Technical Report for EgoTracks in Ego4D Challenge 2023

Mingfang Zhang[1,2*], Yuan Yin[1*], Yifei Huang[1,2†], Yoichi Sato[1]
[1]The University of Tokyo, [2]Shanghai AI Laboratory
{mfzhang,yinyuan,hyf,ysato}@iis.u-tokyo.ac.jp

Abstract

*In this report, we address the problem of egocentric visual object tracking (EgoTracks) through a **long-term weighted boxes fusion method**. Compared to traditional object tracking task, EgoTracks is more challenging because of frequent large camera motions, hand-object occlusions, state changes of interacting objects, and the naturally long time span. We observe that previous methods designed for traditional object tracking show unstable results. Even when using the same model, these methods struggle to generate consistent and reliable predictions for identical video frames. Therefore, we ensemble the box prediction of multiple independent trackers by exploring the usage of predicted confidence scores to generate more reliable results. The experimental results show that our simple and effective method can achieve the state-of-the-art result in the Ego4D dataset.*

1. Introduction

Egocentric visual object tracking (EgoTracks) [6] is a novel yet important computer vision task. Given an egocentric video and a visual template of an object, this task aims to localize the object in each frame of the video with a bounding box. It has great potential for improving the performance of wearable devices, e.g. head mounted camera and AR/VR devices.

Due the wearable camera setting and frequent hand manipulations, EgoTracks faces problems such as frequent large camera movements, occlusions between the hand and objects, state changes of interacting objects, and the naturally extended duration. As a result, methods that are weak at handling rapid changing objects and long-term predictions can only give unstable and inferior results in this challenging task.

Even when using the same tracker on identical video clips with slight data augmentation, this model cannot ensure a consistent tracking prediction in terms of the location of the bounding boxes in each frame. Motivated by this observation, we propose to integrate the predictions from multiple trackers by model ensembling to make the predictions more reliable. By combining the output of multiple independent models that may randomly excel at different aspects of the EgoTracks task, better results are achieved than any individual model alone. Furthermore, we maximize the utilization of the predicted confidence scores for each frame to enhance the accuracy of our tracking predictions. First, by considering the dynamic nature of target disappearances and reappearances across multiple frames [2], we integrate the confidence scores of multiple frames to determine the target visibility, i.e., if the target is visible in the predicted bounding box. Then, we perform a weighted fusion of the bounding boxes predicted by multiple trackers, taking into account the corresponding confidence scores, to generate an optimized bounding box for each frame where the target is discernible.

In the following sections, we first describe our proposed method, and then show the experimental results of our methods on Ego4D dataset [3].

2. Method

In this section, we present our long-term weighted boxes fusion method for the EgoTracks task. The task involves analyzing an egocentric video along with a visual template of an object, with the objective of accurately determining the object's location and providing a confidence score for its presence in each frame. To achieve this, it is necessary to generate a bounding box encompassing the object and assign a corresponding confidence score for object detection in every frame of the video. The following steps outline our methodology in detail.

1. Baseline Trackers Execution. Given the baseline model, STARK [7], we selected N tracker models, $\tau^{(1)}, \tau^{(2)}, ..., \tau^{(N)}$, saved at different epochs during its training process. Note that there are various approaches to select multiple models and we observe that even simply using the same network architecture, our method is effective. Specifically, at every frame t, τ

†Project lead, *Co-first author.

receives input the frame F_t and outputs bounding box b_t and confidence score c_t:

$$b_t^{(n)}, c_t^{(n)} \leftarrow \tau^{(n)}(F_t). \tag{1}$$

2. Target Visibility Determination. Inspired by [2], our method determines whether $\tau^{(n)}$ are correctly following the target, i.e., if it is visible in their predicted bounding boxes. To determine the visibility status $\hat{p}_t^{(n)} \in 0, 1$ of the target in individual frames, we employ a binary classification by applying a threshold of 0.5 to binarize the predicted confidence values:

   $$\hat{p}_t^{(n)} \leftarrow c_t^{(n)} \geq 0.5. \tag{2}$$

   Considering the dynamic nature of target disappearances and reappearances across multiple frames, we take into account the visibility status $\hat{p}_t^{(i)}$ in the last $\hat{T}$ frames to determine a more reliable presence of the target. Specifically, we follow [2] to define the target as visible within the bounding box $b_t^{(n)}$ and assign $p_t^{(n)} = 1$ if the following conditions are met:

   $$p_t^{(n)} \leftarrow \sum_{i=0}^{\hat{T}} \hat{p}_{t-i}^{(n)} > \lfloor 0.75 \times \hat{T} \rfloor. \tag{3}$$

3. Weight Boxes Fusion. For predicted boxes that are not assigned positive p, we conduct a simple average of the results. For predicted boxes that are assigned positive p, we employ weighted boxes fusion [5] to stabilize the prediction results. The detailed steps are described in the following algorithm block.

Algorithm 1: Weighted Boxes Fusion

```
$B = \{b_{t0}^{(n)}\}, n = 1, 2, ..., N$ // predicted boxes at $t_0$
$C = \{c_{t0}^{(n)}\}, n = 1, 2, ..., N$ // confidence scores
for $i$ in range($N$):
    if $p_{t0}^{(i)} \neq 1$: remove $b_{t0}^{(i)}$,$c_{t0}^{(i)}$ from $B, C$
$L = \{\}$ // boxes clusters (buffer)
$F = \{\}$ // fused boxes (result)
for $b$ in $B$:
    if $\exists b_F$ at $F[pos]$, $s.t.\ IOU(b, b_F) > 0.55$:
        $L[pos].append(b)$
        // update $F[pos]$ with the $T$ elements in $L[pos]$
        $C = (\sum_{i=1}^{T} C_i)/T$
        $B = (\sum_{i=1}^{T} C_i \times B_i)/(\sum_{i=1}^{T} C_i)$
    else:
        $L.append([b]), F.append(b)$
```

Table 1. Comparison with baseline. Our long-term weighted boxes fusion method shows advantages employing multiple (3,6) trackers.

Method	F-score	Precision	Recall	AO
GlobalTrack [4]	20.40	31.28	15.14	23.63
LTMU [1]	27.46	**37.28**	21.74	29.33
STARK [7] (baseline)	30.48	34.70	27.17	35.99
STARK-Ours (3)	31.58	31.32	**31.84**	**42.08**
STARK-Ours (6)	**32.18**	35.85	29.18	38.12

3. Experiments

To verify the validity of our proposed method, we conduct experiments on Ego4D dataset [3]. We first compare our method with the baseline to show how our ideas contribute to the EgoTracks task. Then we present some visualization results of our proposed method.

3.1. Comparison with Baseline

In Tab. 1, we present a comprehensive comparison between our method and other existing approaches, including LTMU [1], GlobalTrack [4], and STARK [7]. Notably, our method demonstrates consistent improvements compared to the baseline method, STARK, substantiating its efficacy. Furthermore, we conduct an ablation study to determine the impact of the number of employed trackers in our method. Encouragingly, the results reveal that our approach does not rely on a large number of trackers, thereby ensuring its efficiency.

3.2. Qualitative Results

In this section, we show some visualization results of our methods on Ego4D dataset. We consider two situations as *positive* samples: 1) the object to be tracked is in the frame, and we output a good bounding box with high confidence score. 2) the object to be tracked is not in the frame, and we output a random bounding box with very low confidence score. Similarly, we consider two situations as *negative* samples: 1) the object to be tracked is in the frame, but we output a bad bounding box with high confidence score. 2) the object to be tracked is not in the frame, but we output a random bounding box with high confidence score. We show two groups of positive and negative samples from two test videos in Fig. 1.

From the positive samples, we can see that our method can locate and keep track of the objects with high confidence when they appear in the frames. And it is able to output a low confidence when the objects disappear from the scene, which is also a desirable characteristic. As for the negative samples, the upper one corresponds to the first negative case where the predicted bounding box is offset

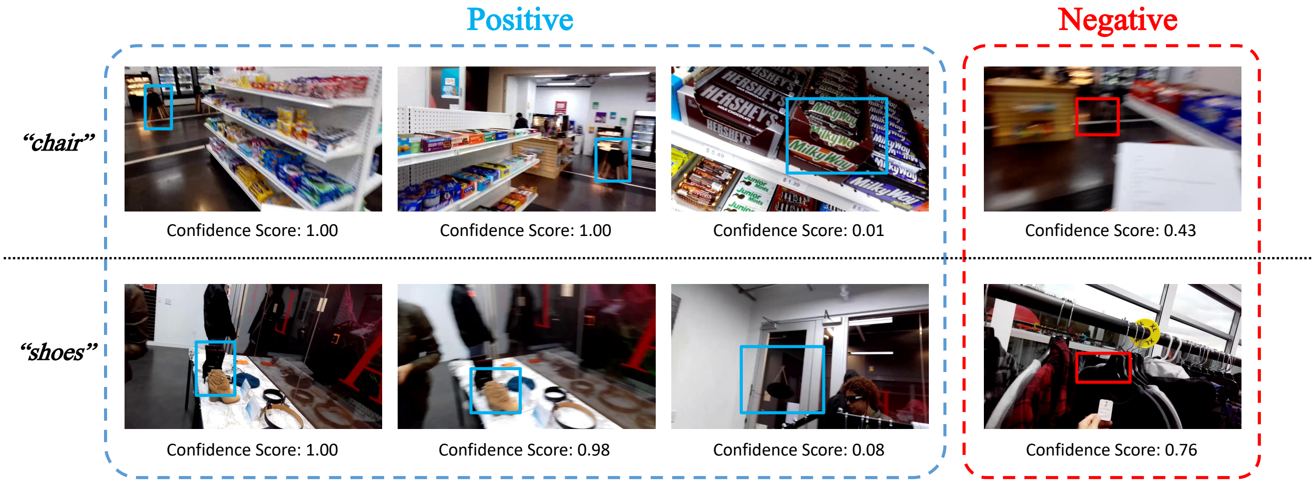

Figure 1. Qualitative results on Ego4D. We present two groups of *positive* (blue) and *negative* (red) samples from its test set. The upper one shows samples from the results of tracking **chair**, and the bottom one shows samples from the results of tracking **shoes**.

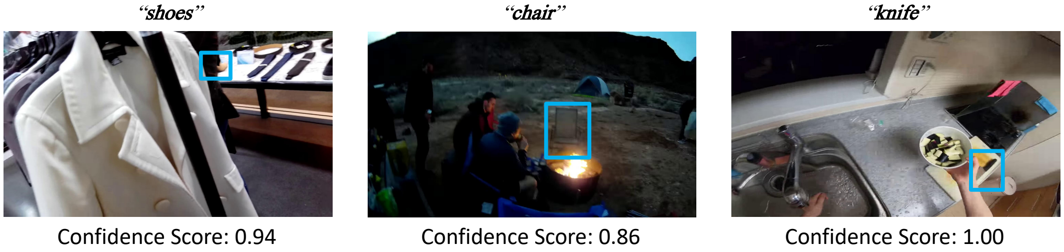

Figure 2. Here we show three examples in which there are obvious occlusions on the objects to be tracked. As can be seen in the figures, our method successfully keeps tracking the objects regardless of the occlusions. We owe this to the long-term guidance in our weight boxes fusion algorithm.

but the confidence score is still high. This might be because the frame is too blurry for the tracker to localize the accurate position of the target object. The bottom one corresponds to the second negative case where our method outputs a bounding box with high confidence score when there is actually no object in the scene. This might be because the tracker confused the target object (shoe) with other objects (a part of the clothes hanger). The failure cases of our method usually occur when the tracker can not detect the target objects well because of blurry appearance or complicated environments. So a promising way to make our method better is to improve our model's ability of detecting objects.

Besides, we also find some examples where our method successfully track the target objects even if there are obvious occlusions. We have three examples here, showing that the shoe is blocked by some other objects in the front. But our method keeps a good tracking of the target objects. We owe this to the long-term information we consider in the weight boxes fusion algorithm.

4. Conclusion

In this report, we propose a long-term weighted boxes fusion method to address the problem of egocentric visual object tracking. By combining the output of multiple independent models that may randomly excel at different aspects of this task, better results are achieved than any individual model alone. We also explore the usage of the predicted confidence scores to further stabilize the bounding box predictions. Qualitative and quantitative experiments are conducted to verify our method's effectiveness.

References

[1] Kenan Dai, Yunhua Zhang, Dong Wang, Jianhua Li, Huchuan Lu, and Xiaoyun Yang. High-performance long-term tracking with meta-updater. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6298–6307, 2020. 2

[2] Matteo Dunnhofer, Kristian Simonato, and Christian Micheloni. Combining complementary trackers for enhanced long-

term visual object tracking. *Image and Vision Computing*, 122:104448, 2022. 1, 2

[3] Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, et al. Ego4d: Around the world in 3,000 hours of egocentric video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18995–19012, 2022. 1, 2

[4] Lianghua Huang, Xin Zhao, and Kaiqi Huang. Globaltrack: A simple and strong baseline for long-term tracking. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 11037–11044, 2020. 2

[5] Roman Solovyev, Weimin Wang, and Tatiana Gabruseva. Weighted boxes fusion: Ensembling boxes from different object detection models. *Image and Vision Computing*, 107:104117, 2021. 2

[6] Hao Tang, Kevin Liang, Kristen Grauman, Matt Feiszli, and Weiyao Wang. Egotracks: A long-term egocentric visual object tracking dataset. *arXiv preprint arXiv:2301.03213*, 2023. 1

[7] Bin Yan, Houwen Peng, Jianlong Fu, Dong Wang, and Huchuan Lu. Learning spatio-temporal transformer for visual tracking. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10448–10457, 2021. 1, 2