3D Guided Weakly Supervised Semantic Segmentation

Weixuan Sun[1 2], Jing Zhang[1 2], Nick Barnes[1]

[1] Australian National University, Australia; [2]CSIRO Data61, Australia

Abstract. Pixel-wise clean annotation is necessary for fully-supervised semantic segmentation, which is laborious and expensive to obtain. In this paper, we propose a weakly supervised 2D semantic segmentation model by incorporating sparse bounding box labels with available 3D information, which is much easier to obtain with advanced sensors. We introduce a 2D-3D inference module to generate accurate pixel-wise segment proposal masks. Guided by 3D information, we first generate a point cloud of objects and calculate a per class objectness probability score for each point using projected bounding-boxes. Then we project the point cloud with objectness probabilities back to the 2D images followed by a refinement step to obtain segment proposals, which are treated as pseudo labels to train a semantic segmentation network. Our method works in a recursive manner to gradually refine the above-mentioned segment proposals. We conducted extensive experimental results on the 2D-3D-S dataset where we manually labeled a subset of images with bounding boxes. We show that the proposed method can generate accurate segment proposals when bounding box labels are available on only a small subset of training images. Performance comparison with recent state-of-the-art methods further illustrates the effectiveness of our method.

Keywords: Semantic segmentation, weak supervision, 3D guidance

1 Introduction

Recent work on 2D image semantic segmentation has achieved great progress via adopting deep fully convolutional neural networks (FCN) [1]. The success of these models [2–6] arises from large training datasets with pixel-wise labels, which are laborious and expensive to obtain. For example, the cost of pixel-wise segmentation labeling is 15 times larger than bounding box labeling and 60 times larger than image-level labeling [7].

Unlabeled or weakly-labeled data can be collected in a much faster and cheaper manner, which makes weakly supervised semantic segmentation a promising direction to develop. Multiple types of weak labels have been studied, including image-level labels [9–12], points [13], scribbles [14–16], and bounding boxes [17–21]. Bounding box annotation offers a simple yet intuitive direction, that is relatively inexpensive, while still offering rich semantic regional information.

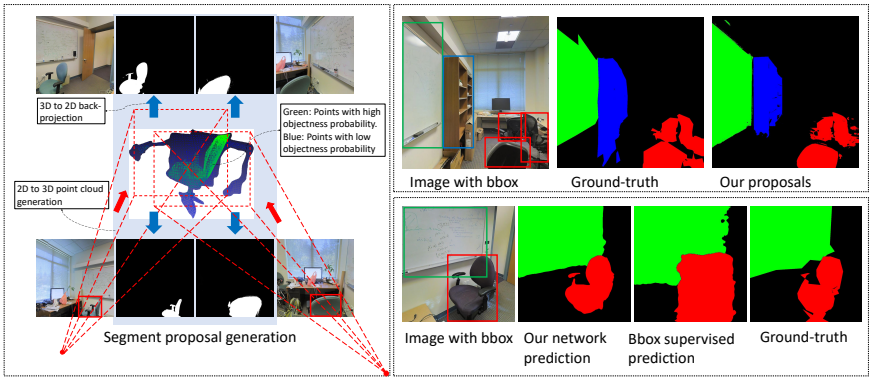

Fig. 1. Left: Our segment proposal generation pipeline. Top right: Example images of our segment proposals compared with ground-truth. Bottom right : Network prediction example supervised with our segment proposals compared with prediction supervised with bounding box masks and ground-truth segmentation map. The sample images are from the 2D-3D-S dataset [8].

Current bounding box based methods [17–21] usually adopt non-learning methods like Conditional Random Fields (CRF) [22], GrabCut [23] or Multiscale Combinatorial Grouping (MCG) [24] to obtain segment proposals, which are then treated as pseudo labels to train semantic segmentation models.

It has been argued that 3D information plays an important role in scene understanding, but most previous semantic segmentation approaches operate only on individual 2D images. With more recent data collection technology and sensors, collection of large scale 3D datasets is no longer a cumbersome process. Not only 2D RGB information but also accurate corresponding 3D information like depth maps, camera trajectories, and point clouds are collected. Especially for indoor scene understanding, datasets like 2D-3D-S [8], SUN3D [25], ScanNet [26] are available. For outdoor autonomous driving there are datasets like Kitti [27], ApolloScape [28] and the Waymo open dataset [29]. With the above-mentioned widely available data, it's natural to raise a question: "*Can we retain comparably good performance while only labeling a few images by using box-level weak supervision together with 3D information?*"

In this paper, we investigate the task of combining bounding box labels with 3D information for weakly supervised semantic segmentation, aiming at reducing annotation cost by leveraging available 3D information. We investigate this by using the Stanford 2D-3D-Semantics dataset (2D-3D-S) [8]. We propose a novel 3D guided weakly supervised semantic segmentation approach, where a small number of images are labeled with bounding boxes and these images have their corresponding 3D data. Our approach can extract segment proposals from bounding boxes on labeled images and creates new segment proposals on unlabeled images of the same object instance. These proposals are then used to train a semantic segmentation network. Further, our approach works in a recursive

manner to gradually refine the above-mentioned segment proposals, leading to improved segmentation results.

The proposed pipeline (2D-3D inference module) is shown in the left of Figure 1, where we use a chair as an example. First, we label the chair from two camera viewpoints and extrude bounding boxes from 2D to 3D space to generate a point cloud of the chair. Then we perform 3D inference to compute an objectness probability for each 3D point, representing the possibility of each point belonging to an object. The objectness probability is computed based on detection frequency across bounding boxes to enhance correct points and suppress noise. As displayed in the left image of Figure 1, green and blue points have high and low objectness probability respectively. We then project from the point cloud with objectness probabilities back to the 2D images to obtain objectness probability masks. Besides the labeled images, we can also back-project the point cloud to new images without labels. During projection, we propose a novel strategy by using depth maps to deal with occlusion. Finally, we refine the objectness probability masks to obtain our final segment proposals. We evaluate our method on the 2D-3D-S dataset [8], and experimental results show that our method considerably outperforms the competing methods using bounding box labels. We summarize our contributions as follows:

- We propose a 3D-guided, 2D weakly supervised semantic segmentation method. Our method leverages information that is widely available from 3D sensors without hand annotation to yield improved semantic segmentation with lower annotation cost.
- We present a novel 2D-3D probabilistic inference algorithm, which combines bounding-box labels and 3D information to simultaneously infer pixel-wise segment proposals for the labeled bounding boxes and unlabeled images.
- Our 3D weakly supervised semantic segmentation model learns an initial classifier from segment proposals, then uses the 2D-3D inference to transductively generate new segment proposals, resulting in further improvements to network performance in an iterative learning manner.
- To the best of our knowledge, it is the first work that uses 3D information to assist weakly supervised semantic segmentation. To evaluate our method we augment the 2D-3D-S dataset [8] with bounding box labels. We demonstrate that our method outperforms competing methods with fewer labeled images.

2 Related Work

We briefly introduce existing fully and weakly supervised semantic segmentation models, and 3D information guided models.

Fully Supervised Semantic Segmentation A series of work has been done based on FCN [1] for fully supervised semantic segmentation. [1, 30, 31] use skip architectures to connect earlier convolutional layers with deconvolutional layers, to reconstruct fine-grained segmentation shapes. The DeepLab series [4, 3, 6] use dilated (atrous) convolution in the encoder, which increases the receptive field to consider more spatial information. In addition, many methods [32, 2, 3,

5, 33–37] improve semantic segmentation performance by adopting context information. [2] proposes pyramid pooling to obtain both global and local context information. An adaptive pyramid context network is proposed in [36] to estimate adaptive context vectors for each local position.

Weakly Supervised Semantic Segmentation A large number of weakly supervised semantic segmentation methods have been proposed to achieve a trade-off between labeling efficiency and network accuracy. They usually take low-cost annotation as a supervision signal, including image-level labels [9–12, 18], scribbles [14–16], points [13], and bounding boxes [17–21]. Current bounding-box based methods extract object segment proposals from bounding boxes, which are then used as a network supervision signal. WSSL [18] proposes an expectation-maximization algorithm with a bias to enable refined estimated segmentation maps throughout training. BoxSup [17] proposes a recursive training procedure, which uses generated proposals as supervision in every iteration. [19] generates segment proposals by incorporating GrabCut [23] and MCG [24]. Most recently, [21] generate segment proposals with dense CRF [22], and proposes box-driven class-wise masking with a filling rate guided adaptive loss in the training procedure. However, none of the above methods adopt 3D information.

3D Information Guided Semantic Segmentation Different from classic 2D RGB semantic segmentation, some work adopts 3D information such as depth maps and point clouds. [38–40] design handcrafted features tailored for RGB with depth information, extracted features are fed into further models. [41, 1] take the depth map as an extra input channel with the RGB images. More recently, [42–45] encode depth maps into three-dimensional HHA (horizontal disparity, height above ground, and angle with gravity). [46] employs 3D convolutions to extract 3D geometry and 3D colour features from point clouds and project them back to 2D images for segmentation. Meanwhile, 3D data is becoming increasingly available from advanced 3D sensors, *e.g.*, [8, 25–29, 47–49] without requiring human intervention. Which offers opportunity to reduce labeling cost by exploiting automatically obtained 3D information. We propose to bring in 3D information to assist proposal generation from bounding boxes and propose a 3D guided weakly supervised semantic segmentation network. As far as we know, this is the first work that combines box-level labels and 3D information for weakly supervised semantic segmentation.

3 Proposed Approach

We propose a 3D information guided bounding-box based weakly supervised semantic segmentation network. Specifically, our method consists of two modules: 1) a segment proposal generation module that adopts the 3D information and bounding box labels, and 2) a semantic segmentation network, which takes 2D images as training data. First, we feed images with box-level labels and their corresponding 3D information into our segment proposal generation framework, which extracts pixel-wise segment proposals from labeled bounding boxes and generates proposals on new images without labels. Then, the segment propos-

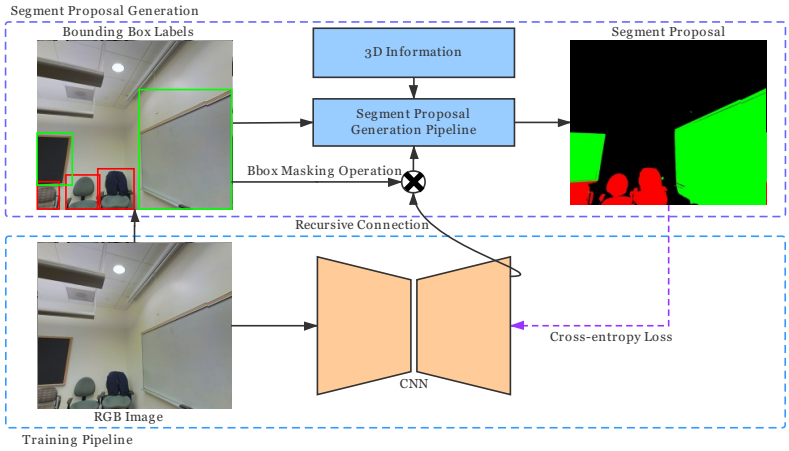

Fig. 2. Pipeline of the proposed method. We first hand-label a subset of all RGB images with bounding box, which is then fed into a 3D semantic projection module to generate segment proposals assisted by 3D information. The generated segment proposals are used as supervision signal for our semantic segmentation network. Meanwhile, the network predictions can also be fed back into the 3D semantic projection module after the bounding box masking operation in a recursive manner. The details of segment proposal generation pipeline are shown in Figure 3.

als are fed into the training pipeline as a supervision signal to train a semantic segmentation network. The network predictions are then fed back into the segment proposal generation module, which generates new segment proposals in a recursive manner. The entire procedure is shown in Figure 2.

Considering a collection of images $X = \{X_1, ..., X_i, ..., X_N\}$, each image has corresponding 3D information, *i.e.*, camera parameters $M_i = (R_i, \tilde{C}_i, f_i)$ and a depth map X_{depth}, where R_i is camera rotation matrix, $\tilde{C}_i$ is camera position and f_i is focal length. In our method, we assume that only a subset of all images are labelled with bounding boxes, where $X_b \subset X$. During labeling, we label images from different camera viewpoints. Then we feed the labels into our segment proposal generation framework with their corresponding camera parameters and depth maps. We present our segment proposal generation module as a function $S = F(X, X_b, M_i, X_{depth})$, where S is the collection of all segment proposals $S = \{S_1, ..., S_i, ..., S_N\}$. Then the segment proposals S are used supervise the semantic segmentation network: $L = L_s(X_p, S)$, where L_s denotes segmentation loss and Y_p denotes network prediction. In addition, in the recursive process, images labeled with bounding boxes X_b can be replaced with the network predictions X_p from the previous iteration, which is displayed as a recursive connection in Figure 2. The recursive process is introduced in detail in Sec. 3.2.

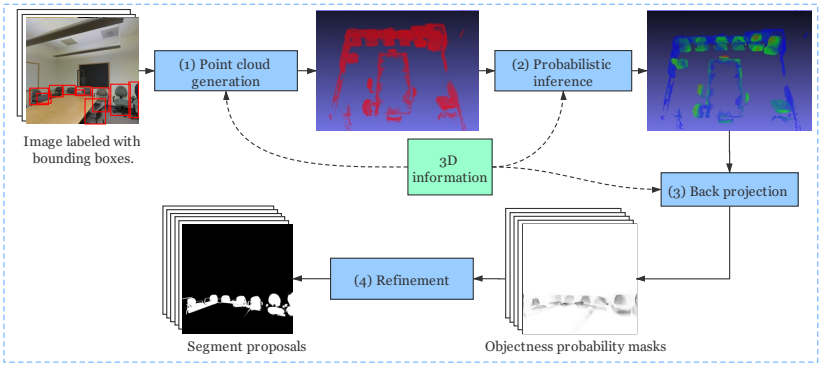

Fig. 3. Our segment proposal generation pipeline. Taking the class of chair as an example, we first project the point cloud for chairs in this room from bounding box labels. Then, we calculate each point's objectness score. Finally, we back-project the point cloud with probabilities to 2D images followed by refinement to get final segment proposals. Note that we can obtain segment proposals on extra images, which means we only need to label a small portion of images.

3.1 Segment Proposal Generation:

Given a set of images X with corresponding 3D information, and annotated bounding boxes for a subset X_b of the images, our framework can learn semantic masks S (segment proposals) for all images of a room. As shown in Figure 3, our approach uses a sequence consisting of four components: (1) bounding boxes to 3D projection to generate point clouds; (2) 3D probabilistic inference to accentuate correct points and diminish noise; (3) point-wise 3D to 2D projection to generate scattered objectness probability masks; and (4) mask refinement to get final segment proposals. We introduce each component in the following.

Point Cloud Generation from Bounding Boxes This process aggregates label information from different camera viewpoints into a globally consistent 3D space[1]. Concretely, given an image X_i, pixels inside bounding boxes are represented as x^i. For one single pixel at position j inside the bounding boxes $x^i_j \in x^i$, we project it into 3D by:

$$P^i_j = [R_i \mid -R_i\tilde{C}_i]^{-1} K_i^{-1} * x^i_j * d^i_j, \quad \text{where} \quad K_i^{-1} = \begin{bmatrix} \frac{1}{f^i} & & -\frac{p^i_x}{f^i} \\ & \frac{1}{f^i} & -\frac{p^i_y}{f^i} \\ & & 1 \end{bmatrix}. \quad (1)$$

[1] 3D information provided by the 2D-3D-S dataset [8] is determinate, and SLAM reconstruction is mature, so high quality 3D information is assumed. Our method is not based on SLAM and every point is projected independently so we don't need to handle accumulated errors.

We follow the finite camera projection model [50] to project the pixel x_j^i into 3D space, K_i^{-1} denotes inverse camera matrix which projects pixels from the 2D image to camera coordinate, f^i is focal length and $[p_x^i, p_y^i]$ is principal point. $[R_i \mid -R_i\tilde{C}_i]^{-1}$ transforms points from camera coordinates to world coordinates, where R_i is a 3×3 rotation matrix representing the orientation of the camera coordinates, and $\tilde{C}_i$ denotes the position of the camera in world coordinates. d_j^i denotes depth information at position j of image i. Then we combine the projected 3D point clouds P^i from different camera views together in world coordinate to obtain a class-specific point cloud. We perform projection for each labeled bounding box, label information from different directions and classes is fused into a single 3D point cloud. By adopting depth maps, only points nearest to the camera are projected, which ensure accurate object shapes and occluded points are ignored. The class of every point is decided by the class of the projecting bounding box, so the point clouds are semantically classified. As shown in Figure 3, the red point cloud displays all chairs in the environment.

Point Cloud Probabilistic Inference Bounding boxes consist of object and background regions. When we project pixels in the bounding boxes into 3D space, the background regions are also projected as background noise. As shown by the red point clouds of Figure 3, wall and table are also projected and wrongly categorized as chair. In order to distinguish points that belong to objects or background, we take advantage of our multiple views and 3D information.

We propose a novel method to sum objectness confidence across multiple views, which emphasizes the correct points and weakens the irrelevant points (background noise). We quantitatively present every point's correctness with a score called the objectness score. Inspired by [51], given that the projection matrix and depth map are known, we can get a 3D bounding frustum from a 2D bounding box, which is the "visible space" of that bounding box. All points inside this 3D bounding frustum can be projected back into the bounding box on 2D image. Thus, we make an assumption that, for some camera viewpoints, if a point projected from one camera viewpoint can also be "seen" by other camera viewpoints, *i.e.*, the point can be projected back into the bounding boxes of other images, the objectness score of the point is higher. In this case, with multiple bounding boxes on a single object, the 3D points with a higher objectness score are more likely to belong to this object than to the background. Specifically, for a 3D point P_j of class c, its objectness score O_j is defined as:

$$O_j(P_j \mid B_c) = \sum_{k=1}^{K} F_o(P_j, b_k), \tag{2}$$

where B_c denotes bounding boxes of class c in this room and $F_o(P_j, b_k)$ is:

$$F_o(P_j, b_k) = \begin{cases} 1 & \text{if back-projected } P_j \text{ is inside } b_k \\ 0 & \text{otherwise} \end{cases}. \tag{3}$$

O_j in Eq. 2 indicates the frequency that the point P_j is projected back to bounding boxes across all the images from different viewpoints. We implement

the above method for each class independently, then normalize objectness scores for each class to obtain an objectness probability $p(O_j)$: $p(O_j) = \frac{O_j}{max(O^c)}$, where O^c denotes the collection of objectness scores of all points that belong to class c. Points with higher objectness scores have more confidence to belong to objects. By doing so, label information from different camera viewpoints are aggregated on points to compose their objectness probability, and reveal objects' shapes. As shown in the top right figure of Figure 3, after probabilistic inference, the chairs stand out from background noise, while wall and tables are suppressed.

Segment Proposal Generation by Point Cloud Back-projection In this stage, we apply 3D to 2D back-projection to generate prototype objectness score masks. Specifically, given a single point P_j in the point cloud, we project it to a image at camera viewpoint i [50]:

$$x_j^i = K_i[R_i \mid -R_i\tilde{C}_i]P_j, \quad \text{where} \quad K_i = \begin{bmatrix} f_i & & p_x^i \\ & f_i & p_y^i \\ & & 1 \end{bmatrix}. \tag{4}$$

We project all points with their objectness probabilities and semantic labels onto 2D images. All 3D points are in the same world coordinate system and can be back-projected to 2D images at any camera viewpoints, no matter whether the images are labeled or not. Therefore, we only need to label a small portion of the images, which can significantly alleviate annotation cost. In addition, since the point cloud is sparse in 3D space, we get 2D masks with scattered points and every point represents normalized objectness probability, which are named objectness probability masks, as shown in the third column of Figure 5.

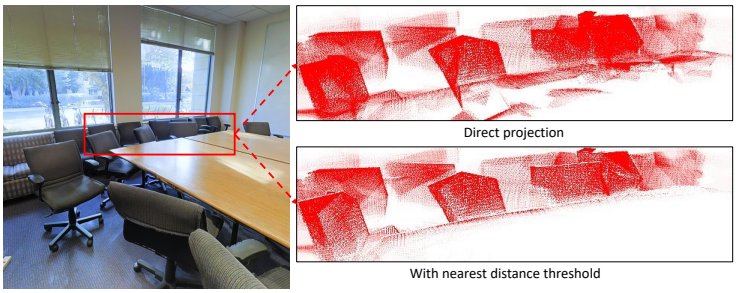

Fig. 4. Illustration of the proposed nearest distance threshold. As shown in the left RGB image, some parts of the chairs are occluded under the table. With direct projection, the occluded parts of the chairs are still wrongly projected. After applying our nearest distance threshold, occluded areas are properly ignored.

Nearest Distance Threshold Occlusion may occur during the back projection, *i.e.*, when objects are overlapped facing a camera, points belong to both visible objects and occluded objects are projected to the same region. To address this issue, we propose a nearest distance threshold. by using the depth

map. Depth represents the nearest surface to the camera, all 3D points behind the surface are occluded which should not be projected to 2D. Concretely, for a 3D point P_j, we calculate its distance to the camera as z_j. Then we project the 3D point back to 2D at camera viewpoint i at position x_j^i and obtain the depth threshold d_j^i at that position. Only points with $z_j <= d_j^i$ can be projected to generate objectness score masks. Sample results are shown in Figure 4.

Segment Proposals Refinement In this section, we propose a method to refine scattered objectness probability masks into segment proposals. We take the chair class as an example and display results in Figure 5. As shown in the third column of Figure 5, the objectness probability mask displays accurate object localization and objectness probability. However, the projected masks are sparse and cannot directly be used as a supervision. To address this issue, we adopt a morphological operation followed by a fully-connected CRF [22] to recover dense segment proposals from the scattered masks.

First, we follow [52] to binarize the projected objectness probability mask, then apply an image close operation. Then, we follow [4] to adopt a fully connected CRF [22] to refine local boundary areas of our segment proposals. Referring to the fourth column of Figure 5, it shows the recovery of image object boundaries based on 2D features, resulting in accurate segment proposals.

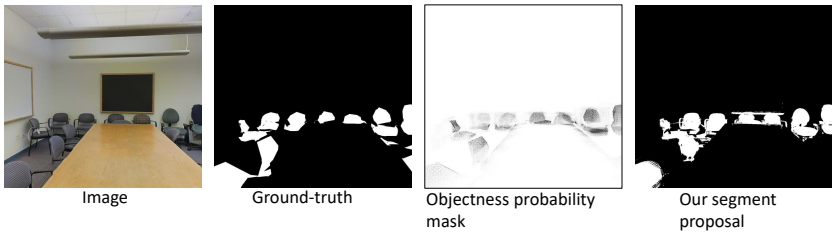

Fig. 5. Examples of the segment proposals for the class of chair, where we reverse the grayscale of objectness probability masks for better visualization.

3.2 Recursive Procedure

We observe that the generated segment proposals capture the object shape significantly better than bounding boxes, which inspires us to adopt a recursive training procedure of transductive segmentation.

First, we generate segment proposals with bounding box labels. Then, after fully training the segmentation network with the segment proposals, the segment predictions are fed back into our pipeline to generate new segment proposals, where the input is segmentation masks instead of bounding boxes. New segment proposals are then used as supervision for the next iteration of training. By cycles of segment proposal generation, learning a semantic segmentation network, and

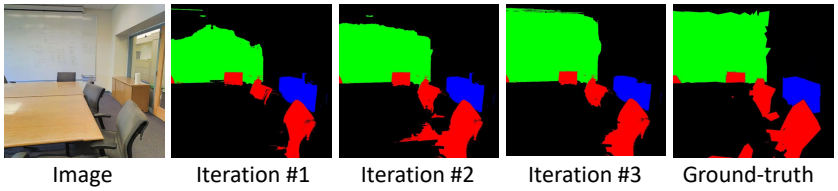

Fig. 6. Recursive updating of the segment proposals, which are progressively refined in each iteration, and then treated as supervision for the next iteration.

using the learned network to find improved masks on previously seen images, we iteratively improve segment proposals and segmentation network.

Moreover, we use the bounding box labels to constrain our network predictions, *i.e.*, bounding box labels ensure that regions outside do not contain any object. So we employ bounding box masks to apply binary masking on our network predictions, which effectively removes false positive areas: $\Phi_p = X_b \otimes X_p$, where $\otimes$ means spatial-wise masking, X_b denotes masks generated by bounding boxes, X_p denotes original network predictions. We feed the masked network predictions Φ_p into our segment proposal generation module as shown in the recursive connection in Figure 2.

3.3 Data Annotation

Currently, there exists no dataset available with multi-view camera parameters, depth data, and bounding box labels. Hence, we augment a subset of dataset 2D-3D-S [8] with hand labeled bounding boxes. 2D-3D-S [8] is an indoor dataset with multiple modalities from 2D, 2.5D and 3D domains, with instance-level semantic and geometric annotations. The dataset is collected in 6 large-scale indoor areas that originate from 3 buildings. The dataset provides a corresponding depth map, camera parameters for each 2D image, we adopt them as 3D information. They also provide 2D segmentation ground-truth projected from semantically labeled 3D mesh model. We pick four common indoor object classes (with clutter as the fifth class) to validate our method, including chair, bookcase, sofa and board as they are well defined and suitable for bounding box labels.

In this paper, we use data from area 1 to validate our method. Since there is no dataset contains both multi-view information and bounding box labels. We manually label bounding boxes on a subset of 1822 images and obtain segment proposals on all training images by adopting our proposed algorithm. Here we introduce the process that how we select a subset of images. First, since 2D images are separated by rooms, in order to obtain more object instances, we choose rooms where our objects are present and correctly labeled in the dataset. Second, in each room, there are several camera locations, we ensure that the labeled images include images with views of the objects from different camera

viewpoints to support our assumption of images over a wide baseline. Then, the generated segment proposals are used as a supervision signal during training. It is worth noting that the annotation signal is entirely derived from the box-level labels. Moreover, our bounding box labels can be used to train a prior 2D detector, *i.e.*, we can automate bounding boxes generation on more data, which further extends our method to a larger scale of data.

4 Experiments

4.1 Setup

We evaluate the proposed method on the 2D-3D-S [8] dataset. The dataset and annotation details are introduced in Sec. 3.3. We adopt the publicly available DeepLabV3+ [53] model as our backbone network. DeepLabv3+ is a recent state-of-the-art segmentation pipeline that uses a ResNet head [54], pre-trained on ImageNet [55]. We keep network structure unchanged, and train models under different supervision conditions to validate the effectiveness of our method. During training, the initial learning rate is 0.01 and is decreased by a factor of 10 after every 10 epochs. SGD is used as our optimizer with momentum of 0.9 and weight decay of 0.0001. All the training data are augmented by random cropping and horizontal flipping. We do not adopt a fully-connected CRF for post-processing of our network predictions. Results reported in Table 1, Table 2 and Figure 8 are all from the first iteration without our recursive method. The evaluation performance is measured in terms of pixel intersection-over-union (mIoU). All experiments were conducted using PyTorch.

4.2 Comparison with Other Methods

In Table 1, comparisons are made to evaluate the impact of different levels of supervision. In our own method, we label bounding boxes on 1822 images and use our approach to obtain segment proposals on 4028 images as our training set, the performances are evaluated on a randomly selected validation set without overlapping without the training set. As a naive baseline, *Bounding boxes 1822*, we fill the bounding boxes of the labeled images as masks and use the filled masks as supervision. We form a second baseline, *CRF refined boxes 1822*, by directly applying a CRF on the bounding boxes to generate segment proposals. Moreover, we adopt GrabCut [23] to directly extract segment proposals from bounding boxes as *GrabCut 1822*. Then we evaluate performance in fully-supervised mode where the ground-truth is provided by the 2D-3D-S [8]. We report performances of the models supervised with both 1822 pixel-wise ground-truth, *pixel-wise 1882* and full 4028 pixel-wise ground-truth, *pixel-wise 4028*. We also report the performance in semi-supervised mode. We randomly select 400 images from the 4028 training set images, replace supervision with the ground-truth provided by [8]. which is called *semi 1822+400* in Table 1.

In addition, we compare with two state-of-the-art methods, *i.e.*, SDI [19], WSSL [18]. For bounding box based weakly supervised semantic segmentation,

the state-of-the-art methods are: [21], BoxSup [17], SDI [19] and WSSL [18]. However, [21, 17, 19] did not release the official implementation code. Further, the released code for [18] was based on another deep framework. We re-implement SDI [19] and WSSL [18] with Pytorch as their performance is still competitive with SOTA and have clear and explicit implementation details. Our implementations will be made publicly available.

Table 1 shows the results of our segment proposals compared with other methods. The bounding-box based baseline achieves 61.78 of mIoU while CRF refined bounding boxes improve the performance to 62.69. To explore the upper-bound on weakly supervised performance, we include results with a fully supervised model, trained using 1822 and 4028 pixel-wise ground-truth images respectively, the scores are 75.30 and 79.38. Our method achieves 72.27, which outperforms all the compared bounding-box based methods with a clear margin and is approaching the 1822 pixel-wise supervised baseline. This validates that our proposed method is effective. By labeling a subset of the dataset, we can extract accurate segment proposals on labeled and unlabeled images. Finally, in semi-supervised mode, we achieve 73.36 mIoU through replacing segment proposals of 400 images (10% of the training set) with pixel-level ground truth, where the performance is comparable with fully supervised models. The performance of our semi-supervised model indicates that we can achieve even better performance if additional supervision information is provided.

4.3 Experiments on Data from Unseen Areas

In our procedure, we label a subset of all images and get segment proposals for all those images by adopting 3D information. However, images from the same room may view the same object instances across the training and validation sets. To validate the effectiveness of our proposed method, we randomly select 2000

Modes	Method	Sofa	Board	Chair	Bookcase	Clutter	mIoU
Full	Pixel-wise 4028	62.77	89.57	70.58	77.77	96.23	79.38
	Pixel-wise 1822	54.44	85.29	66.78	74.64	95.37	75.30
Box	Bounding boxes 1822	34.23	71.23	45.30	67.65	90.46	61.78
	CRF refined boxes 1822	31.47	73.13	49.09	68.26	91.48	62.69
	GrabCut 1822	41.38	79.79	56.91	70.76	93.89	68.55
	WSSL [18] 1822	55.52	75.58	53.43	67.14	93.95	69.06
	SDI [19] $M \cap G+$ 1822	47.28	81.10	56.47	66.94	93.74	69.11
	Ours 1822	63.24	81.23	60.31	62.10	94.45	**72.27**
Semi	Semi 1822+400	61.98	80.33	62.77	67.09	94.64	73.36

Table 1. Comparison of performances under different supervision conditions. The number after the method name means the number of images with human annotations that were used in this setting. $M \cap G+$: using the masks where both MCG and GrabCut agree. In bold is the best performing of the bounding box supervised methods. Our method outperforms competing box supervised methods and is midway to the fully supervised methods.

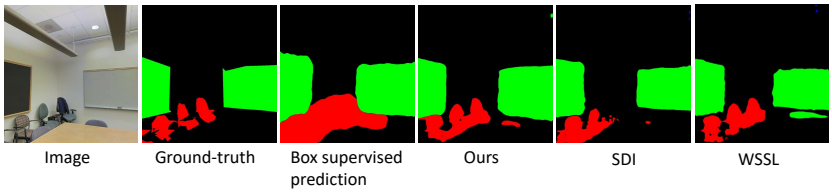

Fig. 7. Examples of semantic segmentation prediction results.

Modes	Supervision	mIoU
Full	Pixel-wise 1822	63.62
Box	Bounding boxes 1822	53.57
	CRF refined boxes 1822	54.82
	Ours 1822	**57.61**
Semi	Semi 1822+1000	59.24

Table 2. Comparison of performances on the test set.

Supervision	Iteration 1	Iteration 2	Iteration 3
Ours	57.61	59.12	59.83

Table 3. Evaluate effectiveness of the recursive training method.

images from new areas to assemble a test set. These new areas are "unseen", which means none of the images in these areas are labeled nor seen in the training or validation. We test our trained model on this test set without fine-tuning. As shown in Table 2, our method outperforms the model trained with only bounding box masks, and achieves comparable performance to the fully supervised model. Thus, it validates that our method captures precise object information, and provides similar general class-wise features as the pixel-wise ground-truth.

4.4 Recursive Training Performance

As introduced in Sec. 3.2, we may apply our proposed method in a recursive manner. In the recursive procedure, the network predictions are fed into our segment proposal generation pipeline to generate new segment proposals. Although we achieve good results in the first iteration, the recursive training process transductively refines the new segment proposals and improves our network progressively. As shown in Table 3, we evaluate performance on the test set, the performance in every iteration is gradually improved.

4.5 Ablation: Effect of Ratio of Annotated Data

Our proposed method labels only a small portion of the dataset and obtains segment proposals on all images, which drastically decreases the cost of annotation.

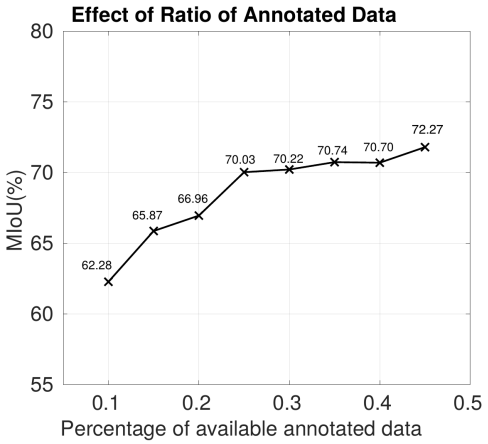

Fig. 8. Performance of our method when we randomly select different percentages of available annotated data. Reasonable segmentation is possible when just 10 percent of the training set is annotated. We can observe clear improvement when more annotation is provided, trailing off above 25 percent.

In this section, we investigate how the percentage of available annotated data affects performance, and attempt to achieve a balance between annotation cost and network performance. 1822 labeled images make up 45% of 4028 training images. We randomly select images from these to get different ratios of available labeled images. Selected labels are then fed into the same pipeline to generate segment proposals on the training set and report results in Figure 8. As shown, more annotation data leads to manifest performance improvements which indicates the effectiveness of our method. When trained using only 25% of training images, we achieve 70.03 mIoU which already outperforms all competing methods (trained on 45%). Moreover, our method still gets a reasonably good results when only 10% of the training data is labeled with bounding boxes.

5 Conclusion

In this paper, we propose a novel 3D weakly supervised semantic segmentation approach, which incorporates box-level labels with corresponding 3D information. By only labeling a small number of images with bounding boxes, our approach extracts segment proposals on labeled and unlabeled images. Then we use the obtained segment proposals to train a semantic segmentation model. Moreover, our method can work in a recursive manner, which further refines our segment proposals. Our proposed method achieves competitive semantic segmentation results with less annotation effort. We evaluate the proposed method on the 2D-3D-S [8] dataset, extensive experimental results show that our proposed method is effective. Our annotations and source code will be made publicly available for future research endeavors in this field.

References

1. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proc. IEEE Conf. Comp. Vis. Patt. Recogn. (2015) 3431–3440
2. Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J.: Pyramid scene parsing network. In: Proc. IEEE Conf. Comp. Vis. Patt. Recogn. (2017) 2881–2890
3. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. IEEE Trans. Pattern Anal. Mach. Intell. **40** (2017) 834–848
4. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Semantic image segmentation with deep convolutional nets and fully connected crfs. ArXiv e-prints (2014)
5. Zhao, H., Zhang, Y., Liu, S., Shi, J., Change Loy, C., Lin, D., Jia, J.: Psanet: Point-wise spatial attention network for scene parsing. In: Proc. Eur. Conf. Comp. Vis. (2018) 267–283
6. Chen, L.C., Papandreou, G., Schroff, F., Adam, H.: Rethinking atrous convolution for semantic image segmentation. ArXiv e-prints (2017)
7. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: Proc. Eur. Conf. Comp. Vis. (2014) 740–755
8. Armeni, I., Sax, A., Zamir, A.R., Savarese, S.: Joint 2D-3D-Semantic Data for Indoor Scene Understanding. ArXiv e-prints (2017)
9. Huang, Z., Wang, X., Wang, J., Liu, W., Wang, J.: Weakly-supervised semantic segmentation network with deep seeded region growing. In: Proc. IEEE Conf. Comp. Vis. Patt. Recogn. (2018) 7014–7023
10. Wei, Y., Xiao, H., Shi, H., Jie, Z., Feng, J., Huang, T.S.: Revisiting dilated convolution: A simple approach for weakly-and semi-supervised semantic segmentation. In: Proc. IEEE Conf. Comp. Vis. Patt. Recogn. (2018) 7268–7277
11. Ahn, J., Kwak, S.: Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation. In: Proc. IEEE Conf. Comp. Vis. Patt. Recogn. (2018) 4981–4990
12. Fan, J., Zhang, Z., Tan, T.: Cian: Cross-image affinity net for weakly supervised semantic segmentation. ArXiv e-prints (2018)
13. Bearman, A., Russakovsky, O., Ferrari, V., Fei-Fei, L.: What's the point: Semantic segmentation with point supervision. In: Proc. Eur. Conf. Comp. Vis. (2016) 549–565
14. Vernaza, P., Chandraker, M.: Learning random-walk label propagation for weakly-supervised semantic segmentation. In: Proc. IEEE Conf. Comp. Vis. Patt. Recogn. (2017) 7158–7166
15. Lin, D., Dai, J., Jia, J., He, K., Sun, J.: Scribblesup: Scribble-supervised convolutional networks for semantic segmentation. In: Proc. IEEE Conf. Comp. Vis. Patt. Recogn. (2016) 3159–3167
16. Tang, M., Perazzi, F., Djelouah, A., Ben Ayed, I., Schroers, C., Boykov, Y.: On regularized losses for weakly-supervised cnn segmentation. In: Proc. Eur. Conf. Comp. Vis. (2018) 507–522
17. Dai, J., He, K., Sun, J.: Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation. In: Proc. IEEE Int. Conf. Comp. Vis. (2015) 1635–1643

18. Papandreou, G., Chen, L.C., Murphy, K.P., Yuille, A.L.: Weakly-and semi-supervised learning of a deep convolutional network for semantic image segmentation. In: Proc. IEEE Int. Conf. Comp. Vis. (2015) 1742–1750
19. Khoreva, A., Benenson, R., Hosang, J., Hein, M., Schiele, B.: Simple does it: Weakly supervised instance and semantic segmentation. In: Proc. IEEE Conf. Comp. Vis. Patt. Recogn. (2017) 876–885
20. Li, Q., Arnab, A., Torr, P.H.: Weakly-and semi-supervised panoptic segmentation. In: Proc. Eur. Conf. Comp. Vis. (2018) 102–118
21. Song, C., Huang, Y., Ouyang, W., Wang, L.: Box-driven class-wise region masking and filling rate guided loss for weakly supervised semantic segmentation. In: Proc. IEEE Conf. Comp. Vis. Patt. Recogn. (2019) 3136–3145
22. Krähenbühl, P., Koltun, V.: Efficient inference in fully connected crfs with gaussian edge potentials. In: Proc. Adv. Neural Inf. Process. Syst. (2011) 109–117
23. Rother, C., Kolmogorov, V., Blake, A.: Grabcut: Interactive foreground extraction using iterated graph cuts. In: ACM transactions on graphics (TOG). Volume 23. (2004) 309–314
24. Pont-Tuset, J., Arbelaez, P., Barron, J.T., Marques, F., Malik, J.: Multiscale combinatorial grouping for image segmentation and object proposal generation. IEEE Trans. Pattern Anal. Mach. Intell. **39** (2016) 128–140
25. Xiao, J., Owens, A., Torralba, A.: Sun3d: A database of big spaces reconstructed using sfm and object labels. In: Proc. IEEE Int. Conf. Comp. Vis. (2013) 1625–1632
26. Dai, A., Chang, A.X., Savva, M., Halber, M., Funkhouser, T., Nießner, M.: Scannet: Richly-annotated 3d reconstructions of indoor scenes. In: Proc. IEEE Conf. Comp. Vis. Patt. Recogn. (2017) 5828–5839
27. Geiger, A., Lenz, P., Stiller, C., Urtasun, R.: Vision meets robotics: The kitti dataset. International Journal of Robotics Research (IJRR) (2013)
28. Huang, X., Cheng, X., Geng, Q., Cao, B., Zhou, D., Wang, P., Lin, Y., Yang, R.: The apolloscape dataset for autonomous driving. In: Proc. IEEE Conf. Comp. Vis. Patt. Recogn. Workshops. (2018) 954–960
29. Sun, P., Kretzschmar, H., Dotiwalla, X., Chouard, A., Patnaik, V., Tsui, P., Guo, J., Zhou, Y., Chai, Y., Caine, B., et al.: Scalability in perception for autonomous driving: Waymo open dataset. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. (2020) 2446–2454
30. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical image computing and computer-assisted intervention. (2015) 234–241
31. Badrinarayanan, V., Kendall, A., Cipolla, R.: Segnet: A deep convolutional encoder-decoder architecture for image segmentation. IEEE Trans. Pattern Anal. Mach. Intell. **39** (2017) 2481–2495
32. Liu, W., Rabinovich, A., Berg, A.C.: Parsenet: Looking wider to see better. ArXiv e-prints (2015)
33. Yuan, Y., Wang, J.: Ocnet: Object context network for scene parsing. ArXiv e-prints (2018)
34. Zhang, H., Zhang, H., Wang, C., Xie, J.: Co-occurrent features in semantic segmentation. In: Proc. IEEE Conf. Comp. Vis. Patt. Recogn. (2019) 548–557
35. Zhou, Y., Sun, X., Zha, Z.J., Zeng, W.: Context-reinforced semantic segmentation. In: Proc. IEEE Conf. Comp. Vis. Patt. Recogn. (2019) 4046–4055
36. He, J., Deng, Z., Zhou, L., Wang, Y., Qiao, Y.: Adaptive pyramid context network for semantic segmentation. In: Proc. IEEE Conf. Comp. Vis. Patt. Recogn. (2019) 7519–7528

37. Fu, J., Liu, J., Tian, H., Li, Y., Bao, Y., Fang, Z., Lu, H.: Dual attention network for scene segmentation. In: Proc. IEEE Conf. Comp. Vis. Patt. Recogn. (2019) 3146–3154
38. Ren, X., Bo, L., Fox, D.: Rgb-(d) scene labeling: Features and algorithms. In: Proc. IEEE Conf. Comp. Vis. Patt. Recogn. (2012) 2759–2766
39. Gupta, S., Arbelaez, P., Malik, J.: Perceptual organization and recognition of indoor scenes from rgb-d images. In: Proc. IEEE Conf. Comp. Vis. Patt. Recogn. (2013) 564–571
40. Silberman, N., Hoiem, D., Kohli, P., Fergus, R.: Indoor segmentation and support inference from rgbd images. In: Proc. Eur. Conf. Comp. Vis. (2012) 746–760
41. Eigen, D., Fergus, R.: Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In: Proc. IEEE Int. Conf. Comp. Vis. (2015) 2650–2658
42. Gupta, S., Girshick, R., Arbeláez, P., Malik, J.: Learning rich features from rgb-d images for object detection and segmentation. In: Proc. Eur. Conf. Comp. Vis. (2014) 345–360
43. Qi, X., Liao, R., Jia, J., Fidler, S., Urtasun, R.: 3d graph neural networks for rgbd semantic segmentation. In: Proc. IEEE Int. Conf. Comp. Vis. (2017) 5199–5208
44. Park, S.J., Hong, K.S., Lee, S.: Rdfnet: Rgb-d multi-level residual feature fusion for indoor semantic segmentation. In: Proc. IEEE Int. Conf. Comp. Vis. (2017) 4980–4989
45. Wang, W., Neumann, U.: Depth-aware cnn for rgb-d segmentation. In: Proc. Eur. Conf. Comp. Vis. (2018) 135–150
46. Hou, J., Dai, A., Nießner, M.: 3d-sis: 3d semantic instance segmentation of rgb-d scans. In: Proc. IEEE Conf. Comp. Vis. Patt. Recogn. (2019) 4421–4430
47. Vechersky, P., Cox, M., Borges, P., Lowe, T.: Colourising point clouds using independent cameras. IEEE Robotics and Automation Letters **3** (2018) 3575–3582
48. Chen, D.Z., Chang, A.X., Nießner, M.: Scanrefer: 3d object localization in rgb-d scans using natural language. arXiv preprint arXiv:1912.08830 (2019)
49. Chang, A., Dai, A., Funkhouser, T., Halber, M., Niessner, M., Savva, M., Song, S., Zeng, A., Zhang, Y.: Matterport3d: Learning from rgb-d data in indoor environments. International Conference on 3D Vision (3DV) (2017)
50. Hartley, R., Zisserman, A.: Multiple view geometry in computer vision. Cambridge University Press (2003)
51. Qi, C.R., Liu, W., Wu, C., Su, H., Guibas, L.J.: Frustum pointnets for 3d object detection from rgb-d data. In: Proc. IEEE Conf. Comp. Vis. Patt. Recogn. (2018) 918–927
52. Otsu, N.: A threshold selection method from gray-level histograms. IEEE Transactions on Systems, Man, and Cybernetics **9** (1979) 62–66
53. Chen, L.C., Zhu, Y., Papandreou, G., Schroff, F., Adam, H.: Encoder-decoder with atrous separable convolution for semantic image segmentation. In: Proc. Eur. Conf. Comp. Vis. (2018)
54. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proc. IEEE Conf. Comp. Vis. Patt. Recogn. (2016)
55. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: Proc. IEEE Conf. Comp. Vis. Patt. Recogn. (2009) 248–255