal/silhouettes as extra information during training. Only taking single-view RGB images as input, our Meta3D still achieves the lowest CD metric by explicitly retaining and using the shape priors.

Table 4. Single-view reconstruction on PASCAL 3D+ compared using CD metrics. The best performance is highlighted in bold.

Method	chair	car	plane	average
3D-R2N2[6]	0.238	0.305	0.305	0.284
DRC [33]	0.158	0.099	0.112	0.122
OGN [32]	-	0.087	-	-
ShapeHD [41]	0.137	0.129	0.094	0.119
DAREC [21]	0.135	0.101	0.108	0.115
Meta3D	0.110	0.065	0.071	0.082

All the experiments and comparison demonstrate the superiority of our Meta3D on 3D shapes reconstruction from a single-view image. Our method can efficiently clean and complete the generated coarse volumes, and handle the self-occlusion and the diverse noisy background very well.

6 Conclusions

In this work, we proposed a Meta3D network that can explicitly store the shape priors into an external memory module, then retrieve and apply priors according to object’s individual needs through an efficient fast domain adaption. The novel devised write and read controller provide the memory module the ability to aggregate shape similar images and encode the shape priors effectively. Experimental results on both synthetic image 3D reconstruction and real-world 3D reconstruction demonstrate the superiority of our Meta3D. The experiments on occluded and truncated images also demonstrates that our Meta3D can handle more difficult samples, which makes our Meta3D more valuable in real-world applications.

References

1. Ba, L.J., Swersky, K., Fidler, S., Salakhutdinov, R.: Predicting deep zero-shot convolutional neural networks using textual descriptions. In: 2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015. pp. 4247–4255 (2015)
2. Bertinetto, L., Henriques, J.F., Torr, P.H.S., Vedaldi, A.: Meta-learning with differentiable closed-form solvers. In: 7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019 (2019)
3. Bertinetto, L., Henriques, J.F., Valmadre, J., Torr, P.H.S., Vedaldi, A.: Learning feed-forward one-shot learners. CoRR **abs/1606.05233** (2016)
4. Chandar, S., Ahn, S., Larochelle, H., Vincent, P., Tesauro, G., Bengio, Y.: Hierarchical memory networks (2016)
5. Chang, A.X., Funkhouser, T.A., Guibas, L.J., Hanrahan, P., Huang, Q., Li, Z., Savarese, S., Savva, M., Song, S., Su, H., Xiao, J., Yi, L., Yu, F.: Shapenet: An information-rich 3d model repository. CoRR **abs/1512.03012** (2015)
6. Choy, C.B., Xu, D., Gwak, J., Chen, K., Savarese, S.: 3d-r2n2: A unified approach for single and multi-view 3d object reconstruction. In: Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VIII. pp. 628–644 (2016)
7. Dibra, E., Jain, H., Oztireli, C., Ziegler, R., Gross, M.: Human shape from silhouettes using generative hks descriptors and cross-modal neural networks. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (July 2017)
8. Fan, H., Su, H., Guibas, L.J.: A point set generation network for 3d object reconstruction from a single image. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017. pp. 2463–2471 (2017)
9. Fan, H., Su, H., Guibas, L.J.: A point set generation network for 3d object reconstruction from a single image. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (July 2017)
10. Finn, C., Abbeel, P., Levine, S.: Model-agnostic meta-learning for fast adaptation of deep networks. In: Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017. pp. 1126–1135 (2017)
11. Finn, C., Xu, K., Levine, S.: Probabilistic model-agnostic meta-learning. In: Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada. pp. 9537–9548 (2018)
12. Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A.C., Bengio, Y.: Generative adversarial nets. In: Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada. pp. 2672–2680 (2014)
13. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Computation **9**(8), 1735–1780 (1997)
14. Hu, X., Mu, H., Zhang, X., Wang, Z., Tan, T., Sun, J.: Meta-sr: A magnification-arbitrary network for super-resolution. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2019)

15. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. In: 2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings (2014)
16. Kumar, A., Irsoy, O., Ondruska, P., Iyyer, M., Bradbury, J., Gulrajani, I., Zhong, V., Paulus, R., Socher, R.: Ask me anything: Dynamic memory networks for natural language processing. In: Proceedings of the 33rd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016. pp. 1378–1387 (2016)
17. Mandikal, P., L., N.K., Agarwal, M., Radhakrishnan, V.B.: 3d-lmnet: Latent embedding matching for accurate and diverse 3d point cloud reconstruction from a single image. In: British Machine Vision Conference 2018, BMVC 2018, Northumbria University, Newcastle, UK, September 3-6, 2018. p. 55 (2018)
18. Miller, A.H., Fisch, A., Dodge, J., Karimi, A., Bordes, A., Weston, J.: Key-value memory networks for directly reading documents. CoRR [abs/1606.03126](https://arxiv.org/abs/1606.03126) (2016)
19. Miller, A.H., Fisch, A., Dodge, J., Karimi, A., Bordes, A., Weston, J.: Key-value memory networks for directly reading documents. In: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016. pp. 1400–1409 (2016)
20. Noh, H., Hongsuck Seo, P., Han, B.: Image question answering using convolutional neural network with dynamic parameter prediction. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2016)
21. Pinheiro, P.O., Rostamzadeh, N., Ahn, S.: Domain-adaptive single-view 3d reconstruction. In: The IEEE International Conference on Computer Vision (ICCV) (October 2019)
22. Ravi, S., Larochelle, H.: Optimization as a model for few-shot learning. In: 5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings (2017)
23. Ren, M., Triantafillou, E., Ravi, S., Snell, J., Swersky, K., Tenenbaum, J.B., Larochelle, H., Zemel, R.S.: Meta-learning for semi-supervised few-shot classification. CoRR [abs/1803.00676](https://arxiv.org/abs/1803.00676) (2018)
24. Richter, S.R., Roth, S.: Discriminative shape from shading in uncalibrated illumination. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2015)
25. Rusu, A.A., Rao, D., Sygnowski, J., Vinyals, O., Pascanu, R., Osindero, S., Hadsell, R.: Meta-learning with latent embedding optimization. In: 7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019 (2019)
26. Santoro, A., Bartunov, S., Botvinick, M., Wierstra, D., Lillicrap, T.P.: One-shot learning with memory-augmented neural networks. CoRR [abs/1605.06065](https://arxiv.org/abs/1605.06065) (2016)
27. Schmidhuber, J.: Learning to control fast-weight memories: An alternative to dynamic recurrent networks. Neural Computation 4(1), 131–139 (1992)
28. Snell, J., Swersky, K., Zemel, R.S.: Prototypical networks for few-shot learning. In: Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA. pp. 4077–4087 (2017)
29. Sukhbaatar, S., Szlam, A., Weston, J., Fergus, R.: End-to-end memory networks. In: Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada. pp. 2440–2448 (2015)

30. Sun, X., Wu, J., Zhang, X., Zhang, Z., Zhang, C., Xue, T., Tenenbaum, J.B., Freeman, W.T.: Pix3d: Dataset and methods for single-image 3d shape modeling. In: 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018. pp. 2974–2983 (2018)
31. Sung, F., Yang, Y., Zhang, L., Xiang, T., Torr, P.H.S., Hospedales, T.M.: Learning to compare: Relation network for few-shot learning. In: 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018. pp. 1199–1208 (2018)
32. Tatarchenko, M., Dosovitskiy, A., Brox, T.: Octree generating networks: Efficient convolutional architectures for high-resolution 3d outputs. In: The IEEE International Conference on Computer Vision (ICCV) (Oct 2017)
33. Tulsiani, S., Efros, A.A., Malik, J.: Multi-view consistency as supervisory signal for learning shape and pose prediction. In: 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018. pp. 2897–2905 (2018)
34. Vinyals, O., Blundell, C., Lillicrap, T., Kavukcuoglu, K., Wierstra, D.: Matching networks for one shot learning. In: Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain. pp. 3630–3638 (2016)
35. Wang, Y.X., Ramanan, D., Hebert, M.: Meta-learning to detect rare objects. In: The IEEE International Conference on Computer Vision (ICCV) (October 2019)
36. Weston, J., Chopra, S., Bordes, A.: Memory networks (2014)
37. Weston, J., Chopra, S., Bordes, A.: Memory networks. In: 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings (2015)
38. Witkin, A.P.: Recovering surface shape and orientation from texture. Artif. Intell. **17**(1-3), 17–45 (1981)
39. Wu, J., Wang, Y., Xue, T., Sun, X., Freeman, B., Tenenbaum, J.: Marrnet: 3d shape reconstruction via 2.5d sketches. In: Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA. pp. 540–550 (2017)
40. Wu, J., Zhang, C., Xue, T., Freeman, B., Tenenbaum, J.: Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling. In: Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain. pp. 82–90 (2016)
41. Wu, J., Zhang, C., Zhang, X., Zhang, Z., Freeman, W.T., Tenenbaum, J.B.: Learning shape priors for single-view 3d completion and reconstruction. In: The European Conference on Computer Vision (ECCV) (September 2018)
42. Xiang, Y., Mottaghi, R., Savarese, S.: Beyond pascal: A benchmark for 3d object detection in the wild. In: IEEE Winter Conference on Applications of Computer Vision (2014)
43. Xie, H., Yao, H., Sun, X., Zhou, S., Zhang, S.: Pix2vox: Context-aware 3d reconstruction from single and multi-view images. In: The IEEE International Conference on Computer Vision (ICCV) (October 2019)
44. Yang, S., Yu, W., Zheng, Y., Yao, H., Mei, T.: Adaptive semantic-visual tree for hierarchical embeddings. In: Proceedings of the 27th ACM International Conference on Multimedia, MM 2019, Nice, France, October 21-25, 2019. pp. 2097–2105 (2019)