MDPI

Article

JMLNet: Joint Multi-label Learning Network for Weakly Supervised Semantic Segmentation in Remote Sensing Images

Rongxin Guo [1,2,3,4], Xian Sun [1,2,3,4]*, Kaiqiang Chen [1,3], Xiao Zhou [1,5], Zhiyuan Yan [1,3], Wenhui Diao [1,3] and Menglong Yan [1,3]

1 Aerospace Information Research Institute, Chinese Academy of Sciences, Beijing 100190, China; sunxian@mail.ie.ac.cn(X.S.);chenkaiqiang14@mails.ucas.ac.cn(K.C.); zhouxiao@aircas.ac.cn(X.Z.); yanzy@aircas.ac.cn(Z.Y.); diaowh@aircas.ac.cn(W.D.); yanml@aircas.ac.cn(M.Y.)
2 School of Electronic, Electrical and Communication Engineering, University of Chinese Academy of Sciences, Beijing 100190, China.
3 Key Laboratory of Network Information System Technology (NIST), Aerospace Information Research Institute, Chinese Academy of Sciences, Beijing 100190, China.
4 University of Chinese Academy of Sciences, Beijing 100190, China.
5 Key Laboratory of Technology in Geo-spatial Information Processing and Application System, Chinese Academy of Sciences, Beijing 100190, China.
* Correspondence: sunxian@mail.ie.ac.cn

Version August 5, 2020 submitted to Journal Not Specified

Abstract: Weakly supervised semantic segmentation in remote sensing images has attracted growing research attention due to the significant saving in annotation cost. Most of the current approaches are based on one specific pseudo label. These methods easily overfit the wrongly labeled pixels from noisy label and limit the performance and generalization of segmentation model. To tackle these problems, we propose a novel joint multi-label learning network(JMLNet) to help the model learn common knowledge from multiple noisy labels and prevent the model from overfitting one specific label. Our combination strategy of multiple proposals is that we regard them all as ground truth and propose three new multi-label losses to use multi-label guide segmentation model in the training process. JMLNet also contains two methods to generate high-quality proposals, which further improve the performance of segmentation task. First we propose a detection-based Grad-CAM (DetCAM) to generate segmentation proposals from object detectors. Then we use DetCAM to adjust GrabCut algorithm and generate segmentation proposals (CAMCut). We report the state-of-the-art results on semantic segmentation task of iSAID when training with bounding boxes annotations.

Keywords: deep learning; image segmentation; weak supervision; remote sensing image; multi-label learning

Semantic segmentation in remote sensing images is a significant task, which aims at classifying each pixel in the given remote sensing images. It is usefull for city planning, weather service and other applications of remote sensing.

Recently, Fully Convolutional Network (FCN) [1] based methods [2–4] have made great progress in semantic segmentation. These works require pixel-level supervised data in the training process. However, it is rather expensive to create pixel-level semantic segmentation training sets. Pixel-level annotations cost about 15x more time [5] than bounding box annotations. Considering bounding boxes are more cheaper, we can research semantic segmentation with bounding boxes supervision.

Several weakly supervised segmentation methods [6–10] explore closing the gap between pixel-level supervision and bounding boxes supervision. These methods mainly refine segmentation

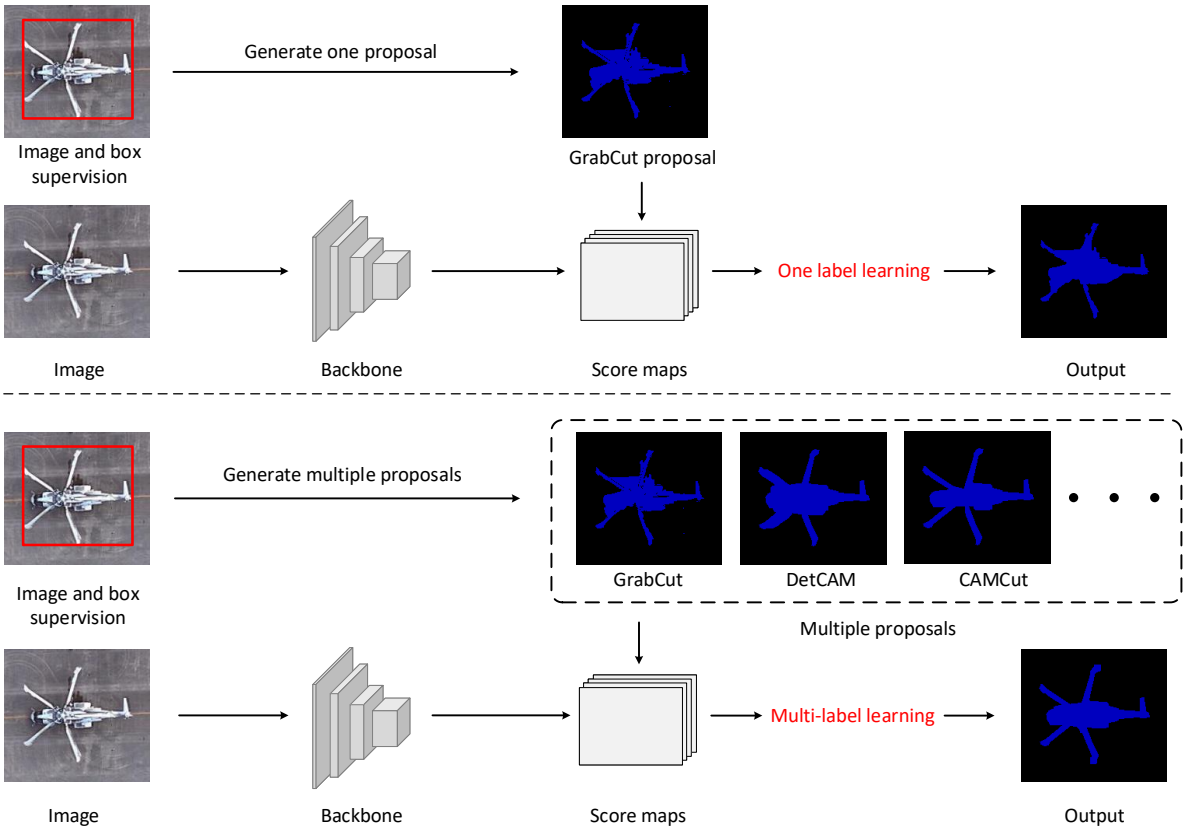

Figure 1. The overall pipeline of previous weakly supervised semantic segmentation methods (top) and our proposed JMLNet (bottom). Previous methods generate one specific proposal and use it in the training process. However, we first generate multiple proposals as multi-label supervision and use multi-label loss to train the segmentation model.

proposals from bounding boxes supervision, then take these segmentation proposals as pixel-level supervision and train deep FCN model. These methods mainly use traditional proposals like CRF [8], MCG [11] and GrabCut [12]. BoxSup [6] takes MCG [11] as initial segmentation proposals and updated the proposals in an iterative way. SDI [13] takes intersection of MCG [11] and GrabCut [12] as segmentation proposals. Song *et al*.[10] use dense CRF [8] as segmentation proposals. These methods all feed one specific proposal to segmentation model, which easily overfit the wrongly labeled pixels from noisy label and limit the performance and generalization of segmentation model. So it is a natural idea to tackle these problems by taking advantage of multiple proposals in the training process.

To train with multiple proposals, traditional combining methods take intersection [13] of two kinds of segmentation proposals as supervision to reduce the noise. Pixels out of intersection are ignore in training. These pixels usually take up mainly part of box area in difficult situations, which reduces the semantic information and limits segmentation model performance. We propose a joint multi-label learning network(JMLNet) to address the issue. The overall pipeline of our JMLNet is in Fig. 1. Different from simply using intersection of two proposals or only use one specific proposal, we regard multiple proposals as multi-label and make all noisy proposals contribute in the training process. Specifically, we propose three multi-label losses for training, including multi-label average loss (MA-Loss), multi-label minimum loss (MM-Loss), and box-wise multi-label minimum loss (BMM-Loss). These loss functions help segmentation model learn common knowledge from multiple noisy labels and prevent the model from overfitting one specific label.

The quality of Proposals is vital to weakly supervised semantic segmentation. Previous approaches train the models with MCG, GrabCut, or CRF proposals based on box supervision. Lacking high-level semantic knowledge, these proposals are easy to confused in complicated scenes. As shown

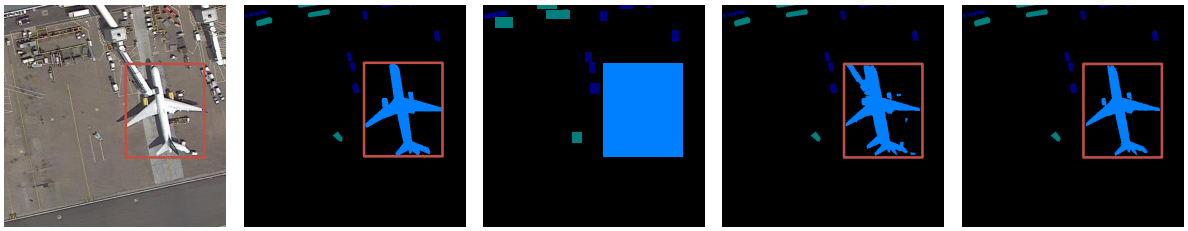

Figure 2. Segmentation proposals obtained from bounding box. (a) A training image. (b) ground truth. (c) rectangle proposals. (d)GrabCut [12] proposals. (e) We propose CAMCut proposals, which perform better than traditional proposals.

in Fig. 2 (c), GrabCut confuses *building* and *plane* because of similar color. Low quality of traditional proposals damages the performance of segmentation model. We address this problem by proposing DetCAM and CAMCut, which generate high-quality pixel-level proposals. First, DetCAM aims to generate visual explanations and proper proposals from object detectors. DetCAM generates reliable proposals because the detection networks learn precise semantic information, as shown in Fig. 2 (d). Second, we use DetCAM to adjust GrabCut algorithm and generate proposals, which is denoted as CAMCut. As shown in Fig. 2 (e), CAMCut proposals are both reliable in the distinguished semantic area and detailed in instance edge. Our method improves the segmentation proposals' quality, which further improves the segmentation performance of JMLNet.

We summarize our contributions as follows:

- We propose a novel joint multi-label learning network(JMLNet), which first regards multiple proposals as multi-label supervision to train weakly supervised semantic segmentation model. JMLNet learns common knowledge from multiple noisy labels and prevent the model from overfitting one specific label.
- DetCAM and CAMCut methods are proposed to generate high-quality segmentation proposals, which further improve the segmentation performance of JMLNet. These proposals perform both reliable in the distinguished semantic area and detailed in instance edge.
- We report the state-of-the-art results on semantic segmentation segmentation tasks of iSAID when training using bounding boxes supervision, reaching comparable quality with the fully supervised model.

1. Related Work

We introduce the weakly supervised semantic segmentation methods of natural image and remote sensing image, region proposal from box supervision, and learning semantic knowledge with noisy labels that are related to our work.

1.1. Weakly Supervised Semantic Segmentation of Natural Image

Weakly supervised semantic segmentation methods of natural image can be classified into four parts, including image labels methods [14–19], points labels methods [9], scribbles labels methods [20,21], and bounding boxes labels methods [6,10,13]. We mainly introduce bounding boxes labels methods in the following paragraph. BoxSup [6] takes MCG [11] as initial segmentation proposals and updated the proposals in an iterative way. SDI [13] takes intersection of MCG [11] and GrabCut [12] as segmentation proposals. Song *et al*.[10] propose an attention model to focus on the foreground regions.

1.2. Weakly Supervised Semantic Segmentation of Remote Sensing Image

Weakly supervised semantic segmentation methods of remote sensing image can be also classified into four parts, including image labels methods [4,22], points labels methods [23], scribbles labels

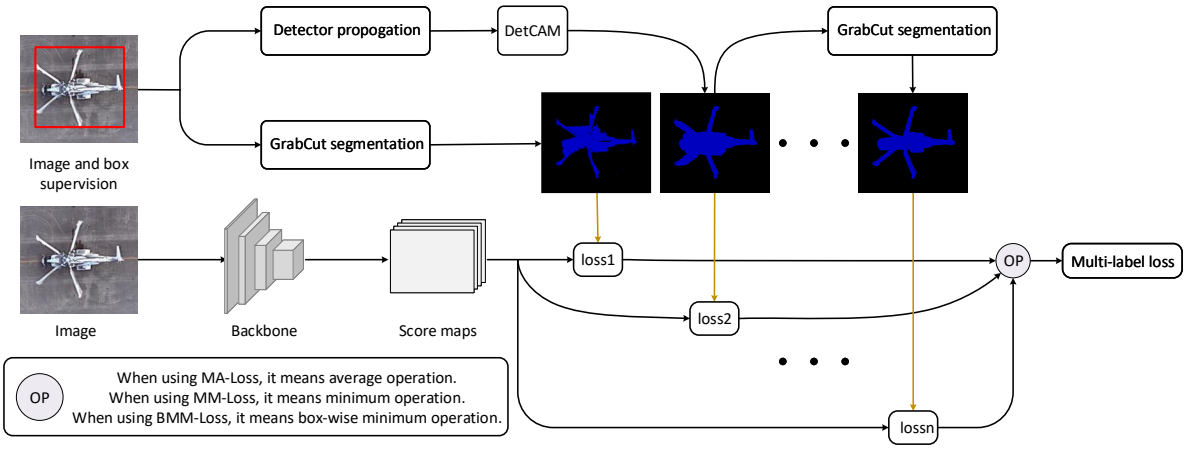

Figure 3. Overview of JMLNet. We generate multiple proposals as multi-label supervision and use multi-label loss to train the segmentation model.

methods [24], and bounding boxes labels methods [25]. WSF-NET [4] introduces a feature-fusion network to fuse different level feature of FCN [1] and increase the ability of feature representation. SPMF-Net [22] combines superpixel pooling to segmentation methods and use low level feature to get detail prediction. Wang *et al*. [23] use CAM [14] proposals as ground truth and train FCN [1] based model. Wu *et al*. [24] propose an adversarial architecture based model for segmentation. Rafique *et al*. [25] convert the bounding box into probabilistic masks and propose a boundary based loss function to restrict the edge of predict map to close to bounding box. We separate weakly supervised semantic segmentation as two aspects, including region proposal from box supervision and learning semantic knowledge with noisy labels.

1.3. Region Proposal from Box Supervision

Without proper pixel-level supervision, weakly supervised methods extract region proposal from box supervision. [8,11,12] are the most popular region proposal methods. BoxSup [6] takes MCG [11] as initial segmentation proposals and updated the proposals in an iterative way. SDI [13] takes intersection of MCG [11] and GrabCut [12] as segmentation proposals. Song *et al*.[10] use dense CRF [8] as segmentation proposals. These region proposal methods extract proposals from class-agnostic low-level features, which lead to generating confusing proposals in complicated scenes because of lacking high-level semantic information. To this end, we propose a DetCAM method to generate visual explanations from object detectors and proper proposals by setting the threshold. DetCAM generates reliable proposals because the detection network learns precise semantic information. Then we use DetCAM to adjust GrabCut algorithm and generate training labels, which performs both reliable in the distinguished semantic area and detailed in instance edge.

1.4. Learning Semantic Knowledge with Noisy Labels

Though we can use [8,11,12] to generate proposals within bounding boxes annotations, there are still so many noises compared with a full-supervised label. How to learn with noisy labels becomes a key problem of weakly supervised semantic segmentation. SDI [13] directly uses the intersection of two kinds of segmentation proposals to reduce the noise. Song *et al*.[10] use different filling rates as priors to help the model training. These methods all use one specific pseudo label. We first propose JMLNet to combine multiple noisy labels in the training process. JMLNet helps the model learn common knowledge from multiple noisy labels and prevent it from overfitting one specific label.

2. Our Method

2.1. Overview

In this section, we introduce the general pipeline of JMLNet. As shown in Fig. 3, we collect multiple proposals like GrabCut, DetCAM, and CAMCut proposals as multi-label supervision and train the segmentation model with the proposed multi-label loss.

Generating pseudo supervision. Except for popular segmentation proposals with bounding boxes labels, we generate DetCAM and CAMCut proposals as pseudo supervision. As shown in Fig. 4, We train a Faster R-CNN [26] with Resnet50 [27] backbone as our object detection network using bounding boxes annotations. Then we calculate the DetCAM in feature map of Faster R-CNN and generate the pixel-level proposals. DetCAM is also used to adjust GrabCut algorithm and generate CAMCut proposals. All these proposals contribute in the training process.

Model training with multiple noisy labels. As shown in Fig. 1, we choose popular Deeplab v3 [28] as semantic segmentation model. Note that we collect multiple proposals $\{CRF, GrabCut, DetCAM, CAMCut\}$ for a single input image, so we propose multi-label average-loss (MA-Loss), multi-label minimum loss (MM-Loss), and box-wise multi-label minimum loss (BMM-Loss) to help the model learn common knowledge from multiple noisy labels and prevent the model from overfitting one specific label.

2.2. Multi-label losses for multiple proposals

Most semantic segmentation methods use pixel-wise cross entropy loss as loss function:

$$\mathcal{L}_{CE} = -\frac{1}{N}\sum_{n=1}^{N}\sum_{c=1}^{C} y_{n,c}\log p_{n,c} \tag{1}$$

where N is the number of pixels, C is the number of classes, $y \in \{0,1\}$ is the ground truth, and $p \in [0,1]$ is the estimated probability.

It is obvious that our pseudo proposals are all noisy within bounding boxes annotations and one specific proposal is hard to perform best in all image sets. Based on the analysis above, we propose three multi-label losses to help the model learn common knowledge from multiple noisy labels and prevent the model from overfitting one specific label. In practice, we propose multi-label average-loss (MA-Loss), multi-label minimum loss (MM-Loss), and box-wise multi-label minimum loss (BMM-Loss). Dealing with multiple noisy labels, an intuitive idea is to calculate the average value of cross entropy losses for multiple proposals. We denote it as multi-label average-loss (MA-Loss):

$$\mathcal{L}_{MA}(p,\mathcal{Y}) = \frac{1}{Z}\sum_{z=1}^{Z}(\mathcal{L}_{CE}(p,y_z)) \tag{2}$$

where $\mathcal{Y}$ denotes pseudo labels set, Z is the number of proposals types.

Further, we calculate the cross entropy losses for multi proposals and take the minimum value in back propagation. We denote it as multi-label minimum loss (MM-Loss):

$$\mathcal{L}_{MM}(p,\mathcal{Y}) = \min_{z}\mathcal{L}_{CE}(p,y_z), z \in [1,Z] \tag{3}$$

In weakly supervised segmentation, a set of box-level labeled data $\mathcal{D} = \{(\boldsymbol{I},\boldsymbol{B})\}$ are given, where $\boldsymbol{I}$ and $\boldsymbol{B}$ denote an image and box-level ground truth respectively. We know the pixels out of $\boldsymbol{B}$ are background class according to ground truth. So pixels in $\boldsymbol{B}$ are key problem for our case. We categorize image pixels into two sets $\mathcal{P}^+$ and $\mathcal{P}^-$ according to their coordinates position by

$$\mathcal{P}^+ = \{(i,j)|(i,j) \in \boldsymbol{B}\} \tag{4}$$

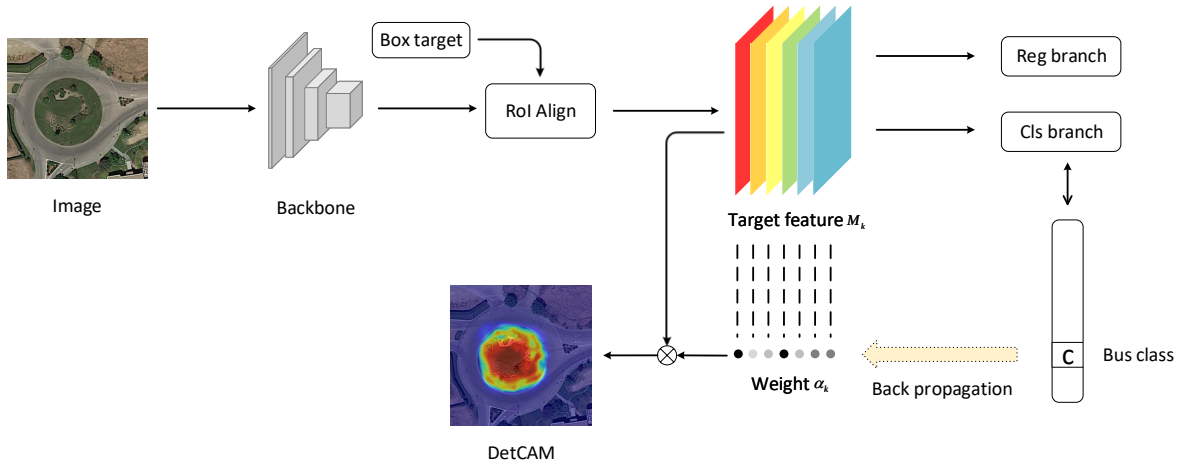

Figure 4. Overview of the DetCAM. We generate DetCAM using back propagation in the detector's classification branch. Best viewed in color.

$$\mathcal{P}^- = \{(i,j)|(i,j) \notin \boldsymbol{B}\} \tag{5}$$

where (i,j) is coordinate.

We calculate the minimum value of cross entropy losses for multi proposals in $\mathcal{P}^+$ as follows:

$$\mathcal{L}^+(p,\mathcal{Y}) = \frac{1}{n^+} \sum_{(i,j)\in\mathcal{P}^+} \min_z (\sum_{c=1} -y^z_{ijc} \log p_{ijc}), z \in [1,Z] \tag{6}$$

where y^Z_{ijc} indicate estimated probability of different proposals and n^+ indicates pixel number of $\mathcal{P}^+$.

For all coordinates (i,j) in $\mathcal{P}^-$, $y_{i,j} = 0$. We use cross entropy loss in $\mathcal{P}^-$ as follows:

$$\mathcal{L}^-(p,\mathcal{Y}) = -\frac{1}{n^-} \sum_{(i,j)\in\mathcal{P}^-} \log p^b_{ij} \tag{7}$$

where p^b_{ij} indicates estimated probability of background and n^- indicates pixel number of $\mathcal{P}^-$.

The $\mathcal{L}^+$ and $\mathcal{L}^-$ make up box-wise multi-label minimum loss (BMM-Loss):

$$\mathcal{L}_{BMM}(p,\mathcal{Y}) = \mathcal{L}^+(p,\mathcal{Y}) + \mathcal{L}^-(p,\mathcal{Y}) \tag{8}$$

Our proposed MA-Loss, MM-Loss and BMM-Loss help the model learn common knowledge from multiple noisy labels and prevent the model from overfitting one specific label.

2.3. Pseudo label generation by DetCAM and CAMCut

The DetCAM of our approach is shown in Fig. 4 and Algorithm 1. In order to obtain the DetCAM $\mathcal{D} \in \mathbb{R}^{u\times v}$ of width u and height v for target class, we first compute the gradient of target score s with respect to feature maps M_k, *i.e.*$\frac{\partial s}{\partial M_{ij}}$. $k \in [1,K]$ and K is the channel number of feature maps. These gradients flowing back obtain the weight α_k, which represents the weight of feature map M_k for target class.

$$\alpha_k = \frac{1}{uv} \sum_i \sum_j \frac{\partial s}{\partial M_{ij}} \tag{9}$$

We calculate a weighted combination of feature maps.

Algorithm 1: Generation of Low DetCAM Proposals $\mathcal{D}_\ell$ and High DetCAM Proposals $\mathcal{D}_h$

Input: Image $\boldsymbol{I}$; box supervision $\boldsymbol{B}$; low DetCAM threshold τ_ℓ; high DetCAM threshold τ_h.
Output: Low DetCAM proposals $\mathcal{D}_\ell$; high DetCAM proposals $\mathcal{D}_h$.

Feed the $\boldsymbol{I}$ into the detector's backbone to produce feature $\boldsymbol{F}^b$;
Feed the $\boldsymbol{F}^b$ and $\boldsymbol{B}$ into the RoIAlign to produce feature $\boldsymbol{F}^{roi}$;
Feed the $\boldsymbol{F}^{roi}$ into the detector's RCNN conv layer to produce feature M_k ;
Feed the M_k into the detector's classification branch to produce target score s ;
Get weight α_k by Eq. 9 ;
Get DetCAM $\mathcal{D}$ by Eq. 10 ;
for *each value* $p \in \mathcal{D}$ **do**
    if $p > \tau_\ell$ **then**
        $\mathcal{D}_\ell$.append(p);
    end
    if $p > \tau_h$ **then**
        $\mathcal{D}_h$.append(p);
    end
end

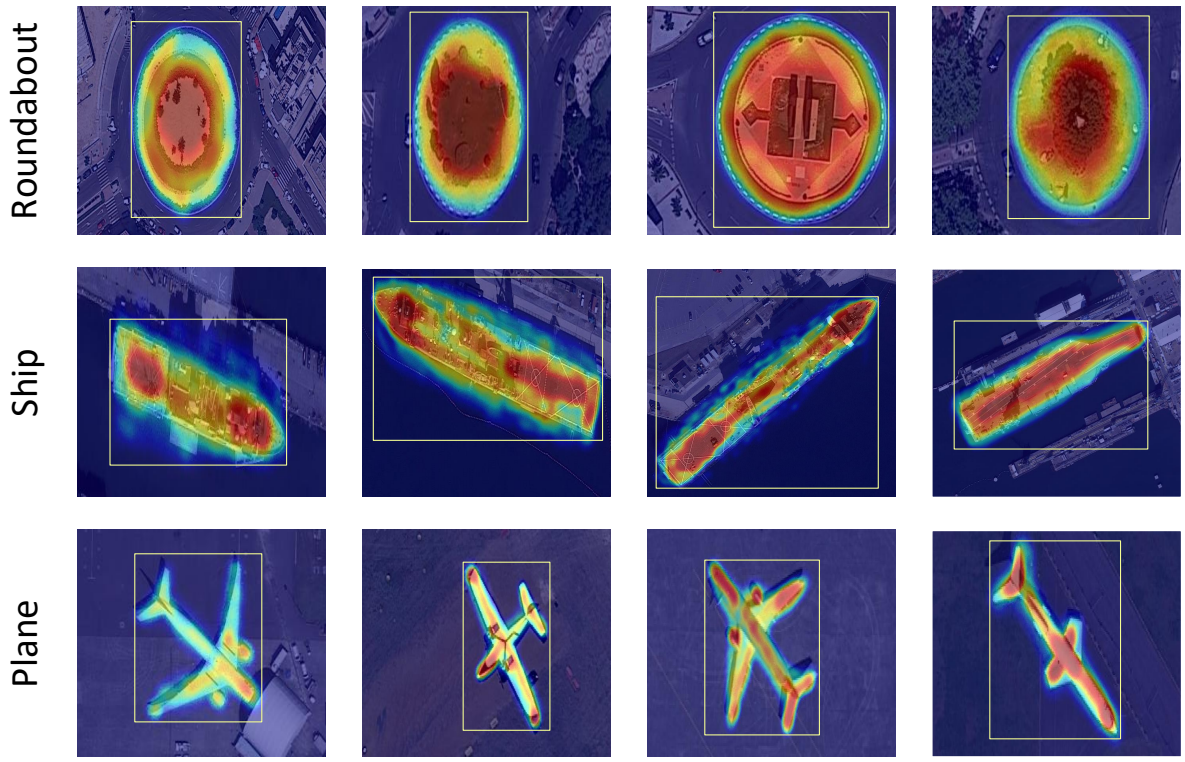

Figure 5. Visualization of the DetCAM. It shows why the detector classifies a specific area as a specific class and covers instance region well.

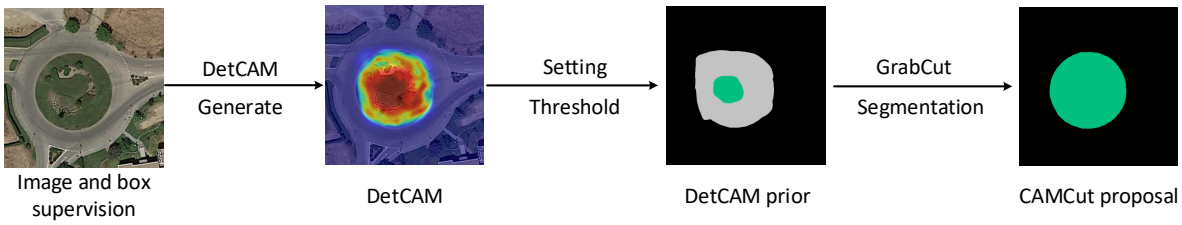

Figure 6. Overview of the CAMCut. We use DetCAM as prior to GrabCut and generate CAMCut. In DetCAM prior, green pixels represent foreground, black pixels represent background, and gray pixels represent uncertainty area. GrabCut takes this information as input and further refines proposal.

$$\mathcal{D} = \sum_{k=1}^{K} \alpha_k M_k \tag{10}$$

As shown in Fig. 5, DetCAM explains why detector classifies a specific area as a specific class and cover instance region well. Based on the observation, we generate high DetCAM proposal and low DetCAM proposal by setting high and low thresholds to DetCAM, as shown in Fig. 4 and Algorithm 1. Low DetCAM proposal $\mathcal{D}_\ell$ is closer to ground truth, and we can use it as the pseudo label to train segmentation model. High DetCAM proposal $\mathcal{D}_h$ can't cover all positive pixels of ground truth but contains less false-positive pixels.

Different from generating visual explanations from classification network, like CAM [14] and Grad-CAM [15], DetCAM generates visual explanations from object detector. Box supervision is fully used, and the detector learns precise semantic information, which improves the proposal quality.

As shown in Fig. 6, we take DetCAM as priors and categorize pixels into thre sets $\mathcal{D}^+$, $\mathcal{D}^-$ and $\mathcal{D}^u$ by

$$\mathcal{D}^+ = \{(i,j)|(i,j) \in \mathcal{D}_h\} \tag{11}$$

$$\mathcal{D}^- = \{(i,j)|(i,j) \notin \mathcal{D}_\ell\} \tag{12}$$

$$\mathcal{D}^u = \{(i,j)|(i,j) \in \mathcal{D}_\ell \cap (i,j) \notin \mathcal{D}_h\} \tag{13}$$

where (i,j) is coordinate. Pixels in $\mathcal{D}^+$ are fixed to the foreground, pixels in $\mathcal{D}^-$ are fixed to background and pixels in $\mathcal{D}^u$ are still uncertain. GrabCut updates proposals by taking these foreground and background information. The updated proposals are denoted as CAMCut proposals. As shown in Fig. 2 (f), CAMCut generates proposals both reliable in the distinguished semantic area and detailed in instance edge.

3. Experiments

In the experiments, we first introduce the experimental setup, then do ablation study of different super parameter, finally compare our method with the state-of-the-art methods.

3.1. Experimental Setup

In experimental setup, we introduce dataset, evaluation method and implementation details of our experiments.

Dataset: We use iSAID [29] dataset to evaluate our method, which is a further semantic labeled version for DOTA [30] dataset. It contains 15 classes different object and 1 background class. The spatial resolution of images ranges from 800 pixels to 13000 pixels, which exceed resolution of natural images by far. We train our method with 1,411 high-resolution images, eval with 458 high-resolution images. We only exploit bounding boxes annotations when training. Although the dataset contains labels for semantic segmentation, we only exploit box-level labels.

Table 1. Evaluating the effectivenesses of JMLNet, including DetCAM proposals, CAMCut proposals and three novel loss functions on iSAID validation set. BOX: rectangle proposals, CRF: CRF proposals, GrabCut: GrabCut proposals.

Loss	Proposals					mIoU
	BOX	CRF	GrabCut	DetCAM	CAMCut	
CE Loss	✓					46.20
		✓				51.27
			✓			53.12
				✓		53.88
					✓	**54.24**
MA-Loss	✓				✓	52.27
		✓			✓	52.64
			✓		✓	53.58
				✓	✓	54.45
			✓	✓	✓	**54.61**
		✓	✓	✓	✓	54.21
	✓	✓	✓	✓	✓	53.72
MM-Loss	✓				✓	52.45
		✓			✓	52.83
			✓		✓	54.25
				✓	✓	54.64
			✓	✓	✓	**54.97**
		✓	✓	✓	✓	54.63
	✓	✓	✓	✓	✓	53.94
BMM-Loss	✓				✓	53.22
		✓			✓	53.41
			✓		✓	54.67
				✓	✓	55.10
			✓	✓	✓	**55.34**
		✓	✓	✓	✓	54.85
	✓	✓	✓	✓	✓	54.05

Evaluation: To evaluate the performance of our method and compare our results to other state-of-the-art methods, we calculate mean pixel Intersection-over-Union(mIoU) as common practice [5,31]. IoU is defined as:

$$IoU = \frac{TP}{TP + FP + FN} \tag{14}$$

and mIoU is defined as:

$$mIoU = \frac{1}{C} \sum_{c=1}^{C} \frac{TP}{TP + FP + FN} \tag{15}$$

where TP,FP,FN are the number of true positives, false positives and false negatives. C indicates the number of classes.

Implementation Details: We crop the high-resolution images to 512x512 pathes. We adopt the classical Deeplab v3 [28] model for our experiments, which takes widely used ResNet-50 [27] as backbone. Firstly, we train a detection model Faster-RCNN [26] with box-level labels of iSAID [29]. Using the proposed DetCAM and CAMCut methods, we generate pseudo segmentation proposals for train set. Secondly, we train the Deeplab v3 model with the CAMCut supervision for 50k iterations, further finetune it with proposed loss function for 10k iterations. We choose SGD as default optimizer. Mini-batch size is seted to 20. We set initial learning rate to 0.007 and multiply by $(1 - \frac{step}{max_step})^{power}$ and $power$ is set to 0.9. We apply random horizontal flipping and random cropping to augment diversity of dataset. We implement our method with the pytorch [32] framework.

Table 2. Influence of τ_ℓ. The hyper-parameter τ_ℓ balances the foreground and background pixels when generating low DetCAM proposals. $\tau_\ell = 0$ means all pixels within boxes annotations are seen as proposals.

τ_ℓ	0	0.05	0.1	0.15	0.2	0.25
mIoU (%)	44.20	51.34	53.46	**53.88**	53.76	53.20

Table 3. Influence of τ_h. The hyper-parameter τ_h influences quality of CAMCut. $\tau_h = 1$ means no positive foreground for GrabCut proposals.

τ_h	0.5	0.6	0.7	0.8	0.9	1
mIoU (%)	53.30	53.58	54.10	**54.24**	54.23	53.67

3.2. Ablation study

We conduct two types of ablation studies, including the analysis of the contribution of proposed loss functions and the performance of the proposal with different threshold.

Proposals quality and losses selection. We do experiments on different proposals and loss functions. As shown in Table 1, experimental results show that our proposed DetCAM and CAMCut proposals perform better than traditional proposals. We train the Deeplab v3 model with different proposals as pseudo labels, including rectangle proposals, CRF proposals, GrabCut proposals, our proposed DetCAM proposals and CAMCut proposals. As shown in Table 1, our proposed DetCAM and CAMCut proposals achieve 53.88% and 54.24% mIoU, outperforming all the compared methods. Experimental results also show that our proposed MA-Loss, MM-Loss and BMM-Loss all improve segmentation results, in which BMM-Loss performs best. We combine different proposals together and use our proposed loss functions to train the Deeplab v3 model. As shown in Table 1, using combination of different proposals and our proposed loss functions, we improve segmentation results significantly. We notice that segmentation performance will not improved by adding rectangle proposals and CRF proposals to $\{GrabCut,DetCAM,CAMCut\}$. Because compared with $\{GrabCut,DetCAM,CAMCut\}$, rectangle proposals and CRF proposals are quite rough and introduce more wrongly labeled pixels. In particular, combination of $\{GrabCut,DetCAM,CAMCut\}$ and BMM-Loss achieve the best performance, 55.34% mIoU. We analyze that the reason why BMM-Loss performs best is BMM-Loss considers the similarity between predictions and multiple proposals in pixel-wise within boxes. The other loss functions, MA-Loss and MM-Loss, only focus on loss of whole image.

Threshold τ_ℓ of low DetCAM proposals $\mathcal{D}_\ell$. Low DetCAM proposals $\mathcal{D}_\ell$ depends on one key hyper-parameter, threshold τ_ℓ. We use $\mathcal{D}_\ell$ as pseudo label to train segmentation model, which is vital to final performance. The threshold τ_ℓ balances the foreground and background pixels within boxes annotations. If τ_ℓ is set to 0, all pixels within boxes annotations are seen as proposals. As τ_ℓ increases, the area of proposals decreases and only the distinguished part of DetCAM remained in proposals. Table 2 shows the influence of threshold τ_ℓ. As τ_ℓ get higher, the area of foreground pixels get lower. Because foreground pixels usually take up most area within boxes annotations, so we find best τ_ℓ in

Table 4. Weakly supervised results on iSAID validation set.

Supervision	Methods	mIoU(%)
Weak	SDI [13]	53.82
	Song *et al*. [10]	54.18
	Ours	**55.34**
Semi	SDI [13]	54.87
	Song *et al*. [10]	55.15
	Ours	**56.76**
Full	Deeplab v3 [33]	59.05

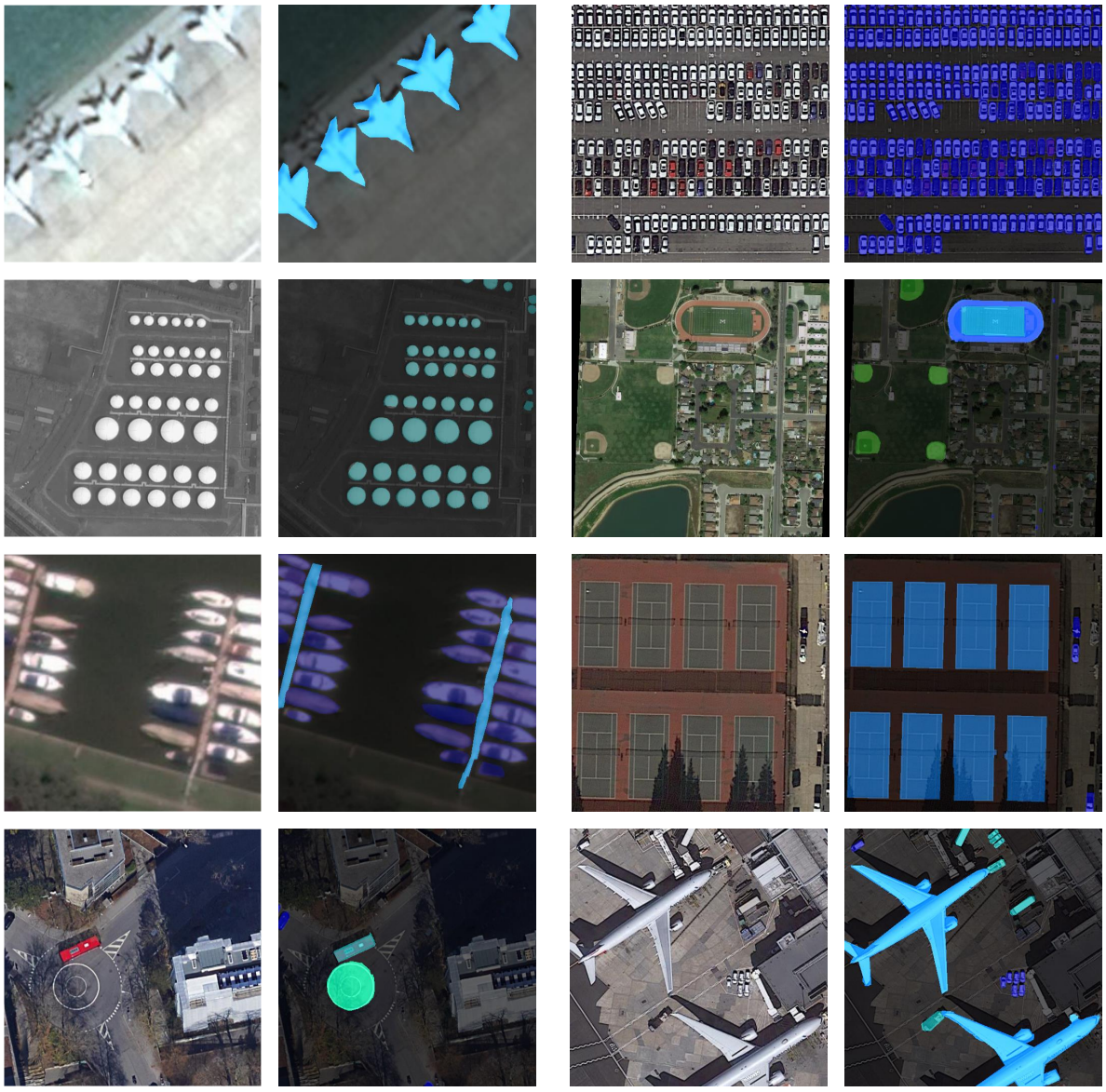

Figure 7. Examples of segmentation results of our method on iSAID. (a)Image. (b)Prediction.

Table 5. Our segmentation results for per category on iSAID validation set, which are evaluated by mIoU(%). ST: storage tank, BD: baseball diamond, TC: tennis court, BC: basketball court, GTF: ground field track, LV: large vehicle, SV: small vehicle, HC: helicopter, SP: swimmingpool, RA: roundabout,SBF: soccerballfield.

Supervision	Ship	ST	BD	TC	BC	GTF	Bridge	LV	SV	HC	SP	RA	SBF	Plane	Harbor	**mean**
Weak	55.36	47.98	73.10	78.81	55.32	56.15	28.22	51.76	28.57	27.05	41.37	62.74	68.84	69.18	42.94	55.34
Semi	56.85	49.62	74.62	80.64	56.56	57.99	29.61	53.10	30.44	28.51	43.26	64.80	70.10	70.12	44.09	56.76
Full	59.74	50.49	76.98	84.21	57.92	59.57	32.88	54.80	33.75	31.29	44.74	66.03	72.13	75.84	45.68	59.05

small values. When $\tau_\ell = 0.15$, using $\mathcal{D}_\ell$ proposals as ground truth, we achieve the best performance. Table 1 indicate that $\mathcal{D}_\ell$ reachs 53.88% mIoU on iSAID validation set. We also fix $\tau_\ell = 0.15$ in generating CAMCut proposals.

Threshold τ_h of high DetCAM proposals $\mathcal{D}_h$. High DetCAM proposals $\mathcal{D}_h$ depends on threshold τ_h. We use $\mathcal{D}_h$ as foreground to adjust GrabCut algorithm and generate CAMCut. Table 3 shows the influence of threshold τ_h. When $\tau_h = 0.8$, CAMCut achieves the best performance. Table 1 indicates that CAMCut reachs 54.24% mIoU in iSAID validation set.

3.3. Comparison with the State-of-the-art Methods

In the comparison with the state-of-the-art methods, we mainly choose SDI [13] and Song *et al*. [10].

Results of Weakly-supervised Semantic Segmentation. As shown in Table 4, our method achieves 55.34% mIoU on iSAID validation set. Specific IOU for per category can be found at Table 5. We compare with SDI [13] and Song *et al*. [10] from two aspects of view. Our method outperforms all compared weakly supervised semantic segmentation approaches. Fig. 7 shows the segmentation results of our method. The results indicate that our proposed method is effective when learning common knowledge from multiple noisy labels.

Results of Semi-supervised Semantic Segmentation. We also do semi-supervised semantic segmentation experiments and compare to state-of-the-art approaches. In semi-supervised task, 141 pixel-level labels, 1/10 of the training sets, are added for training. As shown in Table 4, our proposed method outperforms all the compared methods and achieves 56.76% mIoU. Specific IOU for per category can be found at Table 5. The results indicate that our method is still effective in semi-supervised condition and the performance is very close to the fully supervised model.

4. Conclusion

In this paper, we propose a novel JMLNet, which first regards multiple proposals as multi-label supervision to train weakly supervised semantic segmentation model. JMLNet learns common knowledge from multiple noisy labels and prevent the model from overfitting one specific label. Two segmentation proposals, DetCAM and CAMCut, are proposed to further improve the segmentation performance of JMLNet. JMLNet achieves the state-of-the-art results on the iSAID dataset.

Author Contributions: Conceptualization, R.G., Investigation, R.G.; Formal analysis, R.G., Methodology, R.G.; Supervision, X.S., K.C., X.Z., Z.Y., W.D. and M.Y.; Visualization, R.G.; Writing—original draft, R.G.; Writing—review and editing, X.S., K.C., X.Z., Z.Y., W.D. and M.Y.

Funding: The work was supported by the National Natural Science Foundation of China under Grants 41701508 and 61725105.

Acknowledgments: The authors would like to thank the anonymous reviewers for their very competent comments and helpful suggestions.

Conflicts of Interest: The authors declare no conflict of interest.

1. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 3431–3440.
2. Chen, L.C.; Zhu, Y.; Papandreou, G.; Schroff, F.; Adam, H. Encoder-decoder with atrous separable convolution for semantic image segmentation. Proceedings of the European conference on computer vision (ECCV), 2018, pp. 801–818.
3. Zhang, W.; Huang, H.; Schmitz, M.; Sun, X.; Wang, H.; Mayer, H. Effective fusion of multi-modal remote sensing data in a fully convolutional network for semantic labeling. *Remote Sensing* **2018**, *10*, 52.
4. Fu, K.; Lu, W.; Diao, W.; Yan, M.; Sun, H.; Zhang, Y.; Sun, X. WSF-NET: Weakly supervised feature-fusion network for binary segmentation in remote sensing image. *Remote Sensing* **2018**, *10*, 1970.

5. Lin, T.Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft coco: Common objects in context. European conference on computer vision. Springer, 2014, pp. 740–755.
6. Dai, J.; He, K.; Sun, J. Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation. Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 1635–1643.
7. 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. Proceedings of the IEEE international conference on computer vision, 2015, pp. 1742–1750.
8. Krähenbühl, P.; Koltun, V. Efficient inference in fully connected crfs with gaussian edge potentials. Advances in neural information processing systems, 2011, pp. 109–117.
9. Bearman, A.; Russakovsky, O.; Ferrari, V.; Fei-Fei, L. What's the point: Semantic segmentation with point supervision. European conference on computer vision. Springer, 2016, pp. 549–565.
10. Song, C.; Huang, Y.; Ouyang, W.; Wang, L. Box-driven class-wise region masking and filling rate guided loss for weakly supervised semantic segmentation. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 3136–3145.
11. Pont-Tuset, J.; Arbelaez, P.; Barron, J.T.; Marques, F.; Malik, J. Multiscale combinatorial grouping for image segmentation and object proposal generation. *IEEE transactions on pattern analysis and machine intelligence* **2016**, *39*, 128–140.
12. Rother, C.; Kolmogorov, V.; Blake, A. Grabcut: Interactive foreground extraction using iterated graph cuts. ACM transactions on graphics (TOG). ACM, 2004, Vol. 23, pp. 309–314.
13. Khoreva, A.; Benenson, R.; Hosang, J.; Hein, M.; Schiele, B. Simple does it: Weakly supervised instance and semantic segmentation. Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 876–885.
14. Zhou, B.; Khosla, A.; Lapedriza, A.; Oliva, A.; Torralba, A. Learning deep features for discriminative localization. Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 2921–2929.
15. Selvaraju, R.R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; Batra, D. Grad-cam: Visual explanations from deep networks via gradient-based localization. Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 618–626.
16. Huang, Z.; Wang, X.; Wang, J.; Liu, W.; Wang, J. Weakly-supervised semantic segmentation network with deep seeded region growing. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 7014–7023.
17. Kolesnikov, A.; Lampert, C.H. Seed, expand and constrain: Three principles for weakly-supervised image segmentation. European Conference on Computer Vision. Springer, 2016, pp. 695–711.
18. Ahn, J.; Kwak, S. Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 4981–4990.
19. Zhou, Y.; Zhu, Y.; Ye, Q.; Qiu, Q.; Jiao, J. Weakly supervised instance segmentation using class peak response. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 3791–3800.
20. Lin, D.; Dai, J.; Jia, J.; He, K.; Sun, J. Scribblesup: Scribble-supervised convolutional networks for semantic segmentation. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 3159–3167.
21. Tang, M.; Perazzi, F.; Djelouah, A.; Ben Ayed, I.; Schroers, C.; Boykov, Y. On regularized losses for weakly-supervised cnn segmentation. Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 507–522.
22. Chen, J.; He, F.; Zhang, Y.; Sun, G.; Deng, M. SPMF-Net: Weakly Supervised Building Segmentation by Combining Superpixel Pooling and Multi-Scale Feature Fusion. *Remote Sensing* **2020**, *12*, 1049.
23. Wang, S.; Chen, W.; Xie, S.M.; Azzari, G.; Lobell, D.B. Weakly supervised deep learning for segmentation of remote sensing imagery. *Remote Sensing* **2020**, *12*, 207.
24. Wu, W.; Qi, H.; Rong, Z.; Liu, L.; Su, H. Scribble-Supervised Segmentation of Aerial Building Footprints Using Adversarial Learning. *IEEE Access* **2018**, *6*, 58898–58911.
25. Rafique, M.U.; Jacobs, N. Weakly Supervised Building Segmentation from Aerial Images. IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium, 2019, pp. 3955–3958.

26. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. Advances in neural information processing systems, 2015, pp. 91–99.
27. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.
28. Chen, L.C.; Yang, Y.; Wang, J.; Xu, W.; Yuille, A.L. Attention to scale: Scale-aware semantic image segmentation. Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 3640–3649.
29. Waqas Zamir, S.; Arora, A.; Gupta, A.; Khan, S.; Sun, G.; Shahbaz Khan, F.; Zhu, F.; Shao, L.; Xia, G.S.; Bai, X. isaid: A large-scale dataset for instance segmentation in aerial images. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2019, pp. 28–37.
30. Xia, G.S.; Bai, X.; Ding, J.; Zhu, Z.; Belongie, S.; Luo, J.; Datcu, M.; Pelillo, M.; Zhang, L. DOTA: A Large-Scale Dataset for Object Detection in Aerial Images. The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018.
31. Everingham, M.; Eslami, S.A.; Van Gool, L.; Williams, C.K.; Winn, J.; Zisserman, A. The pascal visual object classes challenge: A retrospective. *International journal of computer vision* **2015**, *111*, 98–136.
32. Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; others. PyTorch: An imperative style, high-performance deep learning library. Advances in Neural Information Processing Systems, 2019, pp. 8024–8035.
33. 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 transactions on pattern analysis and machine intelligence* **2017**, *40*, 834–848.