Interactive Network Perturbation between Teacher and Students for Semi-Supervised Semantic Segmentation

Hyuna Cho, Injun Choi, Suha Kwak, Won Hwa Kim
Pohang University of Science and Technology (POSTECH)
{hyunacho, surung9898, suha.kwak, wonhwa}@postech.ac.kr

1. Additional Implementation Details

In this section, we provide additional details for our implementation which were not included in the main manuscript due to the page limit. Weight decay was set to $1e$-4, and Sync-BN [3] is used for batch normalization. For a single-scale evaluation, the crop size is set to 512×512 for PASCAL VOC 2012 and 800×800 for Cityscapes. We used Online Hard Example Mining (OHEM) loss [5] as supervision losses with the ground truth (i.e., L^t_{sup} and L^s_{sup}) on Cityscapes for all baselines and our method as in CPS [1], n-CPS [2], and PS-MT [4].

2. Trade-off among hyperparameters: α, β, γ

As shown in Table 1, we investigated various weight combinations on the pseudo-supervision losses of our method. Ablation study for α, β, γ, i.e., the hyperparameters for L_{gps}, L_{cps} and L_{fps}, were conducted. The best performances on both PASCAL VOC 2012 and Cityscapes were obtained with a relatively high weight on the L_{cps}. However, our proposed model is insensitive to the intensity of the hyperparameters, and the existence of all the three losses is significant to our method as reported in the ablation study (Table 4) of the main paper.

(a) PASCAL VOC 2012

α	β	γ	mIOU(%)
0.5	1.0	0.5	73.72
0.5	1.0	1.0	73.52
0.5	1.0	1.5	73.31
0.5	**1.5**	**0.5**	**73.95**
0.5	1.5	1.5	73.78
0.5	1.5	1.0	73.71
1.0	0.5	1.0	72.85
1.0	1.0	1.0	73.27
1.0	1.0	0.5	73.54
1.0	1.0	1.5	72.98
1.0	1.5	1.0	72.83
1.5	0.5	1.0	72.85
1.5	1.0	0.5	73.16

(b) Cityscapes

α	β	γ	mIOU(%)
1	3	1	73.77
1	4	1	74.10
1	**5**	**1**	**74.62**
1	6	1	73.87
2	3	2	73.29
2	4	2	73.41
2	6	2	73.96
4	4	4	73.53

Table 1. The results are obtained under $1/8$ labeled partition protocol with ResNet-50. The comparison is performed without CutMix.

3. Visualization of Interactions between TN and SN

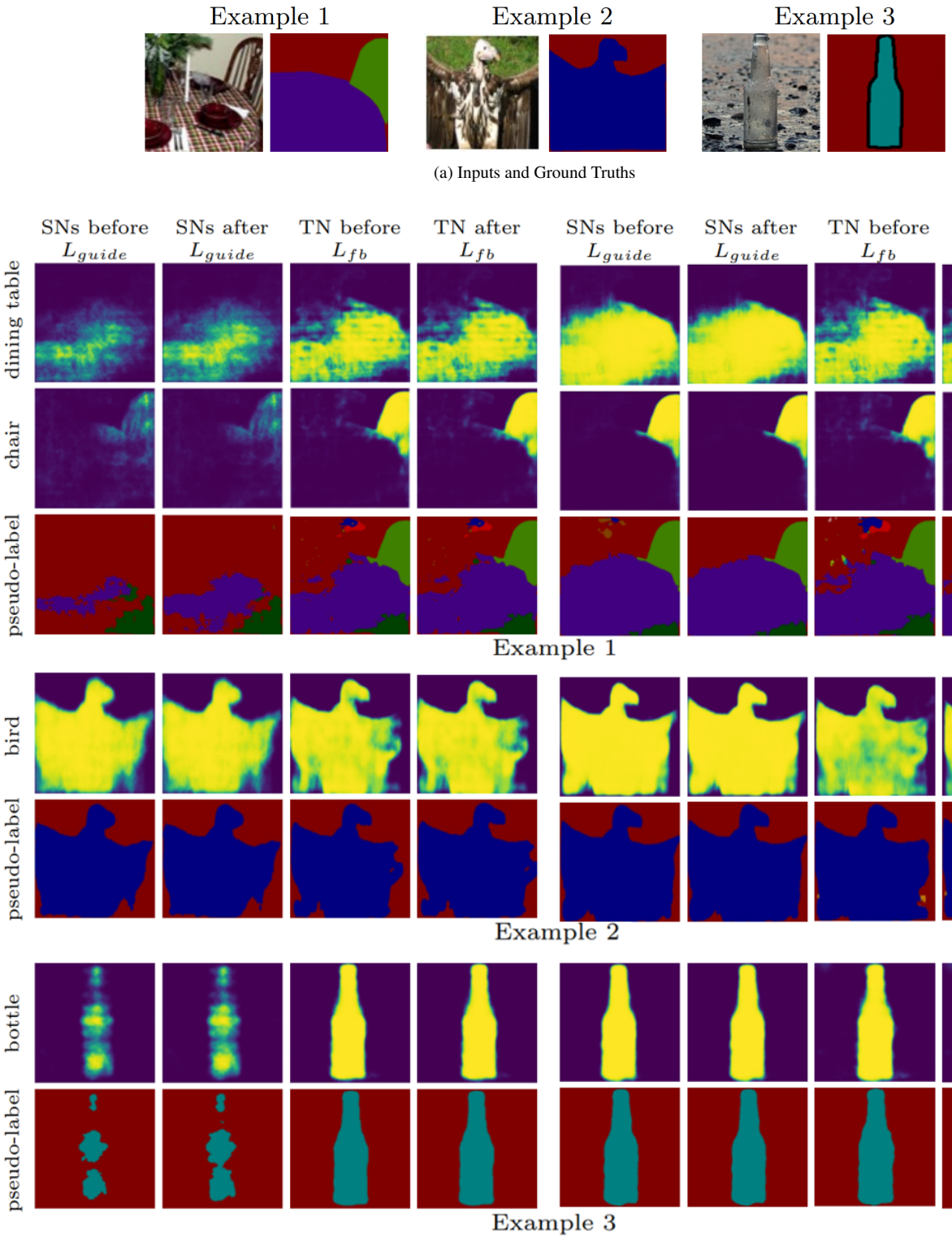
In one iteration, our method contains two losses (i.e., L_{guide} and L_{fb}) for a pair of Student Networks (SNs) and a Teacher Network (TN), respectively. To update the parameters of SNs, the L_{guide} uses pseudo-labels of TN to guide SNs, and concurrently uses pseudo-labels of SNs to teach each other. On the other hand, the L_{fb} takes the pseudo-labels from SNs to update the TN's parameters.

In Fig 1, we visualize confidence maps and pseudo-label maps of a exemplary samples from the 2nd and the last epochs within a single iteration. The qualitative comparison is performed before and after the two losses are applied to update the SNs and the TN in turn. The pseudo-labels of SNs before computing L_{guide} are used for the pseudo supervision between the SNs. After minimizing L_{guide}, they serve as pseudo-labels for the TN working as the feedback pseudo supervision (FPS). With L_{guide}, we empirically observed significant improvement on the prediction qualities of the SNs, and the changes are notably shown in both of the confidence maps and pseudo-labels in the 2nd epoch.

Regardless of L_{fb}, the TN's pseudo-labels are employed as the guided pseudo supervision (GPS) to evaluate SNs' predictions. The difference with the computation of L_{fb} is *when* they act as guides to SNs: before minimizing the L_{fb}, the TN's pseudo-labels are used in the *current* iteration with L_{guide}, whereas the updated TN produces pseudo-labels in the *next* iteration. With Adaptive Ramp-Up, the performance of TN is maintained until the training ends, and thus the TN can consistently provide informative pseudo-labels to SNs. Due to our interactive loss operation between L_{guide} and L_{fb}, the results of SNs outperform the TN's, and the superiority can be seen with clear and vivid confidence maps of SNs compared to the TN's prediction at the last epoch. In summary, students become better than the teacher.

4. Pseudo Code

We present the pseudo code of our whole training process in Algorithm 1.



(a) Inputs and Ground Truths

(b) 2-nd Epoch

(c) Last Epoch

Figure 1. **Visualization of Interactions between a TN and SNs** before and after applying loss functions in one iteration. The resultant pseudo-labels of SNs are forwarded from averaged confidence maps of the two SNs.

Algorithm 1 Training networks with GPS

procedure TRAINING(`epoch`, `batch`, `dataloader`)
  $\sigma(\cdot) \leftarrow \texttt{Softmax}(\cdot)$
  $f_{\theta_t}(\cdot), f_{\theta_{s_1}}(\cdot), f_{\theta_{s_2}}(\cdot) \leftarrow \texttt{TN}, \texttt{SN}_1, \texttt{SN}_2$
  $w(e) = \exp(-(1 - \frac{e}{\texttt{epoch}})^2) \leftarrow$ `Adaptive Ramp-Up function`
  for each `epoch` **do** ▷ Establishing a Teacher Network
    for each `batch` **do**
      $D^l = \{X_i^l\}_{i=1}^n, D^u = \{X_j^u\}_{j=1}^m \leftarrow \texttt{dataloader.iter}()$
      for $X_i^l \in D^l$ **do**
        $P_i^{l,t} \leftarrow \sigma(f(X_i^l; \theta_t))$
      end for
      Compute L_t^l using $P^{l,t}, Y^l$
      Update θ_t
    end for
  end for
  for e in `epoch` **do**
    for each `batch` **do**
      $D^l = \{X_i^l\}_{i=1}^n, D^u = \{X_j^u\}_{j=1}^m \leftarrow \texttt{dataloader.iter}()$
      for $X_i^l \in D^l$ and $X_j^u \in D^u$ **do** ▷ Guiding Step, Students Update
        $P_i^{l,s_1}, P_i^{l,s_2} \leftarrow \sigma(f(X_i^l; \theta_{s_1})), \sigma(f(X_i^l; \theta_{s_2}))$
        $P_j^{u,t}, P_j^{u,s_1}, P_j^{u,s_2} \leftarrow \sigma(f(X_j^u; \theta_t)), \sigma(f(X_j^u; \theta_{s_1})), \sigma(f(X_j^u; \theta_{s_2}))$
        $Y_j^t, Y_j^{s_1}, Y_j^{s_2} \leftarrow \texttt{max}(P_j^{u,t}), \texttt{max}(P_j^{u,s_1}), \texttt{max}(P_j^{u,s_2})$
      end for
      Compute L_s^l using $P^{l,s_1}, P^{l,s_2}, Y^l$
      Compute L_{gps} using $P^{u,s_1}, P^{u,s_2}, Y_j^t$
      Compute L_{cps} using $P^{u,s_1}, P^{u,s_2}, Y_j^{s_1}, Y_j^{s_2}$
      $L_{guide} \leftarrow L_s^l + \alpha L_{gps} + \beta L_{cps}$
      Update θ_{s_1} and θ_{s_2}
      for $X_i^l \in D^l$ and $X_j^u \in D^u$ **do** ▷ Feedback Step, Teacher Update
        $P_i^{l,t} \leftarrow \sigma(f(X_i^l; \theta_t))$
        $P_j^{u,t}, P_j^{u,s_1}, P_j^{u,s_2} \leftarrow \sigma(f(X_j^u; \theta_t)), \sigma(f(X_j^u; \theta_{s_1})), \sigma(f(X_j^u; \theta_{s_2}))$
        $Y_j^{\bar{s}} \leftarrow \frac{1}{2}(P_j^{u,s_1} + P_j^{u,s_2})$
      end for
      Compute L_t^l using $P^{l,t}, Y^l$
      Compute L_{fps} using $P^{u,t}, Y^{\bar{s}}$
      $L_{fb} \leftarrow L_t^l + w(e) \cdot \gamma L_{fps}$
      Update θ_t
    end for
    if $0.5 * \sum^{\texttt{batch}} L_s^l \leq \sum^{\texttt{batch}} L_t^l$ **then**
      $w(e) = 1$
    end if
  end for
end procedure

References

[1] Xiaokang Chen, Yuhui Yuan, Gang Zeng, et al. Semi-supervised semantic segmentation with cross pseudo supervision. In CVPR, 2021. 1

[2] Dominik Filipiak, Piotr Tempczyk, and Marek Cygan. n-cps: Generalising cross pseudo supervision to n networks for semi-supervised semantic segmentation. arXiv preprint arXiv:2112.07528, 2021. 1

[3] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In ICML, 2015. 1

[4] Yuyuan Liu, Yu Tian, Yuanhong Chen, et al. Perturbed and strict mean teachers for semi-supervised semantic segmentation. In CVPR, pages 4258–4267, 2022. 1

[5] Abhinav Shrivastava, Abhinav Gupta, and Ross Girshick. Training region-based object detectors with online hard example mining. In CVPR, pages 761–769, 2016. 1