南京航空航天大學

Nanjing University of Aeronautics and Astronautics

模式分析与机器智能
工业和信息化部重点实验室

MIIT Key Laboratory of
Pattern Analysis & Machine Intelligence

Semi-Supervised Semantic Segmentation under Label Noise via Diverse Learning Groups

ICCV 2023

Contributions

1、present a general architecture that maintains two diverse learning groups to overcome confirmation bias in label assignment problems.

2、introduce a new filter module suitable for semantic segmentation to detect mislabeled pixels, while demonstrating a good trade-off between accuracy and computational cost.

Li, Peixia, Pulak Purkait, Thalaiyasingam Ajanthan, Majid Abdolshah, Ravi Garg, Hisham Husain, Chenchen Xu, Stephen Gould, Wanli Ouyang, and Anton van den Hengel. "Semi-Supervised Semantic Segmentation under Label Noise via Diverse Learning Groups." In Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 1229-1238. 2023.

Method : Problem Setup

labelled set $D_L = \{(x_i, y_i)\}_{i=1}^{N_L}$

unlabelled set $D_U = \{(x_i)\}_{i=1}^{N_U}$

We consider semi-supervised segmentation in the case where we have a small set of potentially noisy labelled examples and a large set of unlabelled examples. Let $\mathcal{D}_L = \{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^{N_L}$ be the labelled set where $\boldsymbol{x}_i \in \mathcal{X} := [0,1]^{h \times w \times 3}$ and $\boldsymbol{y}_i \in \mathcal{Y} := \mathcal{C}^{h \times w}$ denote an image and the corresponding pixel-wise segmentation mask respectively. Here, $\mathcal{C} = \{1, 2, \ldots, \ell\}$ denotes the set of labels. Similarly, let $\mathcal{D}_U = \{\boldsymbol{x}_i\}_{i=1}^{N_U}$ denote the unlabelled set of images.

Method

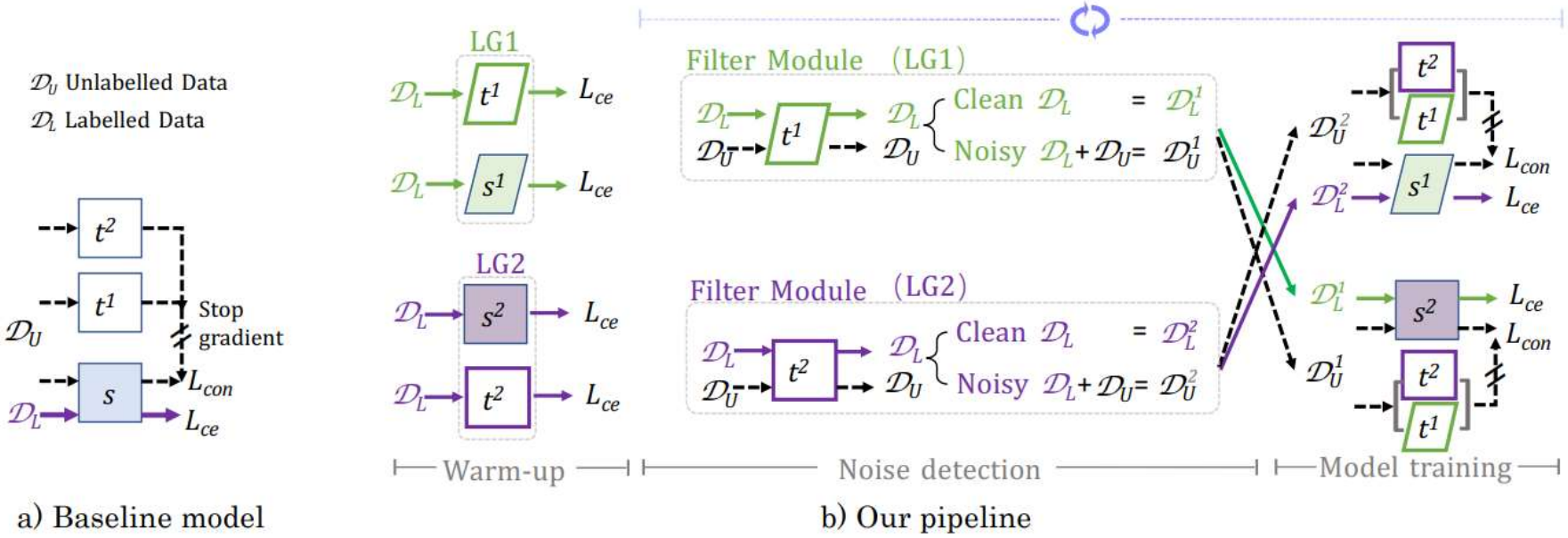


Figure 2: The pipelines of a) the baseline model [20] and b) the proposed approach. In contrast to the baseline that uses two teachers and a single student network, we propose a framework with two diverse Learning Groups (LGs) (t^1, s^1) and (t^2, s^2) where each group contains a teacher and a student network. The two learning groups propagate complementary information and reduce the confirmation bias. Here, t denotes a teacher model, s is a student model, solid lines show the forward flow of labelled data, and dotted lines show the flow of unlabelled data. L_{ce} denotes the cross-entropy loss and L_{con} is the confidence-weighted cross-entropy loss.

Liu, Yuyuan, Yu Tian, Yuanhong Chen, Fengbei Liu, Vasileios Belagiannis, and Gustavo Carneiro. "Perturbed and strict mean teachers for semi-supervised semantic segmentation." In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4258-4267. 2022.

Method : Filter Module

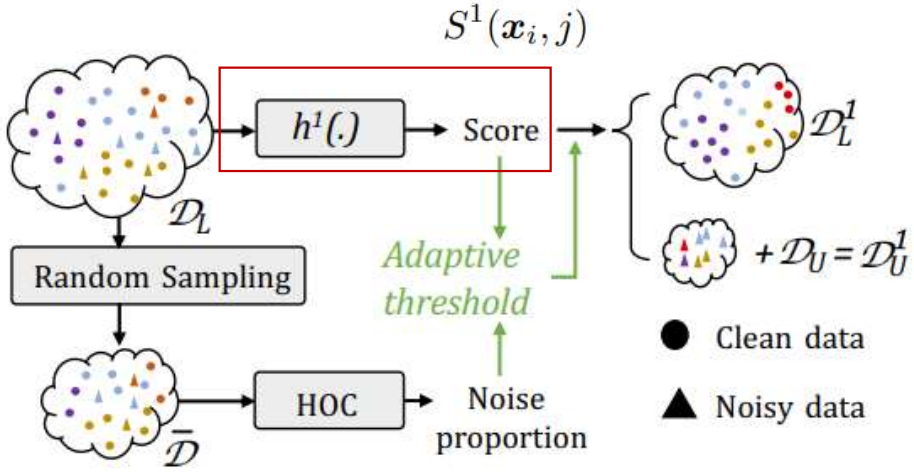


Figure 4: The architecture of our filter module. First, we generate a representative dataset $\bar{\mathcal{D}}$ and calculate the noise proportion based on HOC [39]. Then, we get the adaptive threshold based on the noise proportion and example score. According to the adaptive threshold and example score, the original dataset $\mathcal{D}_L$ is divided into a clean set $\mathcal{D}_L^1$ and a noisy set. The noisy set is added to the unlabelled dataset $\mathcal{D}_U$.

Let $h_j^1(\cdot) : \mathcal{X} \to [0,1]^\ell$ and $h_j^2(\cdot) : \mathcal{X} \to [0,1]^\ell$ be the teacher model mappings to softmax scores in the first and second LG respectively. We compute a noisiness score based on the teacher model predictions, as opposed to using

$$S^1(\boldsymbol{x}_i, j) = \mathrm{CE}(h_j^1(\boldsymbol{x}_i), \boldsymbol{e}_{i,j}) + \lambda\, \mathrm{KL}(h_j^1(\boldsymbol{x}_i) \| h_j^2(\boldsymbol{x}_i))$$

Method : Filter Module

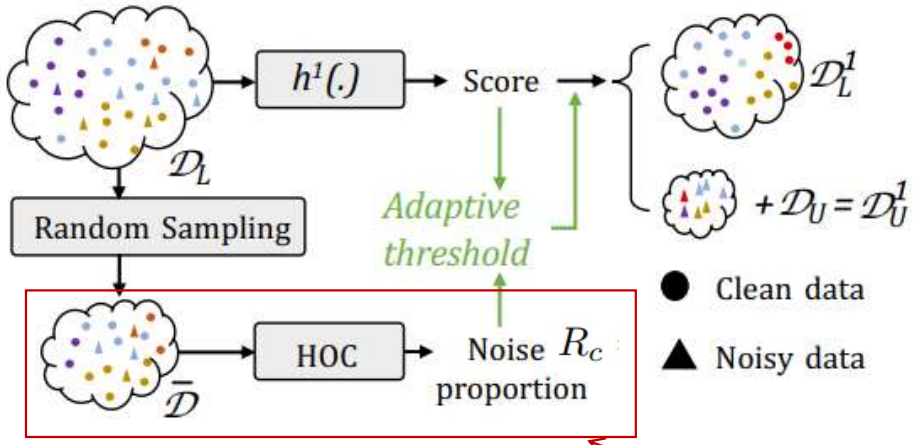


Figure 4: The architecture of our filter module. First, we generate a representative dataset $\bar{\mathcal{D}}$ and calculate the noise proportion based on HOC [39]. Then, we get the adaptive threshold based on the noise proportion and example score. According to the adaptive threshold and example score, the original dataset $\mathcal{D}_L$ is divided into a clean set $\mathcal{D}_L^1$ and a noisy set. The noisy set is added to the unlabelled dataset $\mathcal{D}_U$.

$$\bar{D} = \{(x_i, j, \tilde{y}_{i,j})\}$$

Let $g_j^1(\cdot) : \mathcal{X} \to \mathbb{R}^d$ denotes the backbone feature extractor, *i.e.*, the penultimate layer output for the first LG at pixel j (where $j \in \mathcal{I}(\boldsymbol{x}_i) := \{1, \ldots, w \times h\}$ if images are resized to a fixed size). We construct the embedded representative set $\bar{\mathcal{D}}_G^1 := \left\{(g_j^1(\boldsymbol{x}_i), \tilde{y}_{i,j}) \mid (\boldsymbol{x}_i, j, \tilde{y}_{i,j}) \in \bar{\mathcal{D}}\right\}$ and invoke HOC (Eq. (1)) to estimate the noise proportion $\{R_c^1\}_{c \in \mathcal{C}}$ at pixel-level (Eq. (2)). Analogously, the noise proportion $\{R_c^2\}_{c \in \mathcal{C}}$ for the other LG can be estimated by constructing $\bar{\mathcal{D}}_G^2$.

$$\mathbb{P}(Y), \mathbb{P}(\tilde{Y}|Y) \leftarrow \text{HOC}\left(\{(g(\boldsymbol{x}), \tilde{y}) \mid (\boldsymbol{x}, \tilde{y}) \in \tilde{\mathcal{D}}\}\right), \quad (1)$$

$$R_c = 1 - \mathbb{P}(Y = c|\tilde{Y} = c) .$$

$$\mathbb{P}(Y|\tilde{Y}) = \mathbb{P}(\tilde{Y}|Y)\mathbb{P}(Y)/\mathbb{P}(\tilde{Y})$$

Zhaowei Zhu, Yiwen Song, and Yang Liu. Clusterability as an alternative to anchor points when learning with noisy labels. In International Conference on Machine Learning, pages 12912–12923. PMLR, 2021.

Method : Filter Module

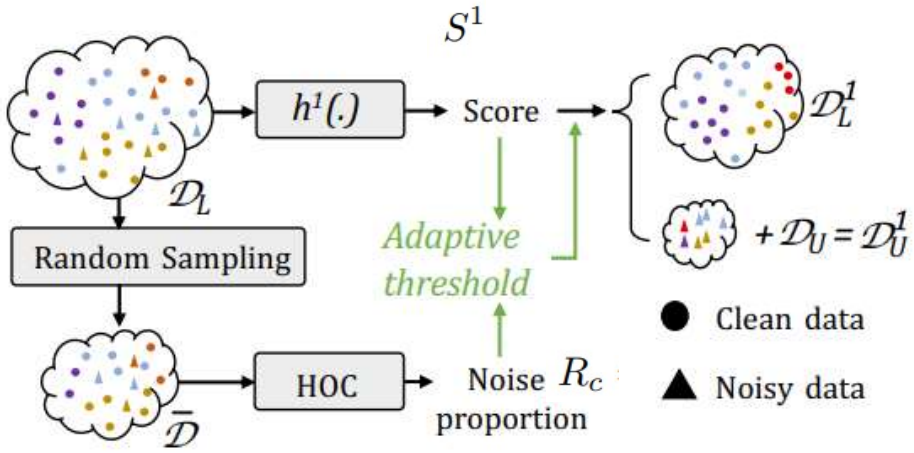


Figure 4: The architecture of our filter module. First, we generate a representative dataset $\bar{\mathcal{D}}$ and calculate the noise proportion based on HOC [39]. Then, we get the adaptive threshold based on the noise proportion and example score. According to the adaptive threshold and example score, the original dataset $\mathcal{D}_L$ is divided into a clean set $\mathcal{D}_L^1$ and a noisy set. The noisy set is added to the unlabelled dataset $\mathcal{D}_U$.

Once the noisiness score and the noise proportion have been calculated, we get the adaptive noise threshold as follows:

$$\beta^1 = \frac{\sigma \ell S^1_{avg1}}{\sum_{c \in \mathcal{C}} R^1_c}, \text{ where } S^1_{avg1} = n^{-1} \sum_{\substack{(\boldsymbol{x}_i, \cdot) \in \mathcal{D}_L, \\ j \in \mathcal{I}(\boldsymbol{x}_i)}} S^1(\boldsymbol{x}_i, j),$$

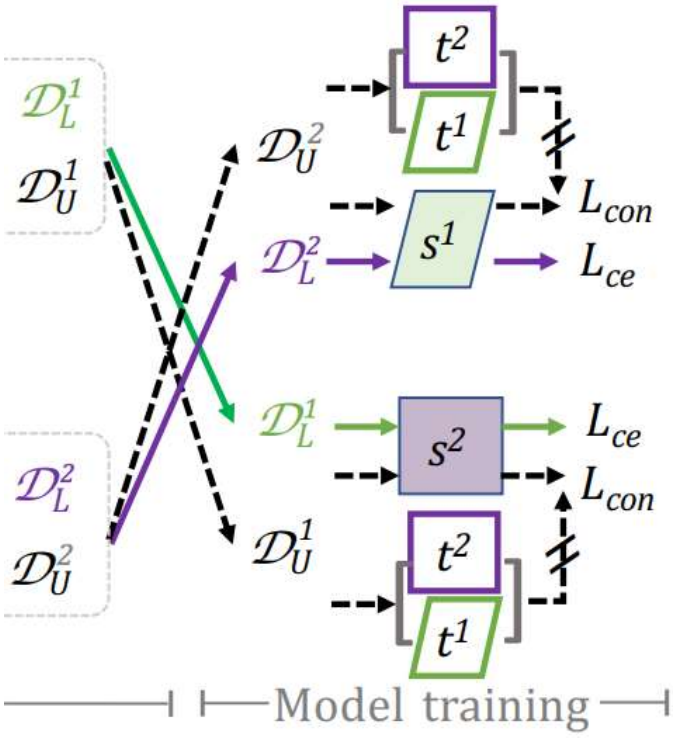
Let us denote the set of pixels that are detected as label noise by the first LG as:

$$\mathcal{M}^1_k := \left\{ (\boldsymbol{x}_i, j) \mid S^1(\boldsymbol{x}_i, j) > \beta^1, (\boldsymbol{x}_i, \cdot) \in \mathcal{D}_L, j \in \mathcal{I}(\boldsymbol{x}_i) \right\} \quad (7)$$

$$\mathcal{D}^1_{L,k} := \mathcal{D}_L \setminus \mathcal{M}^{12}_k$$

$$\mathcal{D}^1_{U,k} := \mathcal{D}_U \bigcup \left\{ \boldsymbol{x}_i \mid (\boldsymbol{x}_i, \cdot) \in \mathcal{M}^1_k \right\}.$$

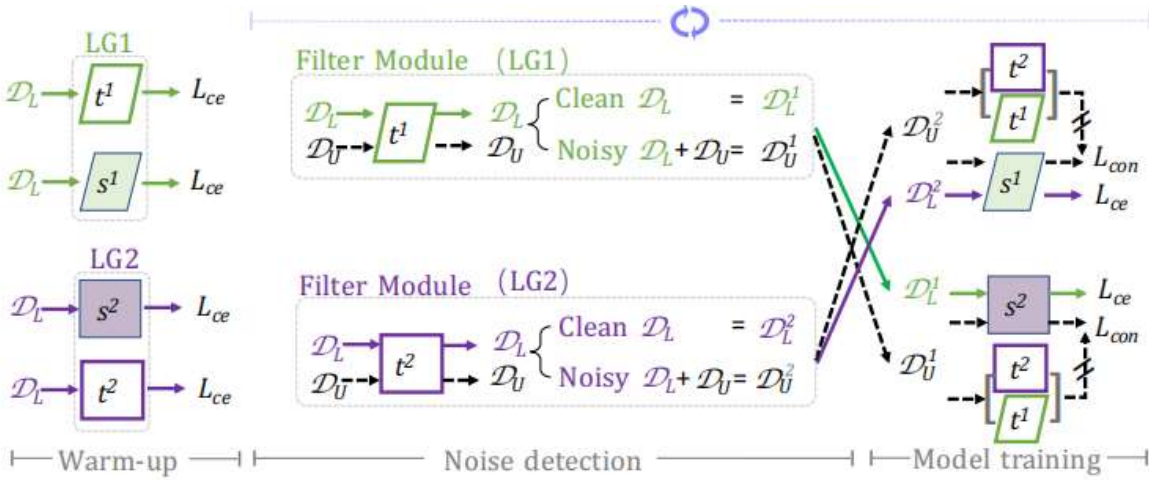
Method : Model training



$$\mathcal{L}\ (D_{L,k}^2, D_{U,k}^2, \theta_{s^1}) := L_{ce}(D_{L,k}^2, \theta_{s^1}) + \alpha L_{con}(D_{U,k}^2, \theta_{s^1})$$

$$\mathcal{L}\ (D_{L,k}^1, D_{U,k}^1, \theta_{s^2}) := L_{ce}(D_{L,k}^1, \theta_{s^2}) + \alpha L_{con}(D_{U,k}^1, \theta_{s^2})$$

Method



b) Our pipeline

$$S^1(\boldsymbol{x}_i, j) = \mathrm{CE}(h_j^1(\boldsymbol{x}_i), \boldsymbol{e}_{i,j}) + \lambda\, \mathrm{KL}(h_j^1(\boldsymbol{x}_i) \| h_j^2(\boldsymbol{x}_i)) \quad (5)$$

$$\beta^1 = \frac{\sigma \ell S^1_{avg1}}{\sum_{c \in \mathcal{C}} R^1_c}\,, \text{ where } S^1_{avg1} = n^{-1} \sum_{\substack{(\boldsymbol{x}_i, \cdot) \in \mathcal{D}_L, \\ j \in \mathcal{I}(\boldsymbol{x}_i)}} S^1(\boldsymbol{x}_i, j)\,, \quad (6)$$

$$\mathcal{M}^1_k := \{(\boldsymbol{x}_i, j) \mid S^1(\boldsymbol{x}_i, j) > \beta^1, (\boldsymbol{x}_i, \cdot) \in \mathcal{D}_L, j \in \mathcal{I}(\boldsymbol{x}_i)\} \quad (7)$$

$$\mathcal{L}(\mathcal{D}^2_{L,k}, \mathcal{D}^2_{U,k}, \boldsymbol{\theta}_{s^1}) := L_{ce}(\mathcal{D}^2_{L,k}, \boldsymbol{\theta}_{s^1}) + \alpha\, L_{con}(\mathcal{D}^2_{U,k}, \boldsymbol{\theta}_{s^1}) \quad (4)$$

Algorithm 1 Our training pipeline

Input: $\mathcal{D}_L$, $\mathcal{D}_U$, total epochs E, warm-up epochs N
Output: Optimized $\{\boldsymbol{\theta}_{s^1}, \boldsymbol{\theta}_{t^1}, \boldsymbol{\theta}_{s^2}, \boldsymbol{\theta}_{t^2}\}$

Initialize parameters $\{\boldsymbol{\theta}_{s^1}, \boldsymbol{\theta}_{t^1}, \boldsymbol{\theta}_{s^2}, \boldsymbol{\theta}_{t^2}\}$,
for $k \in \{0, 1, \ldots, E\}$ **do**
  if $k < N$ **then**
    $\mathcal{D}^1_{L,k}, \mathcal{D}^2_{L,k} := \mathcal{D}_L$ and $\mathcal{D}^1_{U,k}, \mathcal{D}^2_{U,k} := \emptyset$
  else
    $\{(\boldsymbol{x}_i, j, \tilde{y}_{i,j})\} \leftarrow$ Sample from $\mathcal{D}_L$
    for $l \in \{1, 2\}$ **do**
      $\{R^l_c\} \leftarrow \mathrm{HOC}\left(\{(g^l_j(\boldsymbol{x}_i), \tilde{y}_{i,j})\}\right)$
      Scores $\{S^l\} \leftarrow$ Using Eq. (5)
      $\beta^l \leftarrow$ Adaptive $\left(\{S^l\}\right)$ [Eq. (6)]
      $\mathcal{M}^l_k \leftarrow$ Using Eq. (7)
      $\mathcal{D}^l_{L,k} \leftarrow \mathcal{D}_L \setminus \mathcal{M}^l_k$
      $\mathcal{D}^l_{U,k} \leftarrow \mathcal{D}_U \bigcup \{\boldsymbol{x}_i \mid (\boldsymbol{x}_i, \cdot) \in \mathcal{M}^l_k\}$
  repeat
    $\boldsymbol{\theta}_{s^1} \leftarrow$ Backprop. $\mathcal{L}(\mathcal{D}^2_{L,k}, \mathcal{D}^2_{U,k}, \boldsymbol{\theta}_{s^1})$ [Eq. (4)]
    $\boldsymbol{\theta}_{s^2} \leftarrow$ Backprop. $\mathcal{L}(\mathcal{D}^1_{L,k}, \mathcal{D}^1_{U,k}, \boldsymbol{\theta}_{s^2})$ [Eq. (4)]
  until M times
  Update $\boldsymbol{\theta}_{t^1}, \boldsymbol{\theta}_{t^2} \leftarrow$ Moving Average of $\boldsymbol{\theta}_{s^1}, \boldsymbol{\theta}_{s^2}$.

Experiments

1、 Semi-supervised Semantic Segmentation

Table 1: Comparison against SotA approaches on Pascal VOC 2012 in a semi-supervised semantic segmentation setting. All baselines are based on the DeeplabV3+ architecture. The * indicates results reported by [6]. When we replace 50% of the ground truth of images with 'PL', the proportion of noisy pixels is 9%. We keep the same pixel level noise proportion while introducing noise using 'RDE' and 'SCP' for parity.

Methods	Year	Noisy pixel	Noise type	Labelled Data Ratio			
				1/16 (662)	1/8(1323)	1/4 (2646)	1/2 (5291)
MT* [30]	2017	0%	None	66.70	70.78	73.22	75.41
French [8]	2019	0%	None	68.90	70.70	72.46	74.49
CCT* [26]	2020	0%	None	65.22	70.87	73.43	74.75
GCT* [14]	2020	0%	None	64.05	70.47	73.45	75.20
ECS* [23]	2020	0%	None	-	67.38	70.70	72.89
CPS* [6]	2021	0%	None	71.98	73.67	74.90	76.15
CAC* [16]	2021	0%	None	70.10	72.40	74.00	-
PS-MT [20]	2022	0%	None	72.83	75.70	76.43	77.88
Ours	-	0%	None	77.75 (+4.92)	79.31 (+3.61)	79.14 (+2.71)	79.54 (+1.66)
PS-MT [20]	-	9%	PL	61.90	65.14	65.53	66.78
Ours	-	9%	PL	71.60 (+9.70)	73.95 (+8.81)	74.53 (+9.00)	74.44 (+7.66)
PS-MT [20]	-	9%	RDE	58.87	62.18	63.81	63.46
Ours	-	9%	RDE	66.82 (+7.95)	69.95 (+7.77)	73.82 (+10.01)	74.47 (+11.01)
PS-MT [20]	-	9%	SCP	52.14	57.19	56.81	53.69
Ours	-	9%	SCP	67.80 (+15.66)	70.83 (+13.64)	69.38 (+12.57)	75.95 (+22.26)

Pseudo Labels (PL)

Random Dilation and Erosion (RDE)

Similar Class Perturbation (SCP)

Experiments

2、 Weakly-supervised Semantic Segmentation

Pseudo Labels (PL)

Table 2: The mIoU of the weakly-supervised semantic segmentation baselines and ours on the test set of SegTHOR. Best Val and Last Epoch correspond to the checkpoint that performs best on the validation set and the last checkpoint.

Model	Best Val	Last Epoch
Base-ADELE [19]	62.6	59.1
ADELE [19]	71.1	70.8
Ours	**73.2**	**73.5**

Random Dilation and Erosion (RDE)

Table 3: Comparison with state-of-the-art weakly supervised semantic segmentation methods on the Pascal VOC 2012 dataset using mIoU (%) of the validation set.

Model	Year	mIoU (%)
AffinityNet [1]	2018	61.7
SSDD [27]	2019	64.9
SEAM [34]	2020	64.5
CONTA [37]	2020	66.1
SPML [13]	2021	69.5
ADELE [19] + SEAM	2022	69.3
PS-MT [20] (R50)	2022	64.0
PS-MT [20] (PVTv2-B2)	2022	66.7
Ours	-	**71.7**

Sheng Liu, Kangning Liu, Weicheng Zhu, Yiqiu Shen, and Carlos Fernandez-Granda. Adaptive early-learning correc_x0002_tion for segmentation from noisy annotations. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 2606–2616, 2022.

Ablation Study

All experiments in this section are conducted with 1/8 labelled data ratio and 9% PL noisy annotations on Pascal VOC.

Table 4: Ablation study of FM and LG on Pascal VOC.

Method	2 LGs	Filter Module (FM)	mIoU (%)
PS-MT [20]			65.1
PS-MT + FM		✓	67.6 (+2.5)
Ours − FM	✓		69.9 (+4.8)
Ours	✓	✓	74.0 (+8.9)

Ablation Study

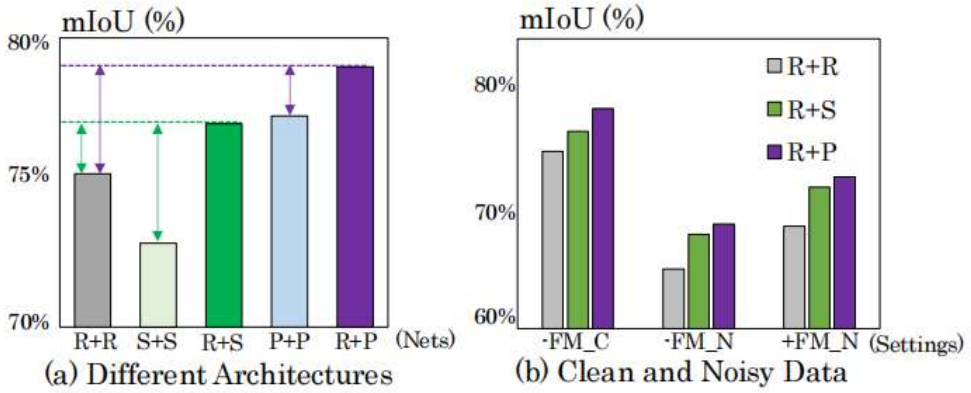


Figure 7: (a) shows the mIoU score of different network combinations on clean data. (b) shows the mIoU score of different combinations on different settings. R, S and P denote R50 [9], Segformer [35] and PVTv2-B2 [33] respectively. -FM_C denotes training our model (without the filter module) with clean data. -FM_N denotes training model with noisy data without using filter module. +FM_N is our complete model trained with noisy data.

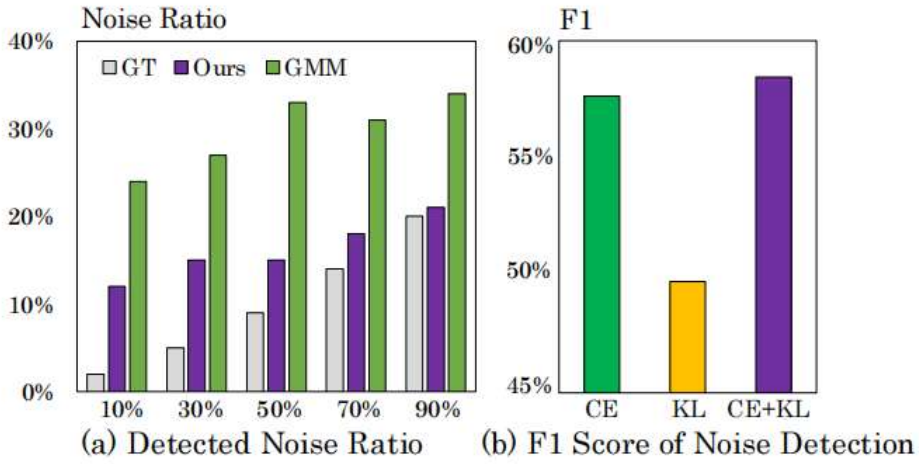


Figure 8: (a) shows detected noise proportion of GMM [18] and ours, where x-axis shows the noisy image ratio and y-axis shows the detected noisy pixel ratio. The detected noise proportion of ours is closer to the GT than GMM. (b) shows the F1 score of noise detection with different criterion. CE is cross entropy loss; KL is KL divergence between the predictions from the two LGs. Our CE+KL performs the best.

Junnan Li, Richard Socher, and Steven CH Hoi. Dividemix: Learning with noisy labels as semi-supervised learning.arXiv preprint arXiv:2002.07394, 2020.

南京航空航天大學

Nanjing University of Aeronautics and Astronautics

模式分析与机器智能
工业和信息化部重点实验室

MIIT Key Laboratory of
Pattern Analysis & Machine Intelligence

THANKS