





Semi-Supervised Single-View 3D Reconstruction via Prototype Shape Priors(Appendix)

Zhen Xing^{1,2}, Hengduo Li³, Zuxuan Wu^{1,2†}, and Yu-Gang Jiang^{1,2}

¹ Shanghai Key Lab of Intell. Info. Processing, School of CS, Fudan University

² Shanghai Collaborative Innovation Center on Intelligent Visual Computing

³ University of Maryland

This supplementary Appendix contains the following.

- Section 1:Dataset and Augmentation details used in our experiments.
- Section 2:Implementation details of our SSP3D framework.
- Section 3:Implementation details of the 3D extensions of the image-based baselines.
- Section 4:Additional qualitative examples from different datasets.
- Section 5:Additional experiments and ablations.

1 Dataset-Details

ShapeNet The ShapeNet [2] dataset is a collection of 3D CAD models that are organized according to the WordNet hierarchy. We use a subset of the ShapeNet dataset which consists of 43,783 models and 13 major categories as in [3,10]. We follow the dataset split in Pix2Vox [10] with train, valid and test sets. On the basis of Pix2Vox [10] dataset split, we randomly divide the training set into supervised data and unlabeled data according to 1%, 5%, 10% and 20%. The resolution of ShapeNet is $32 \times 32 \times 32$. We adopt the same CenterCrop as in [10] to crop the image size of 224×224 before inputting to the network.

Pix3D Pix3D [8] provides a large-scale dataset of real images and ground-truth shapes with precise 2D-3D alignment. The dataset has 395 3D shapes of nine object categories. Each shape associates with a set of real images, capturing the exact object in diverse environments. Further, the 10,069 image-shape pairs have precise 3D annotations, giving pixel-level alignment between shapes and their silhouettes in the images. We follow the S1-split as in Mesh R-CNN [4] with 7,539 train images and 2,530 test images. Similarly, we randomly sample 10% of each category in the training set as labeled data and use the remaining samples as unlabeled data. The voxel resolution of Pix3D is $128 \times 128 \times 128$.

Data Augmentations The data augmentation strategies in this paper consist of strong augmentation and weak augmentation. We use the public code of data augmentations in Pix2Vox++ [11]. The strong augmentation strategy contains RandomCrop, RandomBackground, ColorJitter, RandNoise, RandomFlip, RandomPermuteRGB, while the weak augmentation only contains CenterCrop, RandomBackground and RandomNoise.

2 Implementation Details

In this section, we provide additional information of implementation details of different components in our SSP3D. We will introduce 3D Autoencoder, Image Encoder, Shape Decoder, Prototype Attentive Module, Shape Naturalness Module respectively.

3D AutoEncoder The 3D AutoEncoder is designed to extract volume feature of 3D voxel for the next clustering task. We set the Autoencoder mainly the same as [6]. The encoder contains four sets of 3D convolutional layers, maxpooling layers and ReLU layers to encode 3D voxel, the kernel size are $5^3, 3^3, 3^3, 3^3$. The output channels of the four convolutional layers are 32, 64, 128, 256 respectively. The decoder contains four sets of 3D transposed convolutional layers with batch normalization and ReLU layers. The kernel size is 4^3 with stride of 2^3 , and padding size of 1. We adopt the output of encoder as 3D feature, which is applied for clustering (*i.e.*, KMeans).

Image Encoder The encoder aims to compute image features for the decoder to recover the 3D shape of the object. We adopt the same backbone as [11]. The first three convolutional blocks of ResNet [5] are used to obtain a 512×28^2 feature map from a $224 \times 224 \times 3$ image. We adopt ResNet-50 [5] from Pix2Vox++ [11] as baseline. ResNet is followed by three sets of 2D convolutional layers, batch normalization layers, and ReLU layers to embed semantic information into feature maps. The kernel sizes of the three convolutional layers are 3^2 , with padding of 1. There is a max pooling layer with a kernel size of 2^2 after the second and third ReLU layers, the kernel size of the four convolutional layers are $5^2, 3^2, 3^2, 3^2$ and the output of the convolutional layers are 32, 64, 128 and 256, respectively.

Shape Decoder The decoder is responsible for transforming information of image feature and prior feature into 3D volumes. There are five 3D transposed convolutional layers. Specically, the first four transposed convolutional layers are of kernel sizes 4^3 , with strides of 2^3 and paddings of 1. There is an additional transposed convolutional layer with a bank of 1^3 filter. Each transposed convolutional layer is followed by a batch normalization layer and a ReLU activation except for the last layer followed by a sigmoid function. The output channel numbers of the five transposed convolutional layers are 512, 128, 32, 8 and 1, respectively. To generate 3D volumes at 128^3 resolution, there are seven transposed convolutional layers in the decoder. The output channel numbers of the seven transposed convolutional layers are 512, 128, 32, 32, 32, 8 and 1, respectively.

Prototype Attentive Module Prototype Attentive Module consists of 3D Encoder that encode 3D voxel prototype and multi-head attention mechanism that obtains prior feature of a query image. We will introduce them in detail respectively. The 3D Encoder contains four sets of 3D convolutional layers, and

ReLU layers to embed volume information into feature maps, the kernel size of the four convolutional layers are $5^3, 3^3, 3^3, 3^3$ respectively. The output of the four convolutional layers are 32, 64, 128 and 256. The multi-head attention consists of three linear layers to transform Q, K, V , the hidden size of the linear layer is 2048, and the heads number of the attention mechanism is 2.

Shape Naturalness Module Shape Naturalness Module is a discriminator to judge if a voxel is true (groundtruth) or fake (predicted volume). For the ShapeNet [2] data with 32^3 resolution, the discriminator contains four sets of 3D convolutional layers, maxpooling layers and ReLU layers to embed volume features, the kernel size of the four convolutional layers are $5^3, 3^3, 3^3, 3^3$ and the output of the convolutional layers are 32, 64, 128 and 256, respectively. As for the Pix3D [8] dataset with 128^3 resolution, we set five sets of 3D convolutional layers and ReLU layers with kernel size of 3^3 , stride size of 2^3 and padding size of 1. Then two linear layers are proposed of hidden size of 128 and 1 with ReLU and Sigmoid activation function.

3 Image-based Baseline Details

This section provides implementation details of different baselines used in the paper. We adhere to the base approach proposed in the original works of the respective baselines for all our experiments. Note that, for a given image, same set of augmentations have been applied to all images so that they go through the same set of transformations. The initial learning rate is set to $1e-3$ with decay to $1e-4$ in all our baseline experiments unless stated otherwise. All the baseline models are trained for 250 epochs unless otherwise specified.

Supervised We use the code made public by the authors in Pix2Vox [10] and Pix2Vox++ [11] for the supervised baseline. It is trained using \mathcal{L}_{rec} for 250 epochs and the initial learning rate is kept same as it. Other hyperparameters are kept same as the ones used for the respective datasets in [10]. Note that for simplicity compared, we only adopt the Encoder and Decoder in [11] as supervised baseline and remove the Merger and Refiner module which are particularly designed for multi-view 3D reconstruction.

MeanTeacher The model is trained using the philosophy described in [9]. In this scenario, we have two models, one is the student network and the other is the teacher network. The teacher network has the same backbone architecture as the student. The weights of the teacher network are exponential moving average weights of the student network. Consistency is ensured between the output predicted by the teacher and the student for the unlabeled images. The labeled data, in addition, is trained using \mathcal{L}_{sup} . We use L2 loss as the consistency cost and warm up its weight from 0 to its final value during the first 80 epochs as in [9].

MixMatch We follow the approach in [1] to train our MixMatch baseline approach. We apply 3 different augmentations to unlabeled images set (U) and then computed the average of the predictions across these augmentations. We use the random strong data augmentations as described in subsection 1 in our experiments. The average predictions of K different augmentations are used as labels for the unlabeled images. Then, labeled (L) and unlabeled images with their targets and predicted labels are shuffled and concatenated to form another set W which serves as a source for modified MixUp algorithm defined in [12]. Then for each i^{th} labeled image we compute $\text{MixUp}(L_i, W_i)$ and add the result to a set V' . It contains the MixUp of labeled images with W . Similarly for each j^{th} unlabeled image, we compute $\text{MixUp}(U_i, W_i + |L|)$ and add the result to another set U' . It contains the MixUp of unlabeled images with rest of W . A \mathcal{L}_{rec} loss between labels and model predictions from V' and L2 loss between the predictions and pseudo labels from U' are used for training.

FixMatch For extending the FixMatch [7] baseline to 3D reconstruction methods, we primarily follow the same augmentation and consistency regularization policies laid out in [7]. The images are passed through two different augmentation pathways. In the first pathway, the input images are weakly augmented and used to obtain the pseudo-labels. In the second pathway, the strongly augmented version of the same images are trained for their representations to be consistent with the corresponding pseudo-labels. Specifically, for the implementation of both weak augmentations and strong augmentations, we use the same as in subsection 1.

4 Qualitative Examples

In the Main paper, we provide qualitative examples from ShapeNet and Pix3D dataset. Here we have included some more samples from the two datasets to show the superiority of our methods over the competing baseline methods. Fig. 1 and Fig. 2 contain the example of single-view 3D reconstruction for ShapeNet and Pix3D with 10% labelling ratios, respectively.

5 Additional Experiments

Ablation of Prototype Intuitively, increasing the number of prototypes may help learn more powerful shape priors, but this increases the computational cost of attention module. In our experiments we find that when gradually increasing the number of prototypes from 3 to 10, the performance improvement of the model is limited. Concretely, in the 1% labeling-ratio of ShapeNet setting, the mIoU varies from 46.83% to 47.22% with different number of prototypes, and thus we set the number of prototypes to 3 for the efficiency.

Table 1. The comparison of 50% labelling ratio (mIoU %).

	Supervised	MeanTeacher	FixMatch	Ours
ShapeNet	61.2	62.3	62.9	64.2
Pix3D	44.8	45.1	45.6	47.5

Experiments of 50% label setting The experimental comparison under 50% label is shown in Table 1. It is worth pointing out that since the amount of available data (labeled plus unlabeled) is fixed, the number of unlabeled data decreases as we increase the ratio of labeled data, so the gain of the model is not as large as 10% setting.

References

- Berthelot, D., Carlini, N., Goodfellow, I., Papernot, N., Oliver, A., Raffel, C.A.: Mixmatch: A holistic approach to semi-supervised learning. In: NeurIPS (2019) 4
- Chang, A.X., Funkhouser, T., Guibas, L., Hanrahan, P., Huang, Q., Li, Z., Savarese, S., Savva, M., Song, S., Su, H., et al.: Shapenet: An information-rich 3d model repository. arXiv preprint arXiv:1512.03012 (2015) 1, 3
- 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: ECCV (2016) 1
- Gkioxari, G., Malik, J., Johnson, J.: Mesh r-cnn. In: ICCV (2019) 1
- He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR (2016) 2
- Pinheiro, P.O., Rostamzadeh, N., Ahn, S.: Domain-adaptive single-view 3d reconstruction. In: ICCV (2019) 2
- Sohn, K., Berthelot, D., Carlini, N., Zhang, Z., Zhang, H., Raffel, C.A., Cubuk, E.D., Kurakin, A., Li, C.L.: Fixmatch: Simplifying semi-supervised learning with consistency and confidence. In: NeurIPS (2020) 4
- 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: CVPR (2018) 1, 3
- Tarvainen, A., Valpola, H.: Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In: NeurIPS (2017) 3
- Xie, H., Yao, H., Sun, X., Zhou, S., Zhang, S.: Pix2vox: Context-aware 3d reconstruction from single and multi-view images. In: ICCV (2019) 1, 3
- Xie, H., Yao, H., Zhang, S., Zhou, S., Sun, W.: Pix2vox++: multi-scale context-aware 3d object reconstruction from single and multiple images. IJCV (2020) 1, 2, 3
- Zhang, H., Cisse, M., Dauphin, Y.N., Lopez-Paz, D.: mixup: Beyond empirical risk minimization. In: ICLR (2018) 4

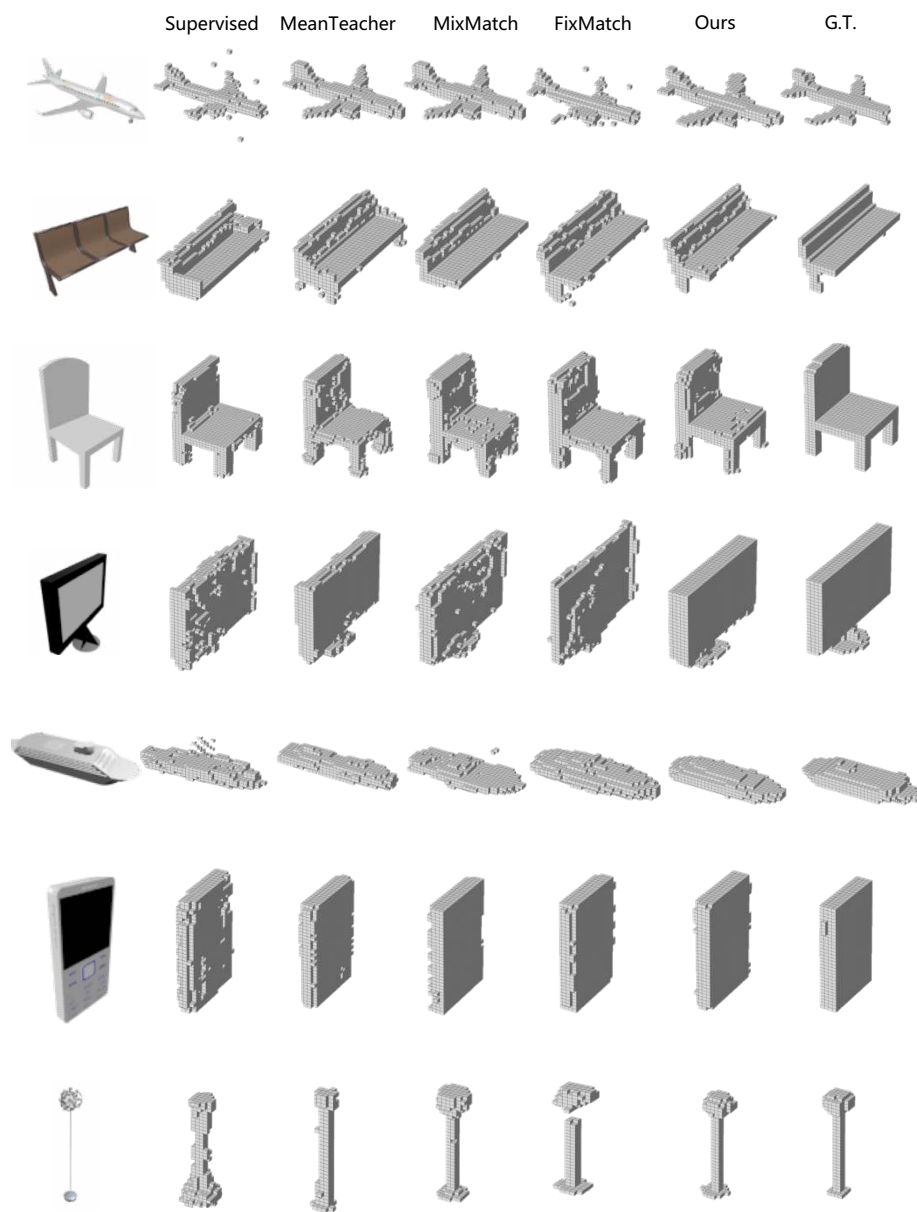


Fig. 1. Examples of single-view 3D Reconstruction on ShapeNet with 10% labels.

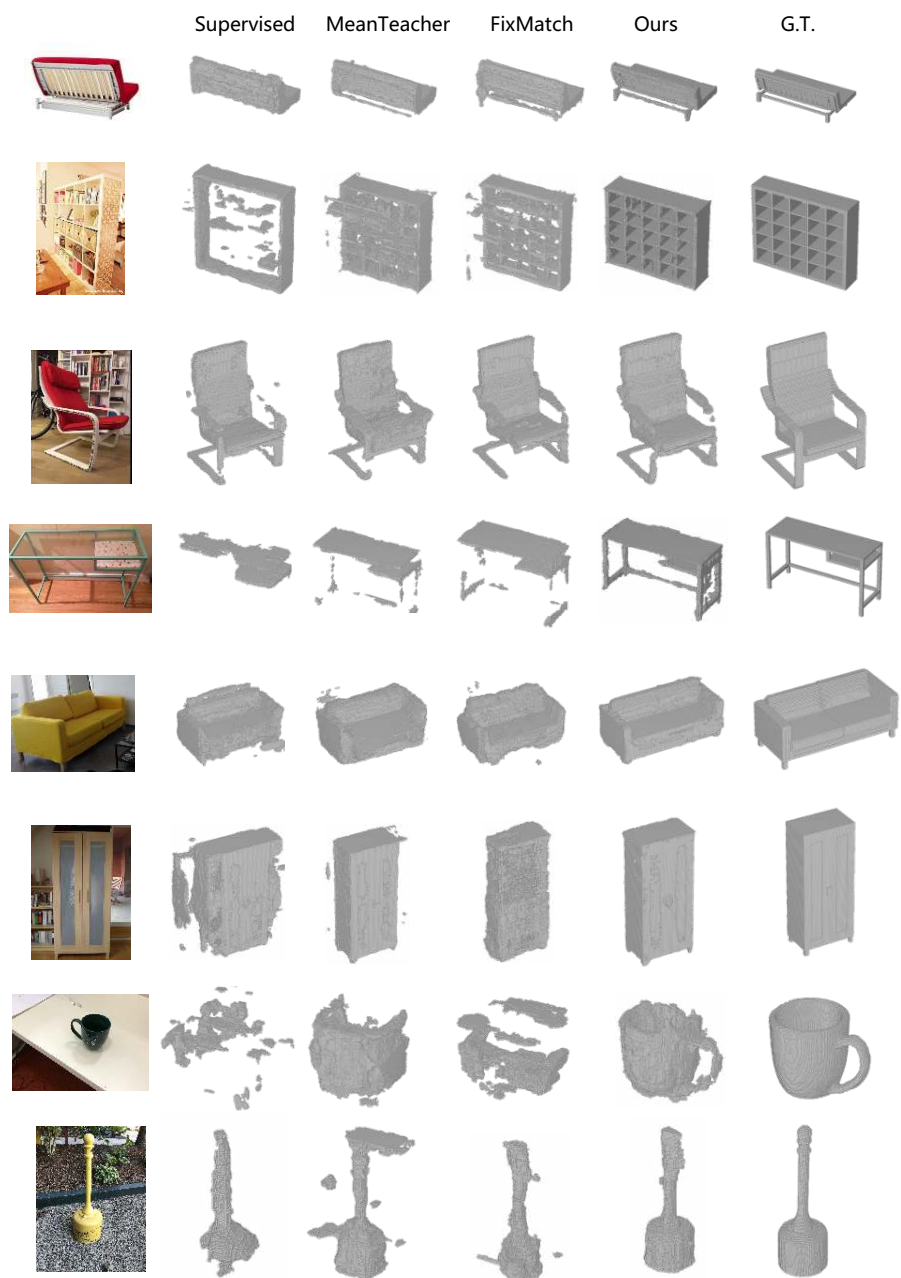


Fig. 2. Examples of single-view 3D Reconstruction on Pix3D with 10% labels.