METHODS
published: 28 April 2020
doi: 10.3389/fnins.2020.00350

3D-BoxSup: Positive-Unlabeled Learning of Brain Tumor Segmentation Networks From 3D Bounding Boxes

Yanwu Xu[1], *Mingming Gong*[2], *Junxiang Chen*[1], *Ziye Chen*[3] and *Kayhan Batmanghelich*[1*]

[1] Department of Biomedical Informatics, University of Pittsburgh, Pittsburgh, PA, United States, [2] School of Mathematics and Statistics, The University of Melbourne, Melbourne, VIC, Australia, [3] School of Compuer Science, Wuhan University, Wuhan, China

Accurate segmentation is an essential task when working with medical images. Recently, deep convolutional neural networks achieved a state-of-the-art performance for many segmentation benchmarks. Regardless of the network architecture, the deep learning-based segmentation methods view the segmentation problem as a supervised task that requires a relatively large number of annotated images. Acquiring a large number of annotated medical images is time consuming, and high-quality segmented images (i.e., strong labels) crafted by human experts are expensive. In this paper, we have proposed a method that achieves competitive accuracy from a "weakly annotated" image where the weak annotation is obtained via a 3D bounding box denoting an object of interest. Our method, called "3D-BoxSup," employs a positive-unlabeled learning framework to learn segmentation masks from 3D bounding boxes. Specially, we consider the pixels outside of the bounding box as positively labeled data and the pixels inside the bounding box as unlabeled data. Our method can suppress the negative effects of pixels residing between the true segmentation mask and the 3D bounding box and produce accurate segmentation masks. We applied our method to segment a brain tumor. The experimental results on the BraTS 2017 dataset (Menze et al., 2015; Bakas et al., 2017a,b,c) have demonstrated the effectiveness of our method.

Keywords: brain tumor segmentation, deep learning, weakly-supervised, 3D bounding box, positive-unlabeled learning

OPEN ACCESS

Edited by:
Spyridon Bakas,
University of Pennsylvania,
United States

Reviewed by:
Shijie Zhao,
Northwestern Polytechnical University,
China
Madhura Ingalhalikar,
Symbiosis International University,
India

****Correspondence:***
Kayhan Batmanghelich
kayhan@pitt.edu

Specialty section:
This article was submitted to
Brain Imaging Methods,
a section of the journal
Frontiers in Neuroscience

Received: *31 August 2019*
Accepted: *23 March 2020*
Published: *28 April 2020*

Citation:
Xu Y, Gong M, Chen J, Chen Z and Batmanghelich K (2020) 3D-BoxSup: Positive-Unlabeled Learning of Brain Tumor Segmentation Networks From 3D Bounding Boxes.
Front. Neurosci. 14:350.
doi: 10.3389/fnins.2020.00350

1. INTRODUCTION

Gliomas are one of the most common brain tumors in adults. They can be categorized into different levels of aggressiveness, including High-Grade Gliomas (HGG) and Lower Grade Gliomas (LGG) (Louis et al., 2016). Gliomas consist of heterogeneous histological sub-regions, including peritumoral edema, the necrotic core, as well as the enhancing and non-enhancing tumor core (Menze et al., 2015). Magnetic Resonance Imaging (MRI) of brain tumors is commonly used to evaluate tumor progression and plan treatments. An MRI usually contains multi-modal data, such as T1-weighted, T2-weighted, contrast enhanced T1-weighted (T1ce), and Fluid Attenuation Inversion Recovery (FLAIR) images, which provide complementary information for analysis of brain tumors.

The automatic segmentation of brain tumors and subregions is a crucial pre-treatment step for the characterization and subtyping ofgliomas. This is a challenging problem because tumors vary in shape and size across patients and may have low contrast in some modalities. Recently, deep convolutional neural network (CNN)-based methods have achieved new records in brain tumor segmentation. Most of these methods are extensions of the U-Net structure (Ronneberger et al., 2015; Çiçek et al., 2016) in various ways (Isensee et al., 2017, 2018; Kamnitsas et al., 2017a,b; Wang et al., 2017; Li et al., 2018). For example, some works focus on the design of new convolutional network structures, such as using a mix between convolutional kernels and modifying the downsampling strategy (Havaei et al., 2015; Kamnitsas et al., 2017b). Other works have aimed to improve the method of fusing multimodal information. For example, Wang et al. (2017) suggested a patch-based framework combined with multi-view fusion techniques to reduce false positive segmentation. Kamnitsas et al. (2017a) proposed another fusion method through aggregation of predictions from a wide range of methods. The overall approach is more robust and reduces the risk of over-fitting to a particular dataset.

A key problem of CNN-based segmentation methods is the requirement of accurate pixel/voxel-level annotations. However, annotating a 3D image at the voxel level requires human expertise and is expensive and time consuming. Motivated by a recent work in weakly supervised segmentation in natural 2D images (Dai et al., 2015), we proposed to learn the segmentation network from 3D bounding box annotations. As pointed out in Dai et al. (2015), boxing out the object location is about 15 times faster than drawing the segmented mask (Dai et al., 2015). In 3D MRI images, the burden of annotating voxels is much higher than that of annotating 2D images because the number of voxels increases exponentially with image dimension. However, the cost of 3D bounding box annotations is comparable to that of 2D bounding boxes. Therefore, learning from 3D bounding boxes is valuable for brain tumor segmentation.

In this paper, we have investigated how to train a segmentation network from coarse but easily accessible 3D bounding box annotation. The main difficulty comes from the inaccurate annotations inside the bounding box. More specifically, the region bounded by a 3D bounding box contains tumor voxels as well background voxels. If one simply considers the voxels inside and outside of the bounding box as two classes, i.e., tumor and non-tumor; the non-tumor voxels inside the bounding box will have the wrong labels, and the learned network tends to classify the voxels outside but close to the tumor boundaries as tumor voxels. To solve this problem, we considered segmentation from 3D bounding boxes as a positive-unlabeled (PU) learning problem (Denis, 1998; Elkan and Noto, 2008) in which we consider the voxels outside of the bounding boxes as a positive class and the voxels inside the bounding box as unlabeled data. We have proposed the "3D-BoxSup" method to train a deep convolutional neural network reliably from 3D bounding box annotations with a non-negative risk estimator that is robust against overfitting (Kiryo et al., 2017). We conducted experiments on the BraTS 2017 dataset, and the results show that our method can obtain competitive accuracy by just learning from coarse bounding box annotations.

2. METHODS

Our 3D-BoxSup method is inspired by the BoxSup method (Dai et al., 2015), which aims to segment objects from 2D bounding box annotations. BoxSup is a straightforward method to train deep CNNs from coarse box annotations. It provides a biased objective function and utilizes the updated network in turn to improve the estimated segmentation masks used for training, which means the estimated segmentation masks in the previous training epoch are used as the ground truth mask for the next epoch. However, this iterative method is not practical for the 3D patch-based method because re-calculating the segmentation mask for each volume for each epoch has a high time-cost. In addition, it is unclear whether the iterative method will finally converge to the optimal solution. To achieve a considerable performance without iteratively updating the segmentation masks, we have cast the segmentation problem as a PU learning problem and applied a non-negative PU risk estimator (Kiryo et al., 2017) as the train objective to learn the segmentation network, where we viewed the 3D box annotated region as the unlabeled data and the area outside the box as positive data. In the following section, we have outlined the base model with a biased box-learning estimator as our baseline and the unbiased-box learning method as our proposed method.

In this section, we have first introduced the basic problem setup with precise mathematical definitions. Second, we have introduced our baseline convolutional neural network architecture, used for predicting the segmentation masks. Third, we have presented how the network is usually trained if the ground truth segmentation mask is available. Last, we have described our PU learning-based 3D-BoxSup method and the corresponding algorithm.

2.1. Problem Setup

In the real application, the accurate segmentation mask is difficult to acquire. Thus, in this paper, we only considered the cases where we only had access to box-labeled segmentation data.

Let S_1 denote the annotated 3D box region where gliomas reside and S_0 denote the background area outside of the bounding box. Assuming we extract 3D patches from S_1 and S_0 for training, which is shown in **Figure 1**, the label of each voxel in S_1 and S_0 is assigned to 1 and 0, respectively. The proportion of non-tumor voxels is denoted as $\pi_{\mathrm{p}} = \frac{n_0}{n_0+n_1}$, where n_1 is the number of voxels inside the box and n_0 is the number of voxels outside of the box.

2.2. Training Data Generation

First, for each volume of a patient, we generated a 3D bounding box to roughly cover the whole tumor (WT) area (S_1), and the uncovered region was considered to be the background area S_0. We show an example of the box-labeled data in **Figure 1B**. For convenience, we only showed the box label with a 2D yellow rectangle. It should be noted that the training label in our case is actually a 3D bounding box for each volume. In our experiment, we generated this 3D bounding box from the accurate ground truth segmentation mask, and we assumed that we did not have access to this accurate segmentation data during the training time, which was available in our testing.

We then followed the standard preprocessing step to process the original 3D input images (Bakas et al., 2018). To reduce the sensitivity to absolute pixel intensities variations, an intensity normalization step is applied to each volume of all subjects by subtracting the mean and dividing by the standard deviation so that each MR volume will have a zero mean and unit variance, which is operated to each volume dependently. In practice, as only the central region that contains the brain is used, the mean and standard deviation are estimated using this brain area; where we exclude the black area outside the brain with voxel value 0. Finally, we extract 200 patches per patient with patch size 48 × 48 × 48 in S_1 and S_0. If the extracted patch of S_1 is partial beyond the boxed area, we pad 0 value to the exceeded part for the segmentation mask. In our experiment, we randomly selected 3D patches from area S_1 and S_0 with a proportion of 0.8 and 0.2, respectively. In all of the following settings, we allocated patches from S_1 with the label 1 and patches from S_0 with the label 0.

2.3. Network Structure

To build a deep network for 3D patch segmentation, we applied the 3D U-Net (Çiçek et al., 2016), consisting of an encoder and a decoder network with skip connections similar to our base model. In contrast to (Çiçek et al., 2016), we removed the last down-sampling layer and the first up-sampling layer for the LHS and RHS of the 3D U-Net, respectively. This is because the down-sampling structure would eliminate edge features of brain tumor. Our modified 3D U-Net is shown in **Figure 1A**.

2.4. Learning With Ground Truth Mask

In the fully supervised brain tumor segmentation task, accurately annotated masks were provided for training. Assuming the mask prediction function modeled by a CNN is $\hat{\mathbf{y}} = f(\mathbf{x};\theta) \in \mathbf{x} \in \mathbb{R}^{d\times d\times d}$, where $\mathbf{x} \in \mathbb{R}^{d\times d\times d}$ is a randomly chosen 3D patch U from a patient V, and θ is the global trainable parameter. The ground truth patch tumor mask is $\mathbf{y} \in \mathbb{R}^{d\times d\times d}$. To learn the network parameters, we can apply the sigmoid function to generate probability values and cross-entropy loss function to evaluate voxel-wise prediction error. The objective function for a single value in the predicted mask can be written:

$$L_{mask}(y,\hat{y}) = (1-y)\cdot\log(\frac{1}{1+e^{-\hat{y}}}) + y\cdot\log(\frac{1}{1+e^{\hat{y}}}), \qquad (1)$$

where y is a single value in the ground truth mask, and $\hat{y}$ is the corresponding value in the predicted mask. The overall empirical risk $\hat{R}_{mask}$ is a summation of $L_{mask}(y,\hat{y})$ on all voxels in all the 3D patches and can be efficiently minimized by using stochastic gradient descent (SGD) methods.

2.5. Positive-Unlabeled Learning With Box Labeled Data

When the images are only provided with bounding box annotations, it is much more difficult to learn the segmentation network $f(\mathbf{x};\theta)$ because the voxels inside the box can be classified

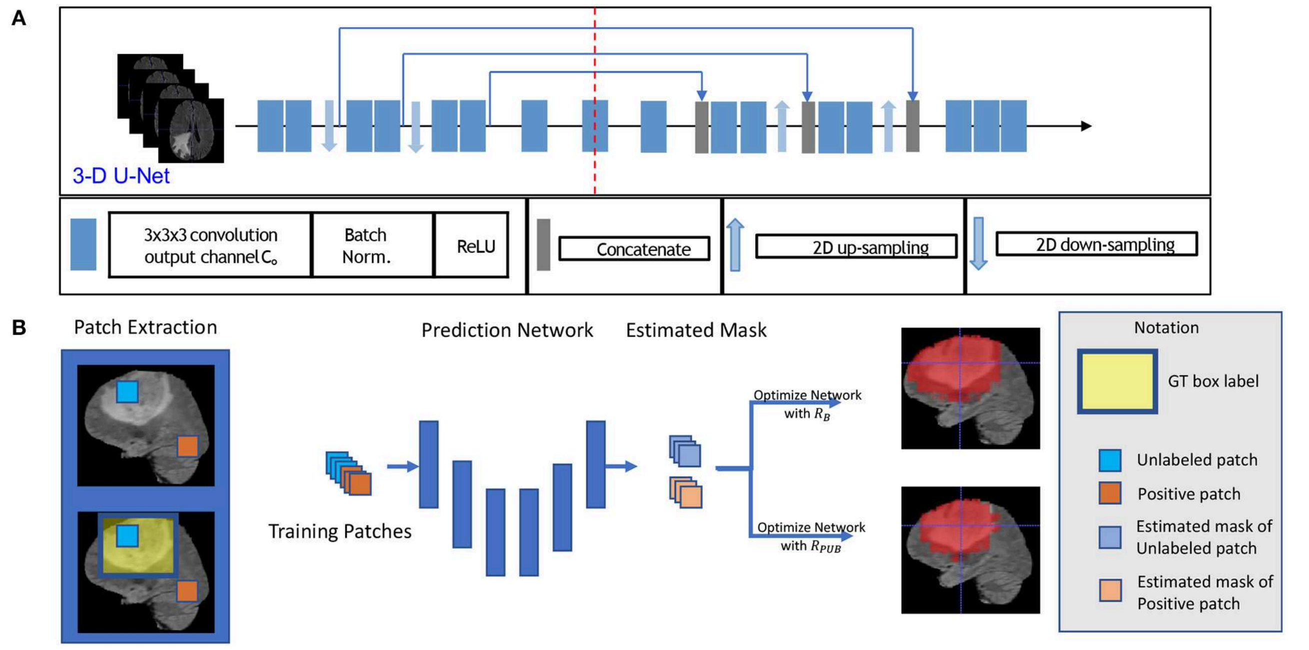

FIGURE 1 | This figure shows the training model with box labeled data. The network structure is shown in sub-figure **(A)**, which is a typical 3D U-Net. The general training process is shown in sub-figure **(B)**. To be intuitive, we applied a 2D slice as our example in **(B)** in the training process, and we fed 3D patches with a 3D bounding box label to optimize our model.

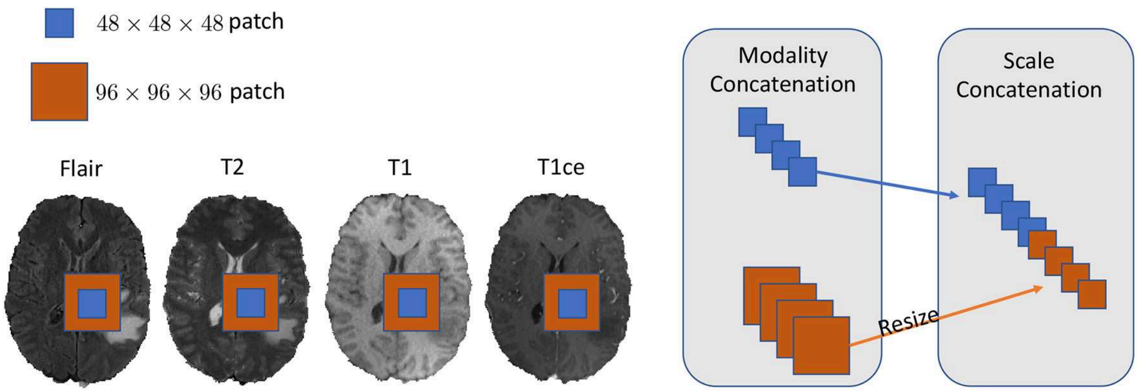

FIGURE 2 | In this figure, we show the example of how to reconstruct our training data in the 2D aspect for better comprehensive as it is easy to implement in 3D level as well. We concatenated the Flair, T2, T1, and T1ce models in the channel dimension and combined extra scale information to enhance the model training.

as either tumor or non-tumor. A straightforward solution would be assigning all the voxels in patches coming from S_1 with label 1 and labeling all the voxels in patches coming from S_0 with label 0. We could then train the segmentation network using the cross-entropy loss (1), which we call the "Naive-BoxSup" method. The problem with the naive method is that some non-tumor voxels inside the box are wrongly assigned with tumor class label 1. As a result, the learned network tends to classify the voxels outside but close to the tumor boundaries as tumor voxels.

To alleviate this problem of the Naive-BoxSup, we proposed to consider segmentation from boxes as a positive-unlabeled learning problem. We can ensure that patches extracted from S_0 only contains positively-labeled voxels (0 is considered the positive label), which are far away from tumor area. In the bounding box area S_1, voxels in S_1 can be considered as an unlabeled object. Thus, segmentation network learning from bounding box annotations is a typical positive-unlabeled learning problem, which tries to learn a classifier to model the distribution of positive data p_p and negative data p_n by using only positive labeled data and unlabeled data. In the following, we have described how we applied a recently proposed non-negative PU-Learning loss (Kiryo et al., 2017) to train our segmentation network. We chose to use this loss because the non-negative constraint on the loss makes it less prone to overfitting when a deep network is being learned.

Let $p(x)$ denote the marginal distribution of input features corresponding to a single output y in the predicted segmentation mask. By stacking all the 3D patches together, we can get a sample $\{(x_i, y_i)\}_{i=1}^n$. Let $p_p(x) = p(x|y=0)$ and $p_n(x) = p(x|y=1)$ denote the positive and negative class conditional distributions, respectively. We have

$$p(x) = \pi_p p_p(x) + (1-\pi_p) p_n(x). \tag{2}$$

Equivalently, $(1-\pi_p)p_n(x) = p(x) - \pi_p p_p(x)$. Let $L(y, \hat{y})$ be a general loss function evaluating the distance between output and ground truth labels, which is cross-entropy loss in our case. This is denoted by

$$R_p^+(\theta) = E_{x\sim p_p(x)} L(f(x,\theta), y=0), \tag{3}$$

$$R_n^-(\theta) = E_{x\sim p_n(x)} L(f(x,\theta), y=1), \tag{4}$$

$$R_p^-(\theta) = E_{x\sim p_p(x)} L(f(x,\theta), y=1), \text{ and} \tag{5}$$

$$R_u^-(\theta) = E_{x\sim p(x)} L(f(x,\theta), y=1). \tag{6}$$

By using (), we can have an approximation of the risk on the true distribution $R(f) = E_{(x,y)\sim p(x,y)} L(f(x,\theta), y) = \pi_p R_p^+(f) + \pi_n R_n^-(f)$ by

$$R_{PU} = \pi_\mathrm{p} R_p^+(\theta) + R_u^-(\theta) - \pi_\mathrm{p} R_p^-(\theta). \tag{7}$$

Theoretically, we can minimize R_{PU} to learn the optimal *theta* for our segmentation network. However, as pointed out in Kiryo et al. (2017), if the model is very flexible, empirical risks on training data will go negative, and we will suffer from serious over-fitting. Since our model is a very complicated convolutional neural network, we applied a non-negative risk estimator (Kiryo et al., 2017), as with the objective function:

$$R_{PUB} = \pi_\mathrm{p} R_p^+(\theta) + \max\{0, R_u^-(\theta) - \pi_\mathrm{p} R_p^-(\theta)\}. \tag{8}$$

In practice, we need to replace the risk terms by their empirical estimates from data:

$$\pi_\mathrm{p}\widehat{R}_\mathrm{p}^+(\theta) = -\pi_\mathrm{p}\frac{1}{n}\sum_{i=1}^{n}(1-y_i)\cdot\log(1-\frac{1}{1+e^{-f(x_i;\theta)}})$$

$$\widehat{R}_\mathrm{u}^-(\theta) = \frac{1}{n}\sum_{i=1}^{n} y_i\cdot\log(\frac{1}{1+e^{-f(x_i;\theta)}})$$

$$\pi_\mathrm{p}\widehat{R}_\mathrm{p}^-(\theta) = -\pi_\mathrm{p}\frac{1}{n}\sum_{i=1}^{n} y_i\cdot\log(1-\frac{1}{1+e^{-f(x_i;\theta)}}).$$

The overall algorithm is shown in Algorithm 1. We used the ADAM optimizer to optimize the empirical risk. In our algorithm, we set $\pi_\mathrm{p} = 0.75$ and we set $\gamma = 1, \eta = 0.5$, which is a very common choice for PU learning.

3. EXPERIMENT

To demonstrate the effectiveness of our method, we presented a number of experiments examining different aspects of our method. After introducing the implementation details, we evaluated our methods on BraTS (Wang et al., 2017) brain tumor training dataset. We compared the segmentation performance of our 3D-BoxSup method with the Naive-BoxSup segmentation method and show advantages of our proposed method over the baseline approach.

Algorithm 1: Optimization of Our 3D-BoxSup segmentation algorithm

```
Input: training data $(x_i, y_i)$;
       hyperparameters $0 \leq \beta \leq \pi_{\mathrm{p}}$ and $0 \leq \gamma \leq 1$
Output: model parameter $\theta$ for $f(x;\theta)$
Let $\mathcal{A}$ be an external ADAM optimizer (Kingma and Ba, 2014)
while no stopping criterion has been met:
    Shuffle $(x_i, y_i)$ into $N$ mini-batches
    for $i = 1$ to $N$:
        if $\widehat{R}_{\mathrm{u}}^{-}(\theta) - \pi_{\mathrm{p}}\widehat{R}_{\mathrm{p}}^{-}(\theta) \geq -\beta$:
            Set gradient $\nabla_\theta \widehat{R}_{\mathrm{pu}}(\theta)$
            Update $\theta$ by $\mathcal{A}$ with its current step size $\eta$
        else:
            Set gradient $\nabla_\theta(\pi_{\mathrm{p}}\widehat{R}_{\mathrm{p}}^{-}(\theta) - \widehat{R}_{\mathrm{u}}^{-}(\theta))$
            Update $\theta$ by $\mathcal{A}$ with a discounted step size $\gamma\eta$
```

3.1. Training Setting

We got all our training data from BraTS web[1] to evaluate our method. The training data consisted of 285 patients, including segmented masks annotated by human experts. These training data were separated into two categories, including HGG and LGG, each containing 210 HGG and 75 LGG images. There is an imbalance between HGG and LGG, and the data distributions of HGG and LGG were also different, especially for TC and ET. Each patient had four sequences, which are FLAIR, T2, T1, and T1ce. In training time, we randomly split the whole training set to 80% training set and 20% as our evaluation set, and we carried out five folds testing in this manner. We only use the ground truth segmented label during evaluation. We fed all of the sequences into our network by combining them in channel dimension. Thus, our input data are in 5D, the dimensions of which are batch, sequences, width, length, and depth. The training model structure is shown in **Figure 1A**. To generate the training data, we followed the abovementioned section 2.2 method, and the proportion of the non-tumor voxels was $\pi_{S_2} = 0.75$.

We set our training batch size to 64, and each training patch voxel size is 48 × 48 × 48 for saving memory, which is sufficient to train the model; another aspect to the design of such a parch size is that a larger patch would contain more background voxel, which means the model would over-fit the background and would not be able learn a pattern of Whole Tumor segmentation. For the data training strategy, we randomly generated 40,000 locations as the center point of patches for each patient volume; finally, only 200 locations were selected as our training patches. To fully utilize the information from each model provided with FLAIR, T2, T1, and T1ce, we reconstructed our multi-modal data by stack theses modals in the channel dimension, which can be directly fed to convolutional neural networks (CNNs). Also, imitating the technique from (Bakas et al., 2018), we enabled the network to capture the multiscale information from data. To do so, we got the 96 × 96 × 96 patches for each modal, extracted in the same way as the 48 × 48 × 48 patches, which were two times bigger than the basic training patch; this bigger patch also belongs to the same center location as the basic training patch. Then, we resized the 96 × 96 × 96 patches to 48 × 48 × 48. Finally, we concatenated all the different models and scaled patches, which is shown in **Figure 2**. Thus, the input patch size of our model was $batchsize \times 8 \times 48 \times 48 \times 48$. We trained our whole network using Pytorch (Paszke et al., 2017), which is a new hybrid front-end seamlessly transitions between eager mode and graph mode to provide both flexibility and speed. NVIDIA TITAN XP GPU was applied to train our network, and the cost was about 11 gigabytes GPU RAM. The whole training process was finished in 4 h with 5 epochs, and each epoch traversed the whole training dataset. To optimize our model, we chose the common gradient descent algorithm Adam.

3.2. Evaluation Metrics

3.2.1. Dice Coefficient

The Dice-Coefficient Score was calculated as the performance metric. This measure states the similarity between clinical Ground Truth annotations and the output segmentation of the model which are A and B respectively. Afterwards, we calculated the average of those results to obtain the overall dice coefficient of the models.

$$D = \frac{2|A \bigcap B|}{|A| + |B|} \tag{9}$$

3.2.2. Hausdorff Distance

The Hausdorff Distance is mathematically defined as the maximum distance of a set to the nearest point in the other set, in other words, how close the segmentation and the expected output are. In most evaluations, we usually adopt the 95% Hausdorff Distance, Hausdorff95, which means the chosen distance is greater or equal to exactly 95% of the other distance in two point sets.

$$\overrightarrow{d_H}(A, B) = \max_{a \in A} \max_{b \in B} d(a, b) \tag{10}$$

$$H(A, B) = max\{\overrightarrow{d_H}(A, B), \overrightarrow{d_H}(B, A)\} \tag{11}$$

3.3. Experimental Results

To compare the model performance straightly, we gave the segmented mask generated by the baseline Naive-BoxSup method and our proposed method 3D-BoxSup, shown in **Figure 3**. Corresponding to the evaluated metric in section 3.2, the quantitative results are shown in **Table 1**. The chosen samples were randomly picked from HGG testing set and LGG testing set.

[1] https://www.med.upenn.edu/sbia/brats2018/data.html

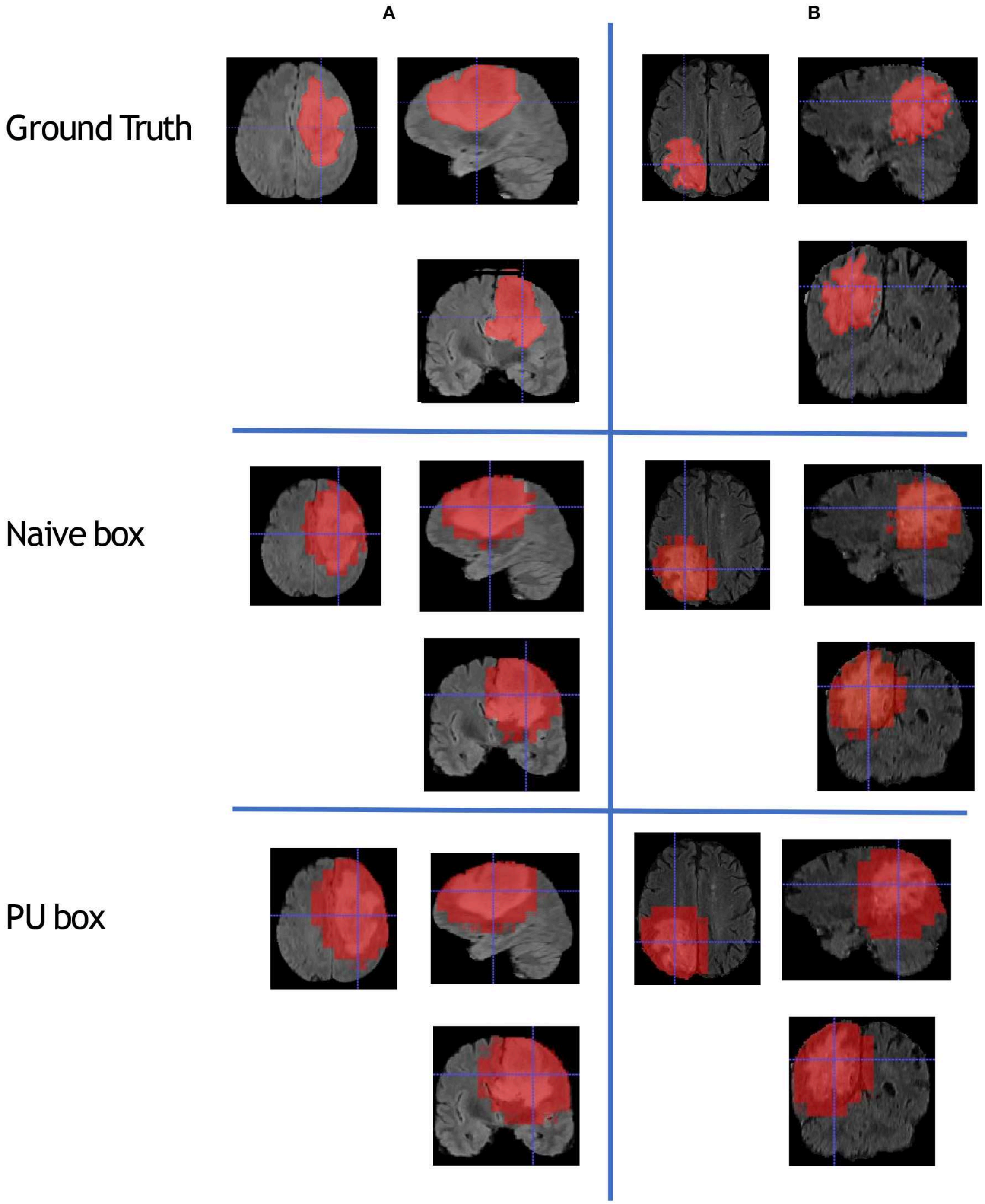

FIGURE 3 | This figure shows that we randomly sampled two patients from HGG testing set and LGG testing set respectively. Each column represents the applied method, and each row is the chosen patient. The A patient is from HGG samples, and the B patient is from LGG samples. The estimated segmentation result of WT is shown by both the naive method and our proposed PU box method. To better visualize the segmented result, we provide three different views: axial, coronal, and sagittal.

As can be seen from **Figure 3**, our proposed 3D-BoxSup method obviously produced a more accurate segmentation mask than the Naive-BoxSup method, which produced a much more noisy mask around the tumor boundary. It seems the Naive-BoxSup over-fits the data from no-tumor area S_0, which verifies that our method is able to alleviate this over-fitting and learning better from box area S_1.

In terms of the quantitative metrics of the Dice Score and Hausdorff Distance, our method performs better than the baseline method in each aspect, especially the Dice Score.

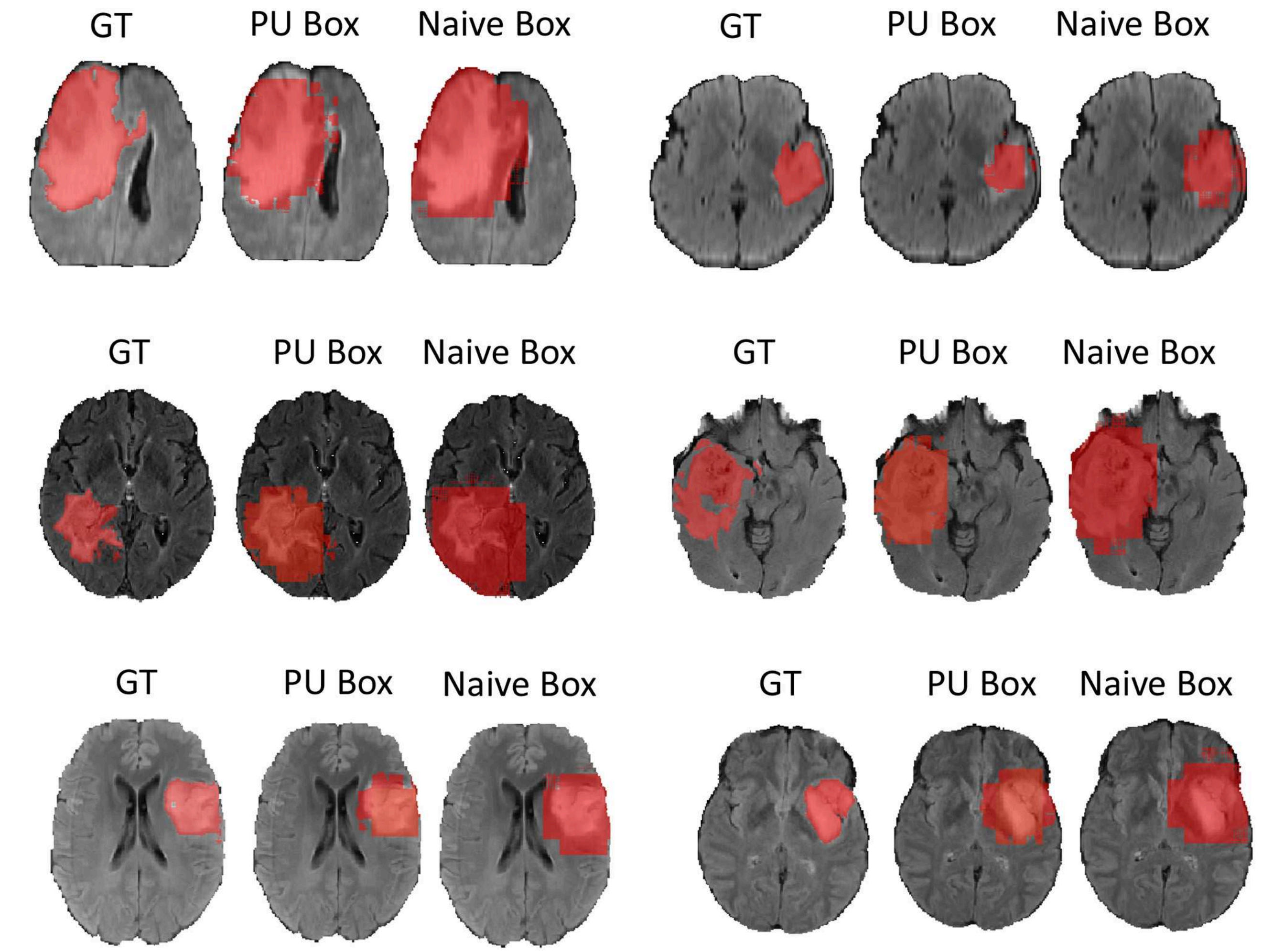

FIGURE 4 | We randomly choose six patients from testing set as more example displaying and for simplicity we only show the view of axial plane.

TABLE 1 | Mean values of Dice and Hausdorff measurements of the proposed method on the BraTS 2018 validation set.

	Dice	Hausdorff (mm)	Hausdorff95 (mm)
	WT	WT	WT
Naive-BoxSup (baseline)	0.49 ± 0.04	31.213 ± 2.316	20.857 ± 1.503
3D-BoxSup (ours)	0.62 ± 0.02	28.641 ± 1.395	15.476 ± 1.132
Region Grow	0.50	39.920	29.151

WT denotes whole tumor.

Visually, as shown in **Figure 4**, out method also generates finer segmentation mask than the baseline method. The variance of our 5-folds evaluation results is also smaller than the baseline model, which means our model is more robust. Also, due to the fact that we only applied a simple post-process for fill in the hole of segmented mask, the Hausdorff Distance could be influenced by the wrong segmentation area, which is beyond the tumor area. Compared with the hand-crafted region grow method (watershed clustering Ng et al., 2006), we set the threshold of discontinuities in gray-scale to be 0.5, and the result of the region grow is shown as below. To evaluate the region grow method, we tested it on both training data and testing data as tge region grow does not need to be trained. Overall, our proposed method shows a superiority when given a weak box annotation.

4. CONCLUSION

Precisely labeled data is limited in real world, especially for medical data; it would cost a significant amount time and labor to annotate the data, and it would also require a highly qualified doctor as the annotator. Thus, we need to refer to some easily labeled data, saving time, and also explore the information derived from these weakly labeled data. In this paper, we explored one of the possibilities of weakly supervised approach on medical image segmentation. Our method is called the "3D-BoxSup," which only acquired a 3D bounding box label for brain tumor.

Compared to the traditional supervised labeled data, which needs a fine boundary for tumor annotation, our annotated data is more accessible. However, training on the box labeled data would lead to over-fitting of the background as well as a biased risk function. Box labeled data is a typical positive-unlabeled task, and we thus proposed to apply the non-negative PU risk function (Kiryo et al., 2017) to boost the performance of our model. We have shown the effectiveness of our proposed method on the data provided by BRATS challenge (Menze et al., 2015). Since our model is a general method when tackling such box labeled data, our method can be further applied to mostly if not all of the segmentation tasks.

DATA AVAILABILITY STATEMENT

The datasets analyzed in this article are not publicly available. Requests to access the datasets should be directed to BRATS challenge.

AUTHOR CONTRIBUTIONS

YX mainly implemented the method and conducted the experiments. MG provided the idea of this paper and contributed to the writing of the paper. JC and ZC substantially contributed to the revision. ZC performed the Region Growing experiments requested by the reviewers, and the JC helped with the writing of the revised paper. KB supervised the whole process, including the development of the concept, writing, revision, and other general advice.

FUNDING

This work was partially supported by NIH Award Number 1R01HL141813-01, NSF 1839332 Tripod+X, and SAP SE. We gratefully acknowledged the support of NVIDIA Corporation with the donation of the Titan X Pascal GPU used for this research. We were also grateful for the computational resources provided by Pittsburgh SuperComputing grant number TG-ASC170024.

REFERENCES

Bakas, S., Akbari, H., Sotiras, A., Bilello, M., Rozycki, M., Kirby, J. S., et al. (2017a). Advancing The Cancer Genome Atlas glioma MRI collections with expert segmentation labels and radiomic features. *Nature Scientific Data* 4:170117. doi: 10.1038/sdata.2017.117

Bakas, S., Akbari, H., Sotiras, A., Bilello, M., Rozycki, M., Kirby, J., et al. (2017c). Segmentation Labels and Radiomic Features for the Pre-operative Scans of the TCGA-LGG collection. *The Cancer Imaging Archive*. doi: 10.7937/K9/TCIA.2017.GJQ7R0EF

Bakas, S., Akbari, H., Sotiras, A., Bilello, M., Rozycki, M., Kirby, J. S., et al. (2017b). Advancing the cancer genome atlas glioma MRI collections with expert segmentation labels and radiomic features. *Sci. Data* 4:170117. doi: 10.1038/sdata.2017.117

Bakas, S., Reyes, M., Jakab, A., Bauer, S., Rempfler, M., Crimi, A., et al. (2018). Identifying the best machine learning algorithms for brain tumor segmentation, progression assessment, and overall survival prediction in the BRATS challenge. *CoRR*, abs/1811.02629.

Çiçek, Ö., Abdulkadir, A., Lienkamp, S. S., Brox, T., and Ronneberger, O. (2016). 3d u-net: learning dense volumetric segmentation from sparse annotation. *CoRR*, abs/1606.06650.

Dai, J., He, K., and Sun, J. (2015). Boxsup: exploiting bounding boxes to supervise convolutional networks for semantic segmentation. *CoRR*, abs/1503.01640.

Denis, F. (1998). "Pac learning from positive statistical queries," in *International Conference on Algorithmic Learning Theory* (Berlin; Heidelberg: Springer), 112–126.

Elkan, C., and Noto, K. (2008). "Learning classifiers from only positive and unlabeled data," in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Las Vegas, NV), 213–220. doi: 10.1145/1401890.1401920

Havaei, M., Davy, A., Warde-Farley, D., Biard, A., Courville, A. C., Bengio, Y., et al. (2015). Brain tumor segmentation with deep neural networks. *CoRR*, abs/1505.03540.

Isensee, F., Jaeger, P., Full, P. M., Wolf, I., Engelhardt, S., and Maier-Hein, K. H. (2017). Automatic cardiac disease assessment on cine-MRI via time-series segmentation and domain specific features. *CoRR*, abs/1707.00587.

Isensee, F., Kickingereder, P., Wick, W., Bendszus, M., and Maier-Hein, K. H. (2018). Brain tumor segmentation and radiomics survival prediction: contribution to the BRATS 2017 challenge. *CoRR*, abs/1802.10508.

Kamnitsas, K., Bai, W., Ferrante, E., McDonagh, S. G., Sinclair, M., Pawlowski, N., et al. (2017a). Ensembles of multiple models and architectures for robust brain tumour segmentation. *CoRR*, abs/1711.01468.

Kamnitsas, K., Ledig, C., Newcombe, V. F. J., Simpson, J. P., Kane, A. D., Menon, D. K., et al. (2017b). Efficient multi-scale 3d CNN with fully connected CRF for accurate brain lesion segmentation. *Med. Image Anal.* 36, 61–78. doi: 10.1016/j.media.2016.10.004

Kingma, D. P., and Ba, J. (2014). Adam: a method for stochastic optimization. *CoRR*, abs/1412.6980.

Kiryo, R., Niu, G., du Plessis, M. C., and Sugiyama, M. (2017). Positive-unlabeled learning with non-negative risk estimator. *CoRR*, abs/1703.00593.

Li, X., Chen, H., Qi, X., Dou, Q., Fu, C.-W., and Heng, P.-A. (2018). H-denseunet: hybrid densely connected unet for liver and tumor segmentation from ct volumes. *IEEE Trans. Med. Imaging* 37, 2663–2674. doi: 10.1109/TMI.2018.2845918

Louis, D. N., Perry, A., Reifenberger, G., Von Deimling, A., Figarella-Branger, D., Cavenee, W. K., et al. (2016). The 2016 world health organization classification of tumors of the central nervous system: a summary. *Acta Neuropathol.* 131, 803–820. doi: 10.1007/s00401-016-1545-1

Menze, B. H., Jakab, A., Bauer, S., Kalpathy-Cramer, J., Farahani, K., Kirby, J., et al. (2015). The multimodal brain tumor image segmentation benchmark (brats). *IEEE Trans. Med. Imaging* 34, 1993–2024. doi: 10.1109/TMI.2014.2377694

Ng, H. P., Ong, S. H., Foong, K. W. C., Goh, P. S., and Nowinski, W. L. (2006). "Medical image segmentation using k-means clustering and improved watershed algorithm," in *Proceedings of the 2006 IEEE Southwest Symposium on Image Analysis and Interpretation*, SSIAI '06 (Washington, DC: IEEE Computer Society), 61–65.

Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., et al. (2017). Automatic differentiation in pytorch.

Ronneberger, O., Fischer, P., and Brox, T. (2015). U-net: convolutional networks for biomedical image segmentation. *CoRR*, abs/1505.04597.

Wang, G., Li, W., Ourselin, S., and Vercauteren, T. (2017). Automatic brain tumor segmentation using cascaded anisotropic convolutional neural networks. *CoRR*, abs/1709.00382.

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.