Supplementary
EgoPoints: Advancing Point Tracking for Egocentric Videos

Ahmad Darkhalil[1] Rhodri Guerrier[1] Adam W. Harley[2] Dima Damen[1]
[1]University of Bristol [2]Stanford University
http://ahmaddarkhalil.github.io/EgoPoints

A. Implementation Details

Here we provide the required additional implementation details to replicate our results.

PIPs++. We adopt the $200k$ iterations checkpoint provided by [6], which is pre-trained on the PointOdyssey dataset. We fine-tune for a further $45k$ iterations using the data mix described in the main paper. Specifically, for each batch, there is a 65% chance of sampling K-EPIC sequences (where half of these are looped for increased re-identifications) and a 35% chance of sampling from the original PointOdyssey training dataset. We use a sequence length of 36 frames for PointOdyssey and 24 frames for K-EPIC. We resize K-EPIC sequences to 384x512. We use a batch size of 2, 128 trajectories per sequence and a constant learning rate of $2.8e^{-7}$ on a single V100 32GB GPU.

CoTracker [5]. We make use of the CoTracker-v2 checkpoint provided by the authors. This was trained for $50k$ iterations on sequences of 24 frames from the TAP-Vid-KUBRIC dataset [2] and utilises the virtual tracks added in the second version of the work. We then fine-tune this model further with the same data mix as PIPs++ above, between K-EPIC and TAP-Vid-KUBRIC. We use a batch size of 1, 196 trajectories per sequence and a learning rate of $5e^{-5}$ with a linear 1-cycle[1] learning rate schedule following CoTracker training. We use two V100 32GB GPUs. We train CoTracker with virtual tracks of 64 following the provided code [5].

CoTracker3 [4]. Similar to CoTracker-v2, we evaluate CoTracker3 at 384x512 resolution. We use the pre-trained online model provided by the authors.

LocoTrack [1]. We evaluate models on their native (training) resolution which is 256x256. Due to memory constraints, we set a maximum limit of 1,000 frames during inference. For sequences exceeding this limit, we sample equally spaced frames ensuring we always include the annotated frames.

BootsTAPIR Online [3]. We use the sequential, causal version of BootsTAPIR, implemented in PyTorch and provided at the official github[2]

B. Qualitative Examples

Three examples of predictions on EgoPoints annotations for both PIPs++ [6] and CoTracker [5], before and after fine-tuning, can be seen in Figure 1. It should be noted that we show the first and final evaluation frames for simplicity. However, each of these examples involve the camera wearer moving around the scene before revisiting the same location in the first frame. Therefore, they are particularly difficult re-identification scenarios for current SOTA models, as discussed in the main paper.

These examples demonstrate a clear improvement over the baselines. The first two examples are good examples of where PIPs++ [6] does better at re-identification than CoTracker [5]. The first example is 830 frames long and it is possible to see that fine-tuning on PIPs++ helps to successfully re-identify the yellow, blue and green points that were lost by the baseline.

The second example is another case of where PIPs++ improves more than CoTracker when fine-tuning. The dark purple, orange and dark green points are all successfully recovered when compared to the baseline. For CoTracker, although the orange and dark green points are tracked correctly after fine-tuning, while points at the bottom of the frame are lost.

The third sequence shows CoTracker performing better after fine-tuning. 5 of the 8 query points are tracked precisely and a sixth point (the dark purple) is tracked close to the ground truth.

We also show qualitative results using dense query grids for CoTracker [5] in Figure 2. In all examples the baseline can be seen to struggle with the complete grid during re-identification. After fine-tuning with K-EPIC, most points

[1]Leslie N Smith and Nicholay Topin. Super-convergence: Very fast training of neural networks using large learning rates. In Artificial Intelligence and Machine Learning for Multi-Domain Operations Applications, volume 11006, page 1100612. International Society for Optics and Photonics, 2019

[2]https://github.com/google-deepmind/tapnet

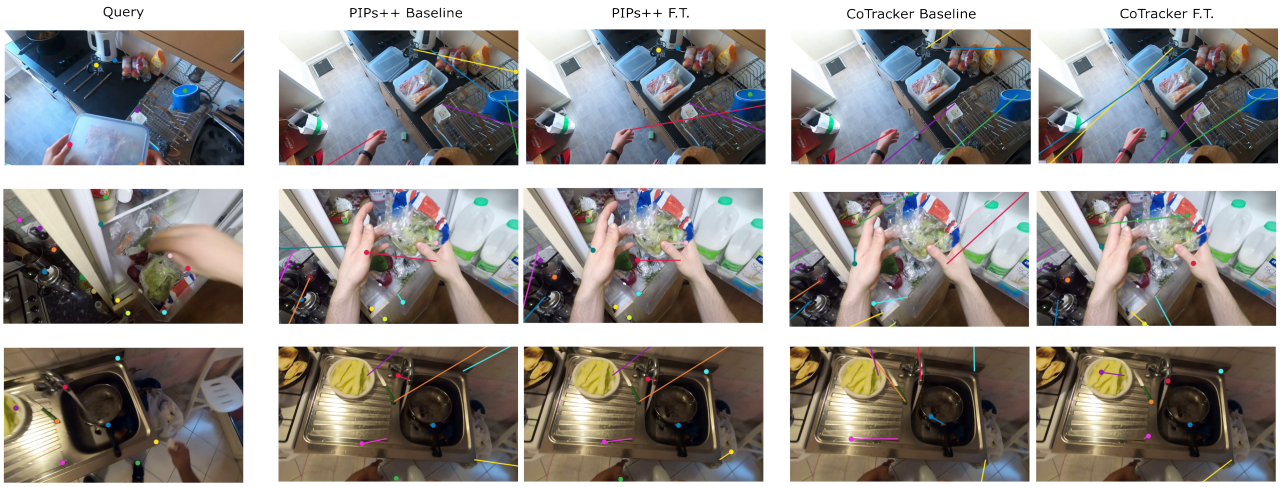

Figure 1. Three examples of EgoPoints evaluations before and after fine-tuning on PIPs++ and CoTracker. Dots represent initial points in the first column, and predictions in the other four columns. We plot a line connecting the prediction to the ground truth so as to show the difference. Points are correctly predicted if no line is attached. Points connected to the image boundary indicates the point is predicted out-of-view.

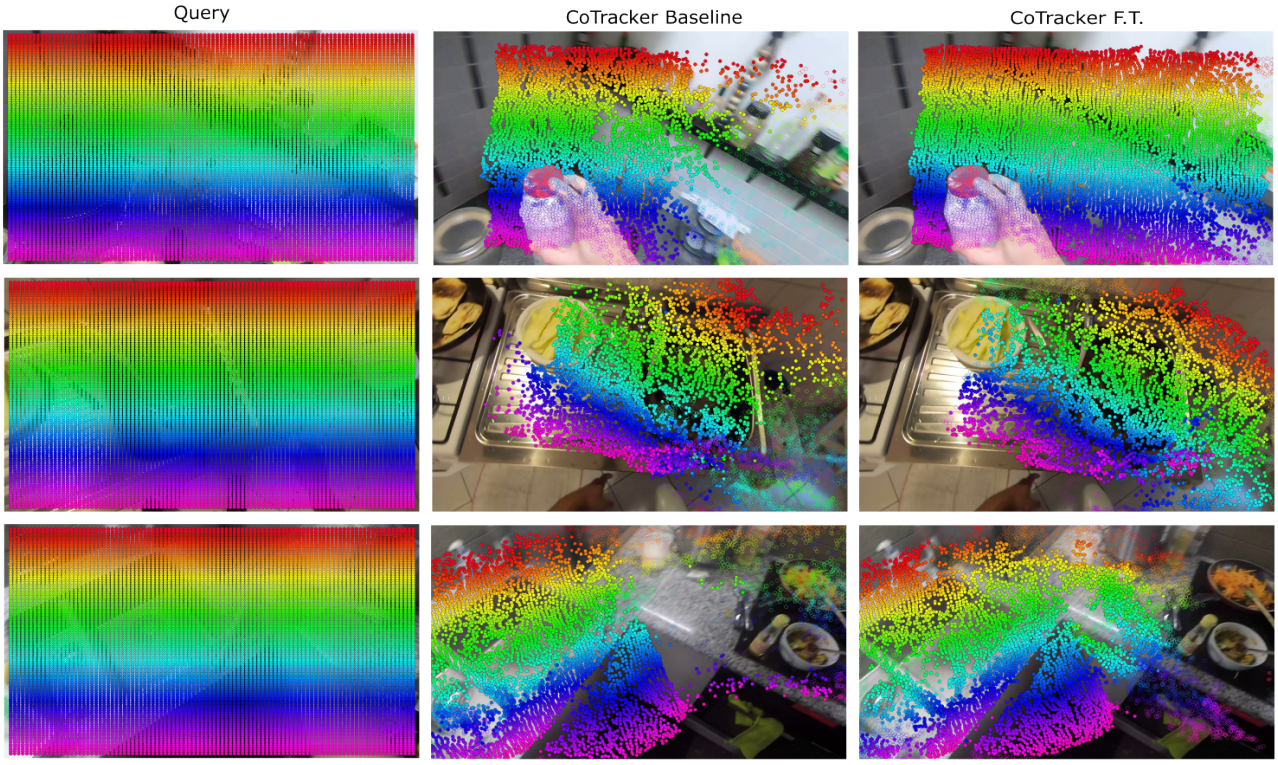

Figure 2. Examples of CoTracker results, before and after fine-tuning, on a dense grid of points (100x100).

are recovered. We share a video of these sequences on the project webpage.

In the main paper, we ablate the performance of fine-tuned models over sequence lengths. As an extension to this, we show here that fine-tuning improves performance across sequence lengths. Figure 3 shows average δ_{16} for

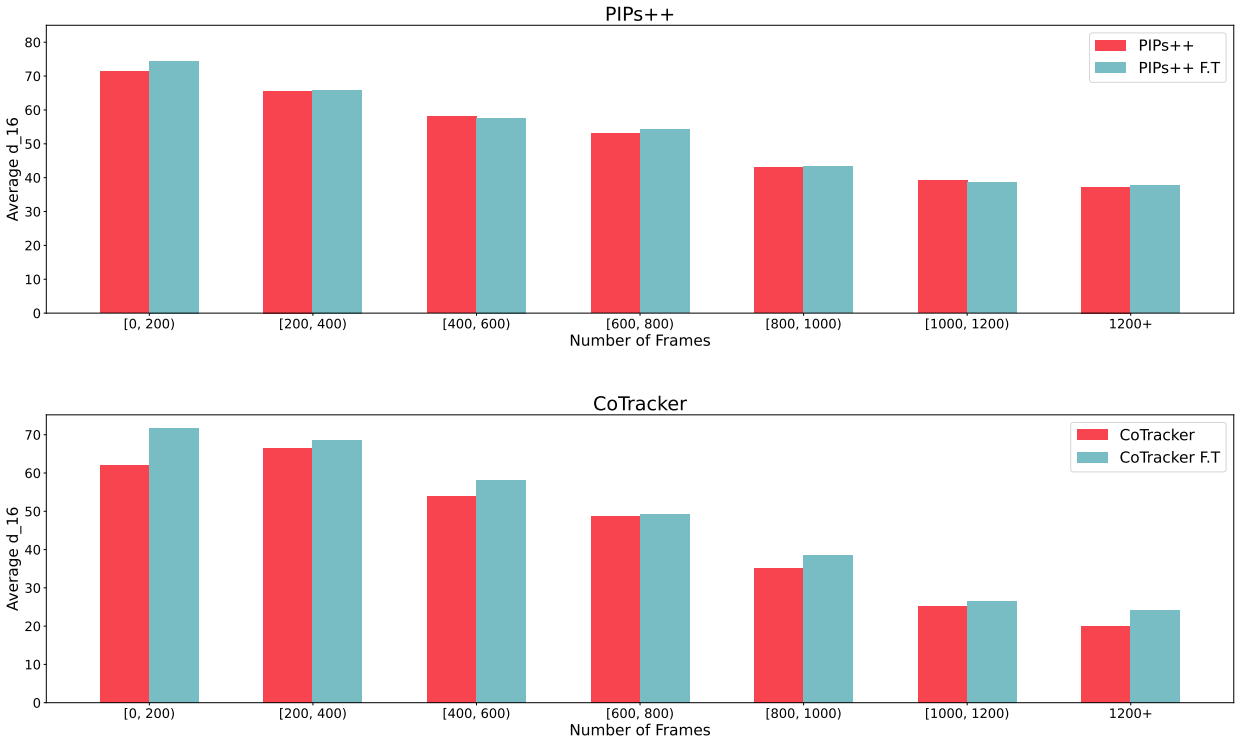

Figure 3. Average δ_{16} vs. sequence length of EgoPoints benchmark, before and after fine-tuning on PIPs++ [6] and CoTracker [5].

ranges of 200 frames. For PIPs++, performance is improved for short sequences ($<$ 200 frames) with comparable performance for sequences (2K-2.2K frames in length). On the other hand, CoTracker shows clearer improvements throughout.

References

[1] Seokju Cho, Jiahui Huang, Jisu Nam, Honggyu An, Seungryong Kim, and Joon-Young Lee. Local all-pair correspondence for point tracking. *ECCV*, 2024. 1

[2] Carl Doersch, Ankush Gupta, Larisa Markeeva, Adria Recasens, Lucas Smaira, Yusuf Aytar, Joao Carreira, Andrew Zisserman, and Yi Yang. Tap-vid: A benchmark for tracking any point in a video. *Advances in Neural Information Processing Systems*, 35:13610–13626, 2022. 1

[3] Carl Doersch, Yi Yang, Dilara Gokay, Pauline Luc, Skanda Koppula, Ankush Gupta, Joseph Heyward, Ross Goroshin, João Carreira, and Andrew Zisserman. Bootstap: Bootstrapped training for tracking-any-point. *arXiv preprint arXiv:2402.00847*, 2024. 1

[4] Nikita Karaev, Iurii Makarov, Jianyuan Wang, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. CoTracker3: Simpler and better point tracking by pseudo-labelling real videos. 2024. 1

[5] Nikita Karaev, Ignacio Rocco, Benjamin Graham, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. Cotracker: It is better to track together. *ECCV*, 2024. 1, 3

[6] Yang Zheng, Adam W Harley, Bokui Shen, Gordon Wetzstein, and Leonidas J Guibas. Pointodyssey: A large-scale synthetic dataset for long-term point tracking. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19855–19865, 2023. 1, 3