# PrevMatch: Revisiting and Maximizing Temporal Knowledge in Semi-Supervised Semantic Segmentation

Supplementary Material

## 7. Additional Experiments and Ablation Studies

In this study, experiments were conducted in the following environments: Ubuntu 20.04, Python 3.10.4, PyTorch 1.12.1, and NVIDIA 3090Ti or A6000 GPUs. Unless otherwise specified, subsequent experiments were primarily conducted using the UniMatch baseline with ResNet-50.

### 7.1. Comparison with State-of-the-Art on the Priority Protocol of Pascal VOC

As described in the main paper, the Pascal VOC dataset contains a high-quality annotation subset (1,464 images) and a coarse-quality annotation subset (9,118 images). In addition to the *Original* and *Blended* protocols presented in the main manuscript, the *Priority* protocol has been used in several studies for comparison. The protocol where the selection of labeled images is first derived from the high-quality subset; if not sufficient, it is complemented by additional images from the coarse subset. Tab. 9 shows that the proposed method consistently improves baseline methods.

| **Pascal [Priority set]** | Encoder | 1/16 | 1/8 | 1/4 |
|---|---|---|---|---|
| U$^2$PL [48] | RN-101 | 77.2 | 79.0 | 79.3 |
| U$^2$PL+ [44] | RN-101 | 77.2 | 79.4 | 80.2 |
| Dual Teacher [31] | RN-101 | 80.1 | 81.5 | 80.5 |
| CorrMatch [41] | RN-101 | 81.3 | 81.9 | 80.9 |
| AllSpark [45] | MiT-B5 | 81.6 | 82.0 | 80.9 |
| FixMatch [40] | RN-50 | 75.4 | 76.5 | 76.9 |
| **+ PrevMatch** | RN-50 | **76.9** | **77.6** | **77.8** |
| Gain (Δ) | | (+1.5) | (+1.1) | (+0.9) |
| UniMatch [54] | RN-101 | 80.9 | 81.9 | 80.4 |
| **+ PrevMatch** | RN-101 | **81.4** | **81.9** | **80.8** |
| Gain (Δ) | | (+0.5) | (+0.0) | (+0.4) |
| UniMatchV2-S [55] | DINOv2-S | 86.6 | 87.3 | 85.9 |
| **+ PrevMatch** | DINOv2-S | **87.3** | **87.8** | **86.4** |
| Gain (Δ) | | (+0.7) | (+0.5) | (+0.5) |

Table 9. Comparison with state-of-the-art methods on the *Priority* protocol of Pascal VOC dataset.

### 7.2. Additional Ablation Studies

Due to space constraints, the ablation study (Tab. 5) in the main body of the paper includes only the 92-label protocol. To further validate the proposed method, we provide additional experiments on the 183- to 1464-label protocols. Tab. 10 shows that the components of PrevMatch consistently improve performance across various evaluation protocols.

| Previous | Random | Random | | | PASCAL | | |
|---|---|---|---|---|---|---|---|
| Guidance | Selection | Weights | 92 | 183 | 366 | 732 | 1464 |
| - | - | - | 71.9 | 72.5 | 76.0 | 77.4 | 78.7 |
| ✓ | - | - | 72.7 | 73.8 | 76.6 | 78.1 | 79.0 |
| ✓ | ✓ | - | 73.2 | 74.9 | 77.4 | 78.3 | 79.2 |
| ✓ | ✓ | ✓ | **73.4** | **75.4** | **77.5** | **78.6** | **79.3** |

Table 10. Additional ablation studies across different evaluation protocols (UniMatch with ResNet-50).

### 7.3. Previous List Length

We investigate the effect of the length ($N$) of the previous list that stores the temporal models. In Tab. 11, the results for $N = 1$ and $N = 2$ are comparable to those of the baseline. This suggests that the aforementioned coupling problem may persist because the previous models in the list are continually updated with the latest model when $N$ is small. The cases of $N = 4$, 8, and 12 consistently outperform the baseline, revealing that the proposed method is not highly sensitive to hyperparameters. However, the performance for $N = 20$ increases only marginally due to the use of outdated teachers. Based on this result, we recommend setting the value of $N$ to approximately 5–15% of the total training epochs, which proves to be appropriate for different datasets (e.g., Pascal=6–10, Cityscapes=8–16, and ADE20k=4–6).

| $N$ | Base. | 1 | 2 | 4 | 8 | 12 | 20 |
|---|---|---|---|---|---|---|---|
| Pascal$_{92}$ | 71.9 | 71.7 | 71.7 | 72.4 | **72.7** | 72.5 | 71.9 |
| Pascal$_{183}$ | 72.5 | 72.7 | 72.9 | 73.4 | **73.8** | 73.5 | 73.3 |

Table 11. Ablation study for the maximum length ($N$) of the previous list. In this setting, only previous guidance is used (i.e., $K$=1, without ensemble).

### 7.4. Upper Bound Number for Random Selection

To generate reliable and diverse pseudo-labels, we proposed a strategy that randomly selects $k$ models (ranging from 1 to $K$) for each iteration. In this strategy, we explore the performance changes regarding the upper bound number $K$. Tab. 12 indicates that including the ensembling cases ($K > 1$, i.e., $k$=1 or $k$ >1 are randomly selected) improves the performance significantly compared to the case of $K = 1$ (i.e., without ensemble). In addition, we observe

| Upper Bound Number ($K$) | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| Pascal$_{92}$ | 72.7 | 73.1 | **73.4** | **73.4** | 73.1 |
| Pascal$_{183}$ | 73.8 | 74.9 | **75.4** | 75.2 | 75.1 |

Table 12. Ablation study on the efficacy of the upper bound number $K$ using $N$=8.

| Save Criteria | 92 | 183 | 366 | 732 | 1464 |
|---|---|---|---|---|---|
| (a) Baseline | 71.9 | 72.5 | 76.0 | 77.4 | 78.7 |
| (b) Every 1 Epoch | 71.8 | 73.5 | 76.3 | 77.5 | 78.6 |
| (c) Every 3 Epochs | 72.2 | 74.7 | 76.8 | 78.0 | 78.6 |
| (d) On Best Epochs (ours) | **73.4** | **75.4** | **77.5** | **78.6** | **79.3** |

Table 13. Ablation study regarding the efficacy of the save criteria using $N$=8 and $K$=3.

the best results at $K = 3$ and a slight performance drop in settings with $K$ greater than 3. Even for large $K$, a varying number ($k$) of models is selected; however, the proportion of large $k$ values increases with $K$. This ensures consistent pseudo-labels but reduces their diversity, potentially degrading performance, as mentioned in Dual Teacher. In conclusion, we select $K = 3$ because it adequately satisfies the diversity and reliability requirements of the pseudo-labels.

### 7.5. Criteria for Saving Previous Models

As described in Sec. 3.2.1 of the main paper, one alternative for storing previous models involves saving the model at regular intervals, a method used in [15, 43, 53]. Thus, we conduct experiments to validate the effectiveness of this approach. As listed in Tab. 13, although case (b) exhibits slightly better overall performance than the baseline (a), the difference is marginal. This suggests that storing models at short intervals does not address the coupling problem between the teacher and student networks. In contrast, case (c) shows a significant improvement compared to case (a). Although case (c) functions well, it exhibits limited improvements compared to case (d) which utilizes the proposed save criteria, demonstrating the superiority of the proposed approach. In addition, our approach does not require additional hyperparameter searches to determine appropriate intervals, thereby reducing unnecessary training costs.

### 7.6. Loss Weight

The loss weight $\lambda$ for previous guidance plays a critical role in the overall training process. Since the primary goal of previous guidance is to mitigate confirmation bias–which tends to occur during the early to middle stages of training–its effect is most beneficial before the model becomes overconfident in incorrect predictions. However, in the early phase, the model's predictions are typically unreliable. To address this, we employ a warmup schedule that gradually increases $\lambda$, similar to commonly used learning rate warmup strategies. Furthermore, as training progresses and the model becomes more stable and accurate, the need for strong regularization from previous guidance naturally decreases. To reflect this, we apply a decay to $\lambda$ in the later stages. Our final schedule (row (e)) allows previous guidance to act most strongly when it is most needed–during the middle of training–while reducing its influence as the model converges.

Tab. 14 presents an ablation study comparing different $\lambda$ scheduling strategies. The fixed (b) and linear decay (c) settings maintain a high weight even in the early stage, where the model's predictions are still unstable. As a result, these configurations show limited improvements, likely due to the guidance based on noisy predictions. In contrast, the linear increase (d) and warmup+decay (e) strategies both incorporate a warmup phase, mitigating this issue and yielding significant performance gains across various label partitions. While both (d) and (e) outperform other variants, the warmup+decay strategy shows slightly better or comparable results, especially in high-label settings. This indicates that reducing the influence of previous guidance in the late stage, when the model is already well-trained, is beneficial for final performance.

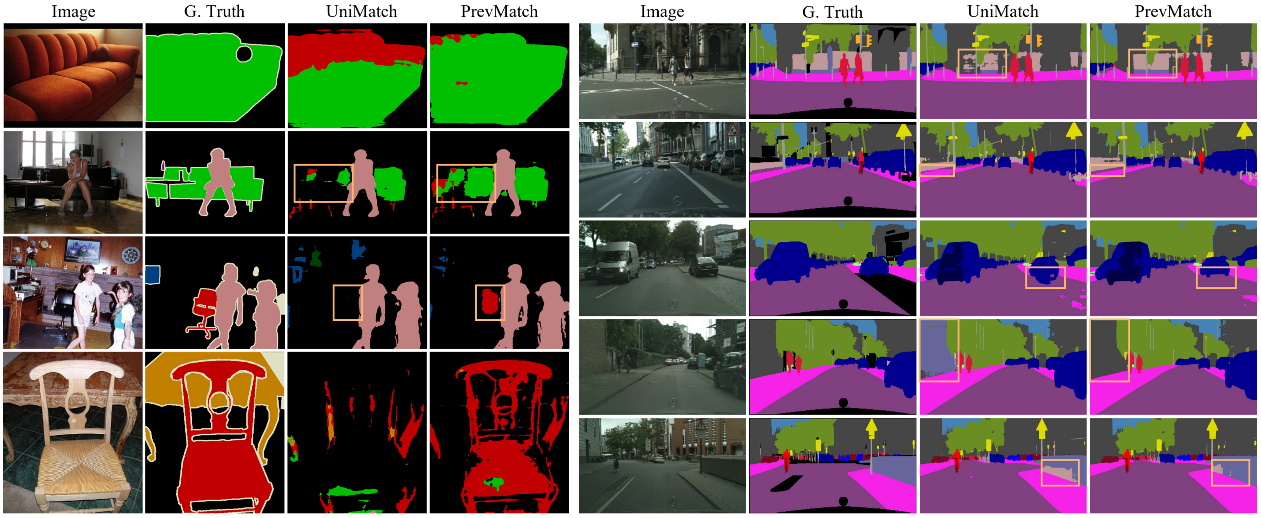
| **Loss weight ($\lambda$)** | 92 | 183 | 366 | 732 | 1464 |
|---|---|---|---|---|---|
| (a) Baseline | 71.9 | 72.5 | 76.0 | 77.4 | 78.7 |
| (b) Fixed (1.0) | 72.1 | 74.2 | 76.7 | 78.2 | 78.6 |
| (c) Linear Decay (1.0→0) | 71.5 | 74.5 | 76.9 | 78.4 | 78.8 |
| (d) Linear Increase (0→1.0) | **73.6** | **75.4** | 77.3 | 78.5 | 79.2 |
| (e) Warmup+Decay (0→1.0→0) | 73.4 | **75.4** | **77.5** | **78.6** | **79.3** |

Table 14. Ablation study for $\lambda$.

## 8. Additional Analysis

### 8.1. Class-wise IoU Scores and Qualitative Evaluation

Tabs. 15 and 16 list all category-wise IoU scores. In particular, Tab. 15 on Pascal VOC shows that the proposed method achieves notable performance gains for the chair and sofa classes, which were particularly challenging for the UniMatch baseline. In addition, Tab. 16 on the Cityscapes dataset shows that the proposed method achieves the largest performance improvements for the wall, fence, and terrain classes, which are the lowest performing among the 19 classes in UniMatch. In addition to the quantitative results, the qualitative results shown in Fig. 3 corroborate these findings, revealing consistent improvements in the same categories. Thus, these results suggest that utilizing previous knowledge helps prevent the catastrophic forgetting problem, even in semi-supervised semantic segmentation scenarios.



(a) PASCAL VOC

(b) Cityscapes

Figure 3. Qualitative segmentation results on (a) Pascal VOC and (b) Cityscapes.

| | ***backgr.*** | ***airplane*** | ***bicycle*** | ***bird*** | ***boat*** | ***bottle*** | ***bus*** | ***car*** | ***cat*** | ***chair*** | ***cow*** | ***d.table*** | ***dog*** | ***horse*** | ***m.bike*** | ***person*** | ***p.plant*** | ***sheep*** | ***sofa*** | ***train*** | ***tv*** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| UniMatch | 91 | 84 | 59 | 89 | 71 | 68 | 92 | 80 | 88 | 8 | 88 | 55 | 85 | 85 | 75 | 80 | 54 | 82 | 33 | 80 | 64 |
| + PrevMatch | 93 | 84 | 61 | 87 | 71 | 68 | 93 | 84 | 89 | 21 | 89 | 57 | 84 | 86 | 77 | 82 | 51 | 82 | 46 | 84 | 58 |
| Gain (Δ) | 2 | 0 | 2 | -2 | 0 | 0 | 1 | 4 | 1 | 13 | 1 | 2 | -1 | 1 | 2 | 2 | -3 | 0 | 13 | 4 | -6 |

Table 15. Class-wise IoU scores for Pascal VOC using a ResNet-50 encoder.

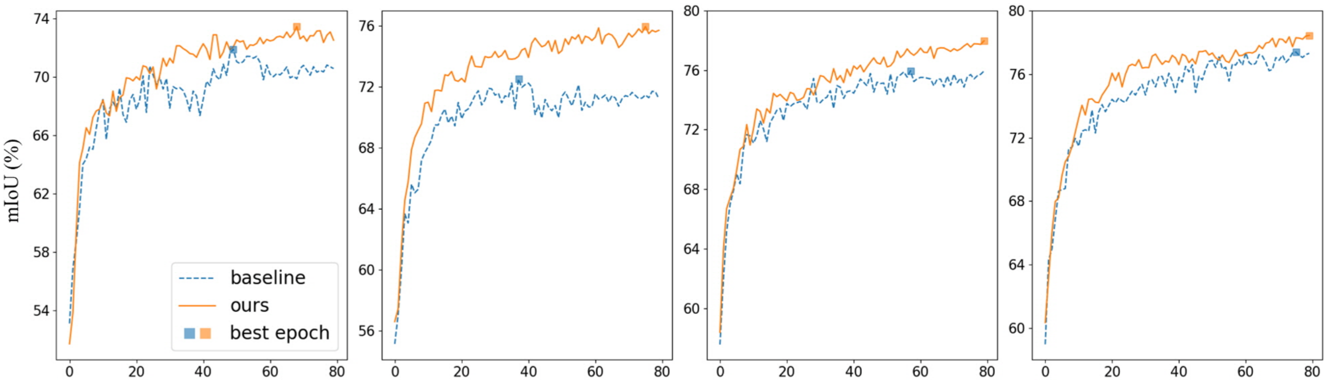
| | ***road*** | ***sidewalk*** | ***build.*** | ***wall*** | ***fence*** | ***pole*** | ***t.light*** | ***t.sign*** | ***veget.*** | ***terrain*** | ***sky*** | ***person*** | ***rider*** | ***car*** | ***truck*** | ***bus*** | ***train*** | ***m.cycle*** | ***bicycle*** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| UniMatch | 98 | 82 | 92 | 56 | 60 | 62 | 71 | 79 | 92 | 60 | 95 | 82 | 63 | 95 | 83 | 87 | 79 | 68 | 77 |
| + PrevMatch | 98 | 84 | 92 | 60 | 63 | 63 | 71 | 80 | 92 | 63 | 95 | 82 | 64 | 95 | 83 | 88 | 81 | 69 | 77 |
| Gain (Δ) | 0 | 2 | 0 | 4 | 3 | 1 | 0 | 1 | 0 | 3 | 0 | 0 | 1 | 0 | 0 | 1 | 2 | 1 | 0 |

Table 16. Class-wise IoU scores for Cityscapes using a ResNet-101 encoder.

### 8.2. Training Stability Across Different Label Partitions

Fig. 4 illustrates the effect of the proposed method on the changes in the validation scores throughout the training process. In fewer label settings (92 and 183), the UniMatch baseline (blue) exhibits significant fluctuations in terms of performance compared to the proposed method (orange). Moreover, when considering the epoch that achieves the best performance, the baseline method struggles to converge consistently across epochs and tends to become trapped in local minima prematurely. This issue is particularly pronounced in scenarios with fewer labels. In contrast, the proposed method consistently converges across epochs without significant fluctuations.

Meanwhile, as mentioned in Sec. 3.2 of the main paper, given the standard and previous guidance, four scenarios can be considered based on whether they are correct or incorrect (standard-previous): (1) correct-correct, (2) correct-incorrect, (3) incorrect-correct, and (4) incorrect-incorrect. Through case (3), the network receives an additional opportunity to be guided in the right direction, away from the wrong one. However, model training may also be hindered through case (2), leading to significant fluctuations. Nevertheless, as shown in Fig. 4, the consistent outperformance of our method over the baseline across almost all training epochs in different label partitions suggests that the positive effects (i.e., case (3)) of previous guidance outweigh any negative effects (i.e., case (2)).



(a) 92 (b) 183 (c) 366 (d) 732

Figure 4. Training curves for different label partition settings on Pascal VOC. The X- and Y-axes represent epochs and validation mIoU, respectively. The square symbol (■) denotes the epoch with the best performance.

## 9. Limitations and Future Work

We showed that the proposed method is effective on several benchmark datasets that share the domain between labeled and unlabeled images. However, it has not been investigated in domain adaptation problems that involve domain discrepancies between the labeled and unlabeled data. This problem poses a more challenging self-training task, a commonly encountered problem in real-world applications. In future work, we will investigate the capability of temporal knowledge in the domain adaptation problem. In addition, we observed a phenomenon where significant fluctuations in pseudo-label accuracy for several classes negatively affect generalization ability. For instance, pseudo-label accuracy recovers slightly after a sharp drop in performance; however, the validation score does not. Although the proposed method has shown that it can mitigate these issues, a more in-depth investigation is required for scenarios involving many classes and imbalanced distributions, such as the COCO and ADE20K datasets. Therefore, we intend to explore this phenomenon extensively across different classes, in terms of the relationship between training instability and generalization ability.