Supplementary Material

1. Implementation Details

In this section we provide the implementation details for our model using the three datasets: 50salads [8], MPII Cooking 2 [7], GTEA [1]. For all the three datasets, we train the model in 3 steps:

1. Until epoch 30 we use the single frame timestamps.
2. From epoch 31 to 50 we use the pseudo ground-truths generated from the timestamps and HOI.
3. From epoch 51 to 70 we train using the fine-tuned action boundaries using HOI.

Regarding Tables 4 and 5 of the main paper, which show variations of our method for ablation studies, this is how these variations correspond to the above steps:

- The "loss+pg" method of Table 4 corresponds to using only the first two steps above, and skipping the third step.
- The "loss+pg+ft" method corresponds to using all three steps as described above.
- The "loss+ft" method of Table 5 corresponds to using only steps 1 and 3, and skipping step 2. So, for the "loss+ft" version, from epoch 31 to 50 we train using the fine-tuned action boundaries using HOI, and we stop training when epoch 50 is done.

The best performance in all three datasets was dependent on the pseudo ground-truth generation parameters σ and τ as explained in the section titled *"HOI influenced Pseudo Groundruth"* of the main paper. For each dataset, the values for σ and τ were chosen automatically, using cross-validation. These are the values we used:

- For 50Salads, $\sigma = 30$ and $\tau = 30$.
- For GTEA, $\sigma = 10$ and $\tau = 75$.
- For MPII Cooking 2, $\sigma = 15$ and $\tau = 15$.

2. Impact of frame selection on performance

The HOI pseudo ground-truths are generated around the annotated frame-level timestamp. To check the system's sensitivity on the initialization of these frame-level annotations, we randomly selected timestamp frames for each action segment and created 10 unique sets of timestamp annotations and their respective HOI pseudo ground-truths for every video. We trained independent models with these newly generated annotations and performed this experiment for GTEA and 50Salads dataset and Table 1 illustrates the mean and standard deviation for each performance metric for these 10 models trained on unique timestamps and HOI pseudo ground-truths. It can be seen that the mean values for these performance metric are still better than the baseline despite random initialization and shows the system can still perform better even if timestamps are annotated randomly.

3. Importance of Pseudo-Ground Truths Using HOI

The information below describes alternative training strategies that we have evaluated. These strategies consist of different choices and orderings among the following modules:

- **Module a**: This is the module described in Section 3.2.1 of the main paper, which generates and uses pseudo-ground truth labels κ. As described earlier, in the implementation details section of this supplementary document, this module is used (in the normal version of our method) for training in epochs 31 to 50.
- **Module b**: This is the "fine-tuning" module described in Section 3.2.2 of the main paper. As described earlier, in the implementation details section of this supplementary document, this module is used (in the normal version of our method) for training in epochs 51 to 70.
- **Module b'**: This is a replacement of the "fine-tuning" module described in Section 3.2.2 with the original boundary detection method used in [5]. In the system overview of Figure 2 of the main paper, this variant

Dataset	F1@{10,25,50}			Edit	Acc
50Salads	76.6 ±0.6	74.2 ±0.6	63.1 ±0.8	69.5 ±0.5	76.2 ±0.3
GTEA	81.4 ±0.9	78.0 ±1.2	61.5 ±1.4	75.5 ±1.3	70.2 ±0.6

Table 1. Variation in performance using 10 unique combinations of randomly generated frame-level annotations. Number to the left of ± indicates the mean for 10 runs and to the right indicates standard deviation

Training Type	F1@{10,25,50}			Edit	Acc
50Salads					
a then b	**77.3**	**75.2**	**63.6**	**69.8**	**75.8**
b then a	73.2	70.1	58.2	64.7	73.8
GTEA					
a then b	**82.1**	**78.7**	**63.0**	**74.8**	**70.4**
b then a	73.6	66.6	49.4	68.8	61.4
MPII Cooking2					
a then b	**44.9**	**40.6**	**28.8**	**43.5**	**51.3**
b then a	35.0	30.1	19.6	30.8	39.5

Table 2. Variation in training using labels generated by pseudo ground-truths generated using HOI (a) and boundary detection using HOI (b)

would correspond to cutting the link between binary labels α and the "primary labels generator".

Changing the order between module a and module b In the standard version of our method, as explained earlier, we train using module a in epochs 31-50, and we train using module b in epochs 51-70. We evaluated switching this order. This variation is denoted on Table 2 as "b then a". Essentially, in this variation we train using module b in epochs 31-50, and we train using module a in epochs 51-70. Table 2 shows the results of this variation. We see that using module a first and module b second gave better performance for all three datasets.

Replacing our fine-tuning module with the original boundary detection of [5].

We also evaluated a variant where we use module b' (the original boundary detection module of [5]) instead of our finetuning module (module b). We tried both possible orderings in training (module a in epochs 31-50 followed by module b' in epochs 51-70, and the other way around).

Table 3 illustrates the performance differences of these variations. It can still be seen that training the network first from epoch 31 to 50 using pseudo ground-truths from HOI helps the network perform better for all three datasets. We can also see, by comparing the "a then b" results in Table 2 with the "a then b' " results of Table 3 that module b, which is one of our contributions, leads to better accuracy than module b' which is the corresponding component in [5].

Training Type	F1@{10,25,50}			Edit	Acc
50Salads					
a then b'	**76.5**	**74.4**	**62.6**	**69.3**	**75.7**
b' then a	71.8	69.0	57.8	64.9	73.7
GTEA					
a then b'	**79.9**	**75.5**	**58.1**	**74.2**	**68.2**
b' then a	73.4	65.7	45.7	70.7	60.3
MPII Cooking2					
a then b'	**44.4**	**40.0**	**28.3**	**42.1**	**50.5**
b' then a	35.9	31.4	20.6	32.6	39.8

Table 3. Variation in training using labels generated by boundary detection without using HOI (b') and pseudo ground-truths generated using HOI (a)

σ (pixels)	F1@{10,25,50}			Edit	Acc
10	80.5	77.2	59.7	74.9	68.9
20	**82.3**	**78.8**	60.5	**76.9**	69.9
25	81.4	78.0	**61.2**	73.8	**70.0**
30	80.3	75.2	59.3	74.4	69.0
35	81.0	77.4	58.6	73.2	68.4
40	77.6	72.2	56.0	72.5	67.2

Table 4. Performance Impact on varying Spatial threshold, σ in pixels with Temporal window $\tau = 30$ for GTEA dataset.

4. Impact of spatial and temporal thresholds

The pseudo ground-truth generated is controlled by 2 variables τ and σ. Variable τ controls the temporal window in which the algorithm finds the bounding box of interaction. Table 4 illustrates the impact of performance by keeping the temporal window constant at $\tau = 30$ frames and varying spatial threshold σ from 10 to 40 pixels. It can be seen that lower spatial thresholds of 10 or 20 pixels performed better as they ensure consideration of smaller movements during the interaction. Table 5 refers to the performance of varying the temporal window τ from 15 frames to 90 frames on GTEA dataset at a fixed spatial threshold σ of 30 pixels. It can be seen that the smaller window of 15 frames performs better as it will avoid overshoot of more frames to re-labelled incorrectly.

5. Accuracy of the generated pseudo ground-truth using HOI

We compared the quality of the pseudo ground-truths generated using HOI and single timestamp with the actual frame-wise ground-truth labels of the datasets. The metrics used were percentage count (%Count) of the frames where the algorithm labelled a frame with a valid action label. Note this analysis was not used to decide spatial and temporal thresholds. Thresholds were solely decided on the

τ (frames)	F1@{10,25,50}			Edit	Acc
15	**80.8**	**76.4**	**60.6**	**75.3**	**69.4**
30	80.3	75.2	59.3	74.4	69.0
45	80.4	76.0	58.4	74.8	68.1
60	79.2	73.3	58.0	73.4	68.6
75	77.2	73.6	56.6	70.9	67.3
90	78.4	75.3	57.4	73.6	68.3

Table 5. Performance Impact on varying Temporal window, τ with Spatial threshold, σ = 30 for GTEA dataset

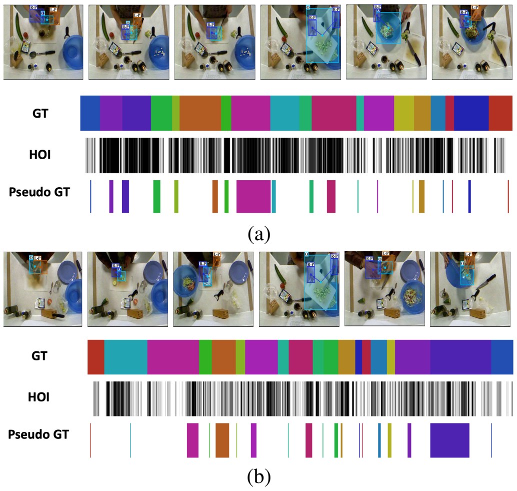

Figure 1. Human Object Interaction Detections (HOI) and the corresponding pseudo ground-truth generation for 50Salads Dataset

network's cross-validation performance. Using those valid labels, we measured how many of those frame-wise labels were accurate when compared to the ground-truth (%Acc).

Table 6 illustrates these values by using the variation of spatial threshold in pixels and keeping temporal window (τ) constant (75 frames for GTEA and 30 frames 50Salads). Increasing σ will enable the algorithm to track the HOI bounding box in neighboring frames at a coarser level, thus enabling the system include more frames in the same action, but the accuracy of these frames drops despite the increase in %Count.

Similarly Table 7 illustrates the same metrics using the variation of Temporal window τ and keeping spatial threshold σ constant (10 pixels for GTEA and 30 pixels for 50Salads). It can be seen that tracking the bounding boxes at longer lengths may cause the %Acc of the labels to reduce, but increase the %Count.

Thus, a good balance of accurate frames that last for longer duration is required and this will vary according to the dataset as some might have fine-grained actions, while others may long duration actions.

σ (pixels)	%Acc / %Count GTEA (τ=75)	%Acc / %Count 50Salads (τ=30)
10	**66.66/58.26**	95.50/8.79
20	54.28/72.06	94.94/15.21
25	48.73/78.88	94.80/18.05
30	42.15/82.89	**93.45/21.00**
35	39.29/86.43	93.04/24.55
40	36.36/89.23	91.83/28.18

Table 6. Variation frame-wise accuracy and count of frames using σ keeping τ constant for 50Salads and GTEA dataset. Highlighted values indicates the setup where the network gave best test performance.

τ (frames)	%Acc/%Count GTEA (σ=10)	%Acc/%Count 50Salads (σ=30)
15	70.88/56.06	94.36/18.22
30	70.29/57.23	**93.45/21.00**
45	70.32/57.80	92.01/23.99
60	68.24/57.31	90.92/26.25
75	**66.66/58.26**	87.75/27.53
90	65.36/58.73	87.65/29.38

Table 7. Variation frame-wise accuracy and count of frames using using τ keeping σ constant for 50Salads and GTEA dataset. Highlighted values indicates the setup where the network gave best test performance.

6. Impact of labels generated using HOI and action change

In Section 3.2.2 of the main paper, titled *Fine-tuning Action Changes*, we use the first frame of non-interaction frames in range $[t_{b_i,FW}, t_{b_i,BW}]$ to decide the boundary change location t_{b_i}. Table 10 illustrates performance of the network by picking up the last frame of non-interaction (last) as compared to the first frame of non-interaction (first) when the action boundary at t_{b_i} was at a location when HOI occured. It can be seen that using the first frame was the better strategy and gave better performance for all three datasets.

7. Timestamp vs. Full Supervision

Tables 8-9 compare the performance of the system with one of the best fully supervised methods as well as with other timestamp supervision methods. These tables illustrate that our method makes a significant step towards closing the accuracy gap between timestamp supervision and fully supervised methods.

8. Limitations

The proposed method makes several assumptions, and is limited by the extent to which those assumptions hold in

Supervision	Method	F1@{10,25,50}			Edit	Acc
Full	MSTCN++ [4]	80.7	78.5	70.1	74.3	83.7
	BCN [9]	82.3	81.3	74.0	74.3	84.4
	ASRF [2]	84.9	83.5	77.3	79.3	84.5
Timestamps	Seg model + plateau [6]	71.2	68.2	56.1	62.6	73.9
	Timestamp [5]	73.9	70.9	60.1	66.8	75.6
	Ours	**77.3**	**75.2**	**63.6**	**69.8**	**75.8**

Table 8. Results with different levels of supervision on 50Salads.

Supervision	Method	F1@{10,25,50}			Edit	Acc
Full	MSTCN++ [4]	88.8	85.7	76.0	83.5	80.1
	BCN [9]	88.5	87.1	77.3	84.4	79.8
	ASRF [2]	89.4	87.8	79.8	83.7	77.3
Timestamps	Seg model + plateau [6]	74.8	68.0	43.6	72.3	52.9
	Timestamp [5]	78.9	73.0	55.4	72.3	66.4
	Ours	**82.1**	**78.7**	**63.0**	**74.8**	**70.4**

Table 9. Results with different levels of supervision on GTEA.

Type	F1@{10,25,0}			Edit	Acc
GTEA					
first	**82.1**	**78.7**	**63.0**	**74.8**	**70.4**
last	81.1	78.1	60.9	74.8	69.9
50Salads					
first	**77.3**	**75.2**	**63.6**	**69.8**	**75.8**
last	75.9	73.8	61.7	68.8	75.4
MPII Cooking 2					
first	**44.9**	**40.6**	**28.8**	**43.5**	**51.3**
last	45.4	40.4	27.7	42.3	49.7

Table 10. Variation in Fine tuning Boundary detection using action change detection and choosing first /vs last non HOI detected frame in range $[t_{b_i,FW}, t_{b_i,BW}]$

a specific dataset. One assumption is that each video displays a single human performing activities involving interaction with objects. This assumption is relevant in many real-world applications, and it is true in commonly used datasets, such as the ones we have used in our experiments. At the same time, clearly there can be action recognition domains where this assumption does not apply. For example, this assumption would not apply for distinguishing between activities such as "walking" and "running".

Also low resolution and dark condition videos, that appear for example in the Breakfast Dataset [3], will not benefit from this approach as the HOI detector fails to detect interactions. The pseudo ground-truth generation can be extended further by using off-the-shelf object detectors and tracking those bounding boxes from the extended interaction frames. Other temporal modelling systems like transformers can be used to improve the performance. The system currently utilizes the idea of interactions to generate the pseudo ground-truths. Future work can involve extraction of features inside the interaction bounding box which can provide more information to the network.

References

[1] Alireza Fathi, Xiaofeng Ren, and James M Rehg. Learning to recognize objects in egocentric activities. In *CVPR 2011*, pages 3281–3288. IEEE, 2011. 1

[2] Yuchi Ishikawa, Seito Kasai, Yoshimitsu Aoki, and Hirokatsu Kataoka. Alleviating over-segmentation errors by detecting action boundaries. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2322–2331, 2021. 4

[3] Hilde Kuehne, Ali Arslan, and Thomas Serre. The language of actions: Recovering the syntax and semantics of goal-directed human activities. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 780–787, 2014. 4

[4] Shi-Jie Li, Yazan AbuFarha, Yun Liu, Ming-Ming Cheng, and Juergen Gall. Ms-tcn++: Multi-stage temporal convolutional network for action segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 2020. 4

[5] Zhe Li, Yazan Abu Farha, and Jurgen Gall. Temporal action segmentation from timestamp supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8365–8374, 2021. 1, 2, 4

[6] Davide Moltisanti, Sanja Fidler, and Dima Damen. Action recognition from single timestamp supervision in untrimmed videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9915–9924, 2019. 4

[7] Marcus Rohrbach, Anna Rohrbach, Michaela Regneri, Sikandar Amin, Mykhaylo Andriluka, Manfred Pinkal, and Bernt Schiele. Recognizing fine-grained and composite activities using hand-centric features and script data. *International Journal of Computer Vision*, 119(3):346–373, 2016. 1

[8] Sebastian Stein and Stephen J McKenna. Combining embedded accelerometers with computer vision for recognizing food preparation activities. In *Proceedings of the 2013 ACM international joint conference on Pervasive and ubiquitous computing*, pages 729–738, 2013. 1

[9] Zhenzhi Wang, Ziteng Gao, Limin Wang, Zhifeng Li, and Gangshan Wu. Boundary-aware cascade networks for temporal action segmentation. In *European Conference on Computer Vision*, pages 34–51. Springer, 2020. 4