# PseudoProp: Robust Pseudo-Label Generation for Semi-Supervised Object Detection in Autonomous Driving Systems – Appendix

Shu Hu[1]*, Chun-Hao Liu[2]†, Jayanta Dutta[2], Ming-Ching Chang[3], Siwei Lyu[1], Naveen Ramakrishnan[4]
[1]University at Buffalo, SUNY, USA [2]Bosch Center for Artificial Intelligence, USA
[3]University at Albany, SUNY, USA [4]Amazon, USA
{shuhu, siweilyu}@buffalo.edu, {Chun-Hao.Liu, Jayanta.Dutta}@us.bosch.com
mchang2@albany.edu, rnaveen83@gmail.com

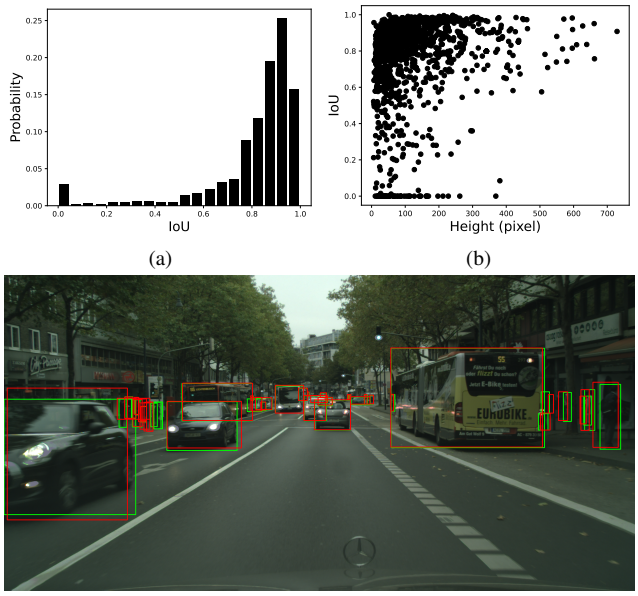
This appendix provides supplementary details of the proposed method and additional results aside from the main paper.

## 1. Self-Consistency of Motion Prediction

We analyze the self-consistency of motion prediction by validating the accuracy of the estimated motion vectors. We first predict the bounding boxes $\hat{Y}_{t+1}$ on $X_{t+1}$ using SDC-Net [2], given the current ground truth bounding box $Y_t$, $X_t$, and $X_{t-1}$. We then reconstruct $\check{Y}_t$ by using reversed motion prediction from $\hat{Y}_{t+1}$, $X_{t+1}$, and $X_{t+2}$. Finally, we measure the IoU between $Y_t$ and $\check{Y}_t$ as the self-consistency estimation. We randomly select 100 images from the Cityscapes dataset [1] and measure such IoU performance. A total of $2,167$ bounding boxes are measured, and the mean of all the measured IoUs is $0.81$. We can see from this result that the SDC-Net motion estimation consistency is indeed high.

Fig. 1(a) shows the probability mass function of the measured IoUs from the above self-consistency test on the 100 random Cityscapes images. There are a few IoU $= 0$ cases, which is mainly due to: (1) The predicted bounding boxes are outside video frames, where the original boxes are near frame boundary: with probability $\Pr(\text{Out} \mid \text{IoU} = 0) = 25\%$. (2) Small objects are more error-prone to reconstruction: with probability $\Pr(\text{Height} \leq 45 \mid \text{IoU} = 0) = 46\%$, where the average height for all objects is 96 pixels.

The scatter plot in Fig. 1(b) shows the relationship between the object height and IoU for this self-consistency test, with Pearson correlation coefficient $0.07$ (little or no relationship). In other words, the IoU is not biased toward either tall or short objects. A similar observation is also found for object area versus IoU. Table 1 lists the per-class average IoU from the self-consistency test. Observe in this table that two specific types of vehicles, namely bus and truck, are with higher IoU. This may be due to the slow motion of buses and trucks, which is easier to estimate (in contrast, other vehicle types tend to move faster). Another potential reason is that buses and trucks do not often appear in groups unlike people and cars. The grouping for objects makes motion estimation difficult due to potential occlusions, and Fig. 1(c) shows one example. On the left-hand side of this figure, motion deviation is large for the group of people. Also, observe that all three buses are with good bounding box reconstruction.




(a) (b)

(c)

Figure 1. **Self-consistency test.** *(a) The probability mass function for IoU using pre-trained SDC-Net. (b) Scatter plot of object height versus IoU. (c) A visualization example, where green depicts the ground truth boxes, and red depicts the reconstructed bounding boxes.*

*Work done while interning at Bosch Center for Artificial Intelligence
†Corresponding author

| Class | car | person | truck | rider | motorcycle | bicycle | bus |
|---|---|---|---|---|---|---|---|
| IoU | 0.84 | 0.76 | 0.89 | 0.78 | 0.76 | 0.78 | 0.92 |

Table 1. *Per-class average IoU from the self-consistency test. Only* 7 *classes are shown, as there are no train instance from the* 100 *randomly sampled images.*

## 2. Additional Details of PseudoProp

In this section, we will provide an example to explain the procedure of BPLP and the details of WBF.

### 2.1. An Example Explaining $\widehat{Y}_{t+1}$

This section explains the bidirectional pseudo-label propagation (BPLP) on the frame $X_{t+1}$ to generate $\hat{Y}_{t+1}$ by setting $k = 2$ in Eq.(2) of the main paper.

Given $k = 2$, then $K = \{-2, -1, 1, 2\}$. Thus we should do motion propagation from $Y_{t+3}$ (for $i = -2$), $Y_{t+2}$ (for $i = -1$), $Y_t$ (for $i = 1$), and $Y_{t-1}$ (for $i = 2$) to $Y_{t+1}$, respectively.

For $Y_{t+3}$, the motion vector should be the combination of $\mathcal{M}(X_{t+2:t+4}, V_{t+3:t+2})$ (for $j = -2$) and $\mathcal{M}(X_{t+1:t+3}, V_{t+2:t+1})$ (for $j = -1$). Therefore, we obtain

$$\hat{Y}_{t+1}^{-2} = \mathcal{T}\big(\mathcal{M}(X_{t+2:t+4}, V_{t+3:t+2}) + \mathcal{M}(X_{t+1:t+3}, V_{t+2:t+1}), Y_{t+3}.\big)$$

For $Y_{t+2}$, the motion vector should be $\mathcal{M}(X_{t+1:t+3}, V_{t+2:t+1})$ (for $j = -1$). Therefore, we have

$$\hat{Y}_{t+1}^{-1} = \mathcal{T}\big(\mathcal{M}(X_{t+1:t+3}, V_{t+2:t+1}), Y_{t+2}\big).$$

For $Y_t$, the motion vector should be $\mathcal{M}(X_{t-1:t+1}, V_{t:t+1})$ (for $j = 1$). Therefore, we have

$$\hat{Y}_{t+1}^{1} = \mathcal{T}\big(\mathcal{M}(X_{t-1:t+1}, V_{t:t+1}), Y_t\big).$$

For $Y_{t-1}$, the motion vector should be the combination of $\mathcal{M}(X_{t-2:t}, V_{t-1:t})$ (for $j = 2$) and $\mathcal{M}(X_{t-1:t+1}, V_{t:t+1})$ (for $j = 1$). Therefore, we have

$$\hat{Y}_{t+1}^{2} = \mathcal{T}\big(\mathcal{M}(X_{t-2:t}, V_{t-1:t}) + \mathcal{M}(X_{t-1:t+1}, V_{t:t+1}), Y_{t-1}\big).$$

Hence the final $\hat{Y}_{t+1}$ should be

$$\hat{Y}_{t+1} = \hat{Y}_{t+1}^{-2} \cup \hat{Y}_{t+1}^{-1} \cup \hat{Y}_{t+1}^{1} \cup \hat{Y}_{t+1}^{2}.$$

### 2.2. The Weighted Box Fusion (WBF)

This section explains details of the weighted box fusion (WBF), and the following procedure is organized from the content of the original paper [3].

1. First, bounding boxes in $\overline{Y}_{t+1,c}$ are sorted and saved in a descending order list $B$ according to their confidence scores.
2. Define two lists $L = \varnothing$ and $F = \varnothing$ for box clusters and fused boxes, respectively. Each position in the list $L$ can contain a set of boxes, which form a cluster. Each position in $F$ contains one box, which is the fused box from the corresponding cluster in $L$.
3. Iterate through boxes in $B$ and try to find a matching box in the list $F$. The matching should satisfy that IoU is greater than a user-defined threshold `Thr`.
4. If a match box is not found, add the current box from $B$ to the end of list $L$ and $F$ as new elements and proceed to the next box in $B$.
5. If a match is found, add this box to the list $L$ at cluster $r$ corresponding to the matching box in list $F$
6. For boxes in each cluster $r$, we calculate their average confidence score $C_r$, and regard their individual confidence score as a weight for their positions and do the weighted average for the positions as follows.

$$C_r = \frac{1}{T}\sum_{l=1}^{T} C_r^l, \quad P_r = \frac{\sum_{l=1}^{T} C_r^l \cdot P_r^l}{\sum_{l=1}^{T} C_r^l},$$

   where $T$ is the total number of boxes in the cluster $r$. $C_r^l$ and $P_r^l$ are the confidence scores and the position of the $l$-th box in the cluster $r$, respectively.
7. Re-scale $C_r$ by $C_r = C_r \cdot \frac{\min(T,|K|+1)}{|K|+1}$, where $|K|$ is the size of the set $K$ from Eq. (3). Finally, $\overline{Y}_{t+1,c}$ only contains the average bounding box information $(c, P_r, C_r)$ from each cluster.

## 3. Additional Experimental Results

In this section, we will provide more experimental results.

### 3.1. The Details of Model Performance

We show the details of model performance under different settings and also report the mAP$^{50}$ performance on each class in Table 2, 3, 4, 5, 6, and 7.

From Table 2, we can find our method can get the best performance when using $1\times$ pseudo-labeled data. However, when we increase pseudo-labeled data, the VideoProp method has better performance. The reason is that the generated pseudo-labels from the VideoProp method are very close to the GT labels. Therefore, the pseudo-labeled data has high quality. But this method can only generate pseudo-labels near the GT. Our model is more flexible and general than the VideoProp. On the other hand, if we compare the model performance in the "train" class, it is clear that our method has high performance in the rare class when using $1\times$ and $2\times$ pseudo-labeled data. For Table 3, we can

| Models | Pseudo-labled Data Ratio | mAP | mAP$^{50}$ | mAP$^{75}$ | bicycle | bus | car | motorcycle | person | rider | train | truck |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| EfficientDet-D1 | - | 19.0 | 35.5 | 17.2 | 29.4 | 45.4 | 53.6 | 22.8 | 32.2 | 36.7 | 38.9 | 25.4 |
| SSD | - | - | 36.7 | - | 30.1 | 47.5 | 60.2 | 26.9 | 36.3 | 37.2 | 28.8 | 26.6 |
| DSPNet | - | - | 36.9 | - | 30.0 | 49.3 | 59.1 | 24.6 | 34.9 | 37.7 | 30.4 | 29.4 |
| VideoProp | 1× | 21.7 | 40.3 | 19.9 | 32.4 | 52.8 | 59.4 | 26.5 | 35.1 | 39.7 | 42.4 | 33.9 |
| | 2× | 21.9 | 43.0 | 19.6 | 32.1 | 55.0 | 60.8 | 27.0 | 36.1 | 42.6 | 56.3 | 33.7 |
| | 3× | 22.3 | 42.0 | 19.8 | 34.1 | 55.1 | 60.3 | 24.4 | 37.6 | 41.5 | 48.4 | 34.7 |
| Naive-Student (iteration 1) | 1× | 20.8 | 39.0 | 18.8 | 29.3 | 51.0 | 55.6 | 25.3 | 33.8 | 36.8 | 50.0 | 30.5 |
| | 2× | 21.2 | 38.9 | 19.6 | 31.1 | 49.7 | 55.5 | 23.4 | 33.9 | 37.7 | 48.3 | 31.8 |
| | 3× | 21.0 | 39.7 | 18.7 | 29.9 | 50.7 | 56.0 | 26.5 | 34.3 | 38.0 | 52.0 | 30.0 |
| PseudoProp (iteration 1) | 1× | 21.6 | 40.4 | 19.9 | 30.9 | 50.3 | 56.3 | 24.5 | 34.9 | 37.5 | 56.4 | 32.2 |
| | 2× | 21.7 | 41.0 | 20.2 | 30.3 | 52.2 | 55.9 | 25.6 | 34.4 | 38.2 | 59.6 | 31.6 |
| | 3× | 21.7 | 40.0 | 19.8 | 31.2 | 50.4 | 57.0 | 25.4 | 35.8 | 38.4 | 49.3 | 32.3 |

Table 2. *Comparison of mAP (%), mAP$^{50}$ (%), and mAP$^{75}$ (%) of different object detection baseline models on the Cityscapes test dataset. For semi-supervised models, we test different pseudo-labeled data ratio. The mAP$^{50}$ (%) performance for each class is also reported.*

| Thresholds | $k$ | mAP | mAP$^{50}$ | mAP$^{75}$ | bicycle | bus | car | motorcycle | person | rider | train | truck |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 21.8 | 39.5 | 20.5 | 31.0 | 50.0 | 56.1 | 26.2 | 34.2 | 38.1 | 49.4 | 31.1 |
| | 2 | 20.4 | 39.9 | 18.0 | 29.5 | 49.7 | 55.3 | 24.9 | 33.6 | 37.1 | 57.6 | 31.2 |
| | 3 | 21.7 | 40.3 | 20.0 | 30.5 | 51.3 | 55.8 | 26.0 | 33.4 | 37.2 | 57.9 | 30.8 |
| 0.1 | 1 | 21.6 | 40.4 | 19.9 | 30.9 | 50.3 | 56.3 | 24.5 | 34.9 | 37.5 | 56.4 | 32.2 |
| | 2 | 21.3 | 39.6 | 19.4 | 30.6 | 51.8 | 55.3 | 25.1 | 34.3 | 38.0 | 52.1 | 29.4 |
| | 3 | 20.8 | 40.1 | 18.9 | 30.7 | 50.9 | 55.4 | 24.1 | 34.5 | 37.8 | 56.0 | 31.1 |
| 0.2 | 1 | 21.8 | 40.3 | 20.3 | 29.5 | 51.9 | 56.2 | 24.8 | 33.8 | 37.4 | 58.4 | 30.2 |
| | 2 | 20.6 | 39.1 | 18.6 | 31.0 | 49.2 | 55.3 | 23.4 | 33.7 | 37.5 | 55.1 | 27.9 |
| | 3 | 20.5 | 39.5 | 18.4 | 31.0 | 48.5 | 55.1 | 24.5 | 33.9 | 37.2 | 54.7 | 31.2 |
| 0.3 | 1 | 21.0 | 40.1 | 18.6 | 31.7 | 48.6 | 56.5 | 22.2 | 34.0 | 37.1 | 58.5 | 32.3 |
| | 2 | 20.7 | 39.2 | 18.0 | 30.7 | 48.0 | 55.5 | 23.8 | 33.9 | 37.3 | 55.1 | 29.5 |
| | 3 | 20.7 | 39.3 | 19.7 | 30.1 | 48.3 | 55.4 | 21.2 | 33.8 | 36.9 | 56.4 | 32.4 |

Table 3. *Comparison of mAP (%), mAP$^{50}$ (%), and mAP$^{75}$ (%) of the PseudoProp model on the Cityscapes test dataset when using different thresholds and different $k$ values. The mAP$^{50}$ (%) performance for each class is also reported.*

| Fusion Methods | mAP | mAP$^{50}$ | mAP$^{75}$ | bicycle | bus | car | motorcycle | person | rider | train | truck |
|---|---|---|---|---|---|---|---|---|---|---|---|
| NMS | 21.0 | 39.7 | 19.1 | 30.0 | 51.1 | 55.3 | 24.6 | 34.3 | 37.3 | 54.5 | 30.8 |
| NMW | 21.0 | 39.8 | 19.1 | 29.1 | 50.0 | 55.2 | 24.9 | 34.3 | 36.0 | 56.5 | 32.3 |
| SNMS | 21.2 | 39.8 | 19.3 | 30.2 | 50.7 | 55.1 | 24.6 | 33.2 | 36.5 | 57.7 | 30.1 |
| WBF | 21.0 | 39.6 | 19.1 | 30.6 | 49.4 | 55.3 | 24.6 | 34.0 | 37.0 | 55.9 | 30.1 |
| SWBF | 21.6 | 40.4 | 19.9 | 30.9 | 50.3 | 56.3 | 24.5 | 34.9 | 37.5 | 56.4 | 32.2 |

Table 4. *Comparison of mAP (%), mAP$^{50}$ (%), and mAP$^{75}$ (%) of the PseudoProp model on the Cityscapes test dataset when using different fusion methods. The mAP$^{50}$ (%) performance for each class is also reported.*

| Methods | Labled Data Size | mAP | mAP$^{50}$ | mAP$^{75}$ | bicycle | bus | car | motorcycle | person | rider | train | truck |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Naive-Student (iteration 1) | 2000 | 20.8 | 39.8 | 18.3 | 30.0 | 49.4 | 55.5 | 25.0 | 35.1 | 38.1 | 56.2 | 29.5 |
| | 1000 | 18.5 | 36.4 | 16.5 | 29.3 | 47.3 | 53.8 | 24.1 | 33.0 | 34.8 | 41.3 | 27.5 |
| | 500 | 17.7 | 34.7 | 15.5 | 28.8 | 45.7 | 53.6 | 21.4 | 32.5 | 34.7 | 37.6 | 23.4 |
| PseudoProp (iteration 1) | 2000 | 20.8 | 39.8 | 18.3 | 30.0 | 49.4 | 55.5 | 25.0 | 35.1 | 38.1 | 56.2 | 29.5 |
| | 1000 | 19.6 | 37.2 | 17.5 | 28.2 | 47.5 | 54.4 | 23.8 | 33.3 | 36.1 | 42.2 | 32.3 |
| | 500 | 18.6 | 36.1 | 16.7 | 28.5 | 49.9 | 54.2 | 22.1 | 33.2 | 34.4 | 36.4 | 30.1 |

Table 5. *Comparison of mAP (%), mAP$^{50}$ (%), and mAP$^{75}$ (%) of the Naive-Student and PseudoProp models on the Cityscapes test dataset when using different small labeled data size. The mAP$^{50}$ (%) performance for each class is also reported.*

| Methods | mAP | mAP$^{50}$ | mAP$^{75}$ | bicycle | bus | car | motorcycle | person | rider | train | truck |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Naive-Student (iteration 2) | 22.2 | 40.8 | 20.3 | 30.9 | 50.6 | 56.7 | 25.7 | 36.1 | 38.1 | 55.5 | 32.7 |
| PseudoProp (iteration 2) | 22.6 | 41.4 | 20.9 | 32.9 | 50.0 | 58.2 | 24.7 | 36.9 | 39.5 | 55.7 | 33.6 |

Table 6. *Comparison of mAP (%), mAP$^{50}$ (%), and mAP$^{75}$ (%) of the Naive-Student and PseudoProp models on the Cityscapes test dataset at iteration 2. The mAP$^{50}$ (%) performance for each class is also reported.*

find the mAP, mAP$^{50}$, and mAP$^{75}$ performance of PseudoProp method can achieve the best when we set $k = 1$. For Table 4, we can find the SWBF fusion method outperforms other methods. Specifically, when we compare WBF and SWBF, it is clear that applying the similarity method to the WBF method can improve the model performance. For Table 5, when we decrease the labeled data size, the performance gap between Naive-Student and our PseudoProp

| Models | Pseudo-labled Data Ratio | mAP | mAP$^{50}$ | mAP$^{75}$ | bicycle | bus | car | motorcycle | person | rider | train | truck |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Naive-Student* (iteration 1) | 2× | 22.8 | 43.3 | 19.8 | 34.0 | 54.6 | 60.5 | 26.1 | 38.0 | 41.2 | 56.6 | 35.6 |
| | 3× | 23.1 | 43.2 | 21.5 | 33.3 | 54.1 | 60.5 | 28.3 | 38.5 | 41.2 | 51.8 | 38.2 |
| PseudoProp* (iteration 1) | 2× | 23.2 | 44.4 | 20.9 | 34.7 | 50.8 | 60.8 | 31.4 | 38.3 | 41.4 | 62.1 | 35.6 |
| | 3× | 23.1 | 43.9 | 21.3 | 34.2 | 55.2 | 61.3 | 30.8 | 39.0 | 41.7 | 53.4 | 35.5 |

Table 7. *Comparison of mAP (%), mAP$^{50}$ (%), and mAP$^{75}$ (%) of the Naive-Student* and PseudoProp* models on the Cityscapes test dataset at iteration 1 when using different pseudo-labled data ratio. The mAP$^{50}$ (%) performance for each class is also reported.*

will become large. This means the generated pseudo-labels from our model are more reliable. For Table 6, comparing Naive-Student and PseudoProp, we can find the proposed SWBF method can be well adapted to the teacher-student semi-supervised learning framework. For Table 7, when we increase pseudo-labeled data size, both model performances will be decreased. The reason is that more pseudo-labeled data indicates more noise will be inserted and used in the training procedure. However, we can find our method can also get the best performance in mAP$^{50}$.

### 3.2. Additional Visual Results

We compare the visual results in Figure 2, 3, 4, and 5, for the ground truth, Naive-Student, VideoProp, and our proposed PseudoProp respectively on the Cityscapes validation dataset. From these figures, we can see that our PseudoProp model can eliminate miss and false detections. This means the pseudo-labels generated by our model are more robust.


Figure 2. *Visual comparison for the ground truth, Naive-Student, VideoProp, and our proposed PseudoProp on Cityscapes.*

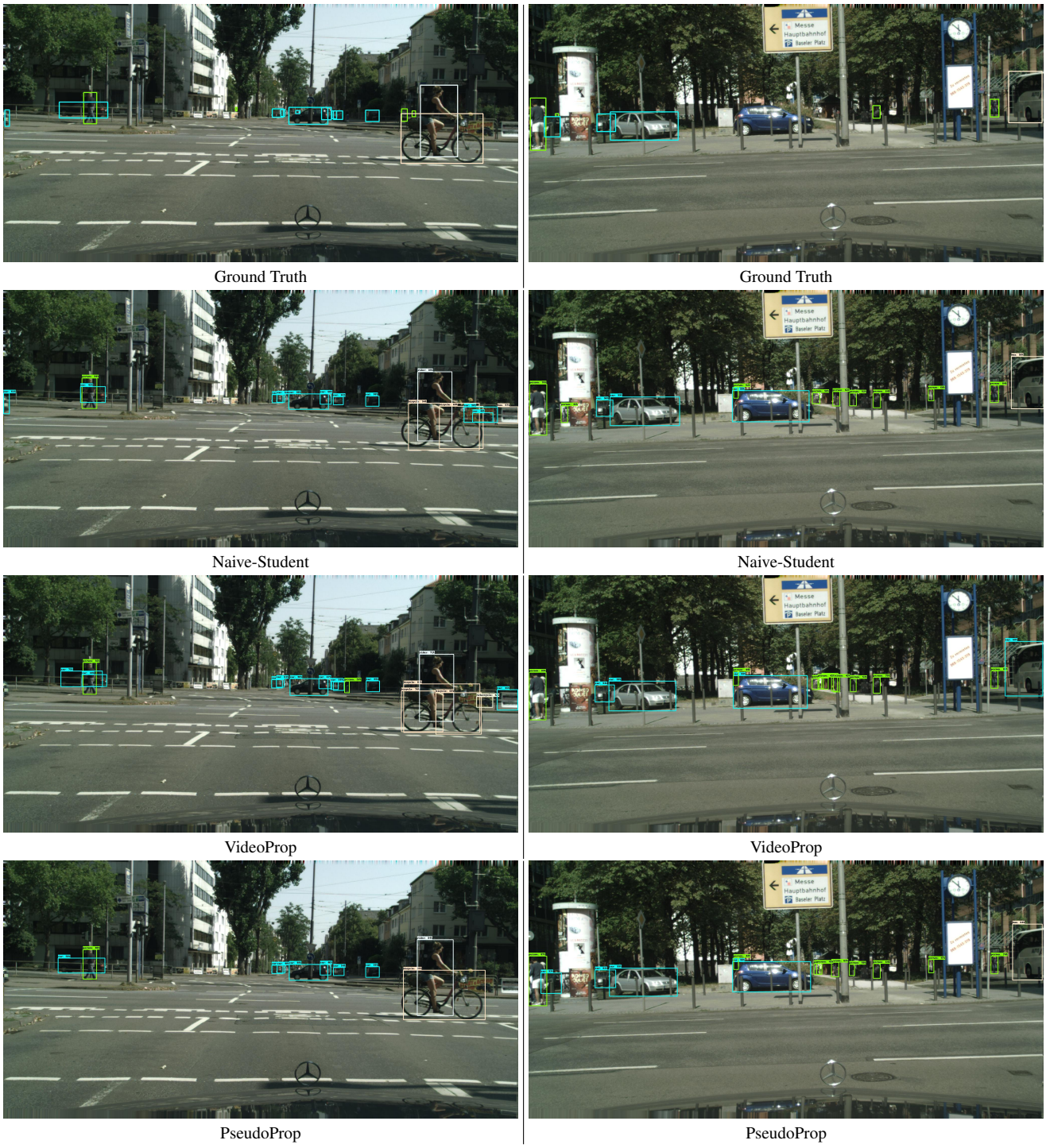



Figure 3. *Visual comparison for the ground truth, Naive-Student, VideoProp, and our proposed PseudoProp on Cityscapes.*

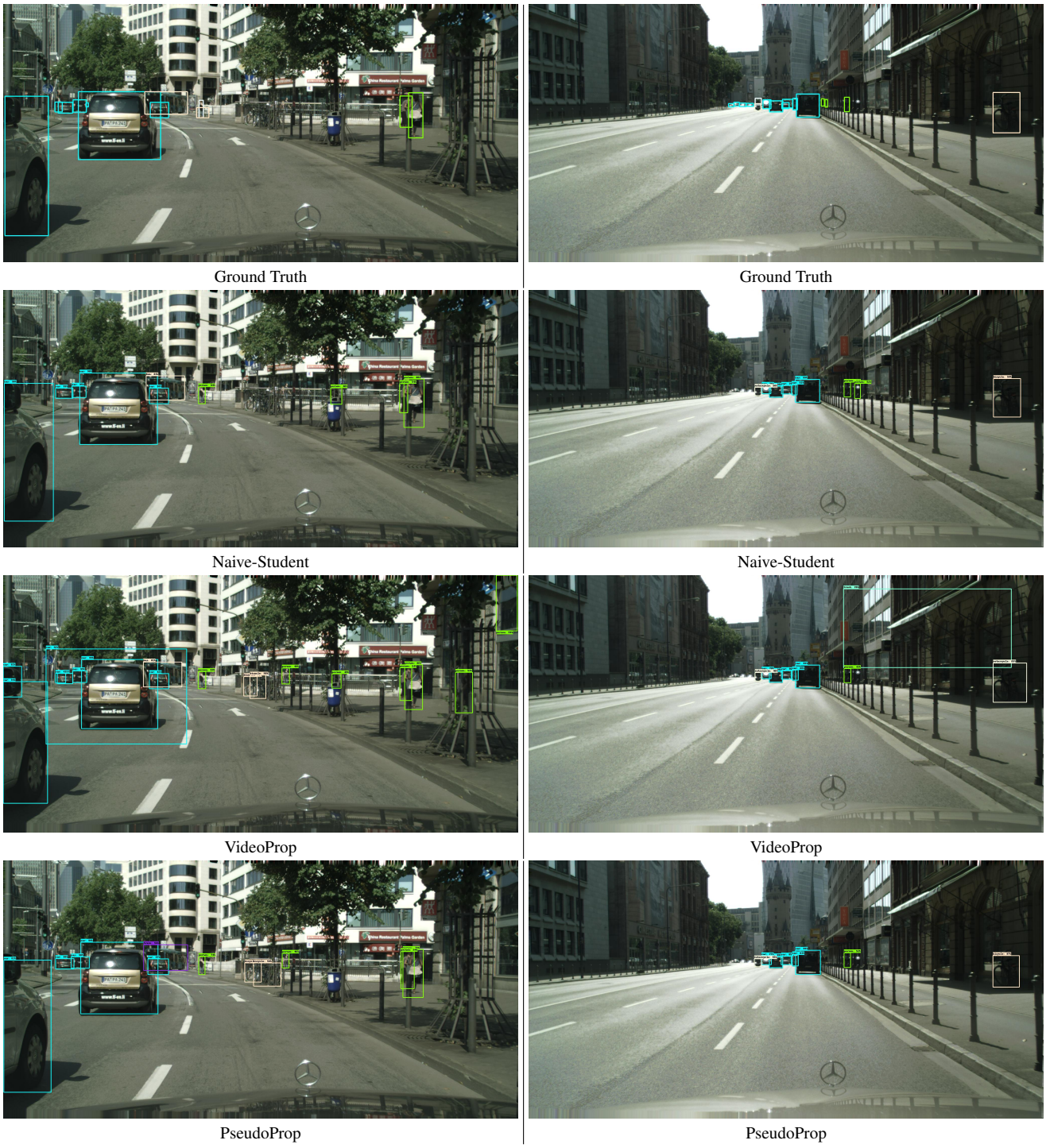



Figure 4. *Visual comparison for the ground truth, Naive-Student, VideoProp, and our proposed PseudoProp on Cityscapes.*


Figure 5. *Visual comparison for the ground truth, Naive-Student, VideoProp, and our proposed PseudoProp on Cityscapes.*

## References


[1] Marius Cordts, Mohamed Omran, and et al. The Cityscapes dataset for semantic urban scene understanding. In *CVPR*, 2016. 1

[2] Fitsum A. Reda and et al. SDC-Net: Video prediction using spatially-displaced convolution. In *ECCV*, 2018. 1

[3] Roman Solovyev and et al. Weighted boxes fusion: Ensembling boxes from different object detection models. *IVC*, 2021. 2