An assessment of Multi Object Tracking on low framerate conditions

Anis Yassine BEN MABROUK[1], Gabriele FACCIOLO[1], Rafael GROMPONE VON GIOI[1], Axel DAVY[1,2]

[1]Université Paris-Saclay, CNRS, ENS Paris-Saclay, Centre Borelli, 91190 Gif-sur-Yvette, France

[2]HGH Systèmes Infrarouges, 10 rue Maryse Bastié 91430 Igny, France

{anis.ben_mabrouk1,gabriele.facciolo,grompone,axel.davy}@ens-paris-saclay.fr

Résumé – La performance des méthodes de l'état de l'art en suivi multi-objets est généralement montrée sur des vidéos à fréquence image élevée, où les objets bougent très peu d'une image à l'autre. Il y a cependant des intérêts à travailler dans le cas plus difficile d'une fréquence faible ou alternativement de mouvements forts. Cet article étudie à quel point la performance de suivi est affectée par une diminution de fréquence image, et identifie les méthodes les plus adaptées à cet usage alternatif.

Abstract – The performance of state-of-the-art multiple object tracking methods is usually shown on high-framerate videos, for which objects move very little between two consecutive frames. Nonetheless, there is an interest in working in the harder scenario of low framerate or similarly, strong motion. This article studies how much tracking performance is affected by a decrease in image frequency, and identifies the methods that are better suited for this alternative use-case.

1 Introduction

The aim of Multi Object Tracking (MOT) is to assign a unique identifier for each target object in a video sequence, remaining consistent throughout the sequence in spite of hurdles such as occlusions or changes of scale. This problem recently gained much attention from the computer vision community, in part for its applications in self-driving cars, particle tracking in microscopy imaging and reliable camera surveillance.

Most methods are based on the *detect to track* paradigm, in which the problem is divided into detection and tracking steps. The detection step aims at detecting the target objects (for example persons or cars) in individual frames. Then, the tracking step tries to associate or pair the detections on one frame to the detections on another frame. The pairing can be done either using motion criteria such as Optical Flow or the Kalman filter [1, 17, 12], by re-identification (ReID) using appearance cues [14], or a mixture of both [15, 20, 18, 16]. Affinity scores may be used to measure similarity between two targets.

Achieving real-time performance on standard video framerate is an important challenge for MOT methods and the main focus of the literature. Yet, there is an interest in being able to track in low framerate videos [4] or, equivalently, when large motion is present with pronounced changes from frame to frame.

This work makes a short review of the recent state-of-the-art literature to assess the impact of the framerate on the performance. For this, low framerate sequences were emulated by frame sub-sampling and the different methods were compared on normal and sub-sampled sequences with an appropriate metric. The quality of the detection step has an impact as well as the association step; in order to make a fair comparison of the association steps, a second evaluation was performed using the ground truth detections instead of the internal detections. Our experiments show that most methods suffer a significant performance drop on low framerates, while some, particularly those that rely on visual cues, are more robust, see Figure 1.

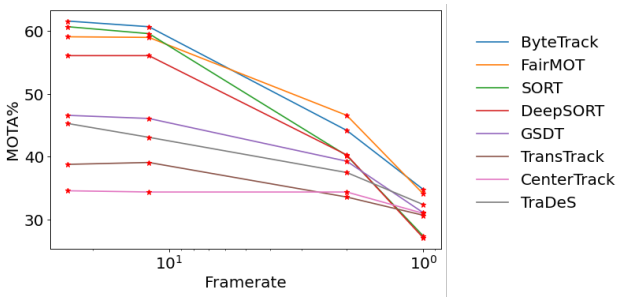


Figure 1: Multi object tracking accuracy (metric MOTA [11]) of different tracking methods on the MOT20-01 sequence [2] with different framerates (logarithmic scale). As the framerate is reduced, and thus the change between frames increases, all methods suffer a significant drop in performance.

This paper is organized as follows: Section 2 provides a description of the methods to be evaluated, while Section 3 explains the evaluation methodology. The results are presented and discussed in Section 4. Section 5 concludes the paper.

2 Evaluated methods

2.1 Detection backbones

There are various detection backbones that can be used for the detection step in each tracking method. For the deployed tracking methods, the detection backbones are:

CenterNet [21] This anchor-free detection network predicts a dense detection heatmap and has a regression head for the bounding box positions. The heatmap is learned to be maximal at the center of objects. According to [18], anchor-free networks are more adequate for extracting re-identification features.

YOLOX [3] is also an anchor free detector based on the same backbone as YOLOv5 which is an advanced CSPNet [13] with an additional PAN [6] head. On top of the backbone, there are two decoupled heads, one for regression and the other for classification. The regression head also has an intersection over union (IOU) aware branch and directly predicts bounding box details at each location of the heatmap.

Deformable DETR [22] is an object detector that uses transformers on top of a convolutional backbone to learn positional embeddings called objects queries. It employs multi-scale deformable attention to overcome limited resolution problems.

2.2 Tracking methods

SORT [1] has a simple and effective design: it retrieves the detections from an object detector, originally FrRCNN[9], then predicts the positions of the objects of the previous frame using a Kalman filter. Associations are conducted using the IOU between the predicted bounding boxes and the newly detected bounding boxes. Concretely, an optimal pairing that minimizes the total cost is obtained through the Hungarian algorithm [5].

DeepSORT [15] is an extension of SORT. It introduces a modification on the association step: its association cost is a linear combination of two terms. The first term relates to motion and is the Mahalanobis distance between the predicted position of the previous detection and the new detection, where the prediction is done using a Kalman filter. The second term, related to the appearances, is the cosine distance between the previous and the new detections on a separately learned ReID embedding.

ByteTrack[17] tracks all targets using a Kalman filter, similarly to SORT, but splits the association step into two incremental phases depending on the detection score. High score detections are paired in a first phase. Then, unmatched tracks of the first phase are matched with low score detections if the movement is well predicted by the Kalman filter. High confidence detections are kept even if they cannot be associated with previous tracks.

CenterTrack [20] supplements CenterNet [21] with a motion prediction branch that predicts the movement of detected objects from the previous frame. To do so, in addition to the previous and the current frames, the network is fed with the centers of the previous detections in the form of a heatmap. Tracking is then performed by predicting the new object centers.

FairMOT [18] builds on top of CenterNet by adding a re-identification branch. The re-identification branch consists of a convolutional layer added on top of the backbone feature that extracts Re-ID features for all locations. The Re-ID features are learned through a classification task. Tracking is done similarly to DeepSORT by computing the same association cost based on both motion and visual cues. The Re-ID features of the tracks are updated each time step.

TraDeS [16] is similar to FairMOT in the sense that it conducts detection and re-identification in a joint manner. It uses a cost volume association approach for the re-identification step. It also follows the same approach as CenterTrack [20], learning the object center displacements. The final tracking offset is conditioned on the cost volume computed and can be calculated as the dot product between the likelihoods obtained through the cost volume and the actual offset values. The tracking is performed by calculating this offset at object centers in each frame similarly to CenterTrack.

GSDT [14] Graph neural networks for Simultaneous Detection and Tracking (GSDT) falls under the category of joint detection and re-identification methods. It improves on previous joint approaches by adding a graph neural network to model the spatial-temporal relations between the detected objects, which helps to perform the association step.

TransTrack [12] is built on a Transformer based encoder-decoder framework. The method has one encoder to generate feature maps or keys from two consecutive frames and two decoders, one to infer object detections which outputs bounding boxes and the other for object tracking which predicts the location of previously seen objects. The association step is done by applying the Hungarian algorithm [5] to the IOU values between detection boxes and tracking boxes.

3 Experimental setup

In our experiments FairMOT [18], CenterTrack [20], TraDeS [16] and GSDT [16] use the DLA34 backbone from CenterNet, while TransTrack [12] uses Resnet-50. YOLOX is used for SORT, DeepSORT [15] and ByteTrack [17]. DeepSORT uses [7] for the Re-ID model with market-1501 [19] pretrained weights. All methods were trained on the MOT17 [8] training set, with FairMOT, CenterTrack, GSDT, TraDeS and TransTrack also being pretrained on the CrowdHuman dataset [10]. In all our experiments we use pre-trained network weights made available along with the public implementations.

The methods were evaluated on the challenging MOT20 [2] sequences, which present realistic scenarios with small targets and heavy occlusions. The training set of MOT20 [2] was used as it has available ground truth (unlike the test set), which was required to perform our custom evaluation. This is not problematic because none of the methods were trained on it. *We also note that MOT20-03 and MOT20-05 sequences have more challenging movement while MOT20-01 and MOT20-02 have harder occlusion cases.* We generated versions of each sequence at various framerates by sub-sampling: one frame out of x was kept, where $x = \frac{1}{sr}$ and sr is the sampling rate. For example, for a sampling rate of 0.5, one frame out of two was selected, starting from the first frame, see Figure 2.

Table 1: Multi object tracking accuracy (MOTA) [11] of the different tracking methods on the MOT20 [2] training sequences (original and undersampled).

Framerate	Sequence id	SORT[1]	DeepSORT[15]	ByteTrack[17]	CenterTrack[20]	FairMOT[18]	TraDeS[16]	GSDT[14]	TransTrack[12]
25	01	60.7%	56.1%	**61.6%**	34.6%	55.5%	45.3%	46.6%	38.8%
	02	58.3%	55.6%	**58.4%**	34.5%	52.1%	45.3%	40.6%	41.1%
	03	66.3%	65.7%	**67.7%**	37.8%	56.4%	40.3%	50.6%	43.9 %
	05	66.8%	64.4%	**68.3%**	24.4%	51.8%	31.4%	39.8%	43.2%
1	01	27.4%	27%	**34.8%**	31%	34.1%	32.4%	31.1%	30.7%
	02	24.1%	30.4%	**36.6%**	33%	31.4%	31.8%	27.7%	32.9%
	03	30.1%	38.3%	**47.2%**	35.4%	31.9%	31.4%	29.6%	39.3%
	05	32.3%	41.5%	**48.3%**	22%	35.5%	21%	27.5 %	35.6%

Table 2: Multi object tracking accuracy (MOTA) [11] of the different tracking methods on the MOT20 [2] training sequences (original and undersampled) using the ground truth detections instead of the internal detections.

Framerate	Sequence id	DeepSORT[15]	ByteTrack[17]	CenterTrack[20]	FairMOT[18]	GSDT[14]
25	01	95.1%	95%	**98.1%**	94.6%	94.5%
	02	96.1%	96.2%	**97.2%**	95.7%	95.6%
	03	96.5%	95.7%	**99.2%**	95.6%	95.4%
	05	**99.5%**	98.5%	99%	98.4%	97.3%
1	01	60.4%	54.3%	**70.3%**	60.9%	63.8%
	02	68.1%	63.1%	**72.8%**	58.9%	67.5%
	03	**76.8%**	60.1%	70.2%	51.2%	57.1%
	05	**74.8%**	67.2%	70.1%	60.7%	62.8%

Our quantitative comparison uses the Multi-Object Tracking Accuracy (MOTA) metric [11] which is defined as

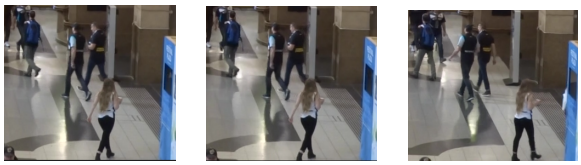
$$MOTA = 1 - \frac{\sum_t (FN_t + FP_t + IDSW_t)}{\sum_t GT_t}, \qquad (1)$$

where t is the frame index, GT is the number of ground truth objects, FN/FP the number of false negatives/positives respectively and $IDSW$ the number of identity switches. For a fair comparison, at a given sampling rate, the metric of the original sequence is also calculated only on the frames of the sub-sampled sequence.

The quality of the detection step plays a crucial role in the performance of tracking methods. A second experiment was added where the ground truth detections were used instead of the ones produced by each method. This allows us to evaluate the potential of the different association methods in the context of low framerate where significant appearance changes are observed.

4 Results and discussion

Figure 1 compares the performance of the selected methods on the MOT20-01 sequence for several framerates. As the framerate decreases, all the methods suffer a significant performance drop. Table 1 illustrates this in detail. ByteTrack and SORT achieve a good performance at high framerate. DeepSORT and FairMOT come right after, DeepSORT being better than the latter. The remaining methods have relatively close performance except CenterTrack which performs the worst, especially on the MOT20-05 sequence. This low performance is mostly due to the mediocre quality of the detections. At low framerate, ByteTrack retains its position while SORT suffers a drastic drop. CenterTrack on the other hand suffers the lowest relative drop. TransTrack is the second-best method on average at this framerate which might hint at transformers being better at handling motion. DeepSORT and FairMOT behave similarily on average with DeepSORT having an upperhand indicating that Re-ID features are helpful. The remaining methods are close, not far behind. At both framerates, FairMOT is the most consistent.



$t = 0$ $t = 1$ $t = 10$

Figure 2: Example of changes between frames in the MOT20-01 [2] sequence. In the original sequence, $t = 0$ and $t = 1$ would be two consecutive frames whereas, in the sub-sampled sequence (with a sampling rate of 0.1) $t = 0$ and $t = 10$ would be two consecutive frames.

When replacing internal detections with the ground truth ones, see Table 2, all the methods benefit from a major performance uplift at the original sequence framerate, especially CenterTrack. This illustrates that the detection performance is a major limitation to tracking methods. As we reduce the framerate to 1, although a significant performance drop is still observed, DeepSORT and CenterTrack obtain significantly better performance than the other trackers and than in Table 1. The performance of DeepSORT is significantly better than FairMOT, showing that joint re-id and detection learning still has its shortcomings as expected. We also notice that both motion prediction and re-identification seem to be promising, with each working better in certain scenarios, i.e. sequences with irregular movements are better handled by re-identification while sequences with heavy occlusions are better treated with motion estimation.

Finally, even if a better detection helps tracking performance, most current methods are still not effective enough to reliably track targets subject to significant changes between frames.

As a limitation of these experiments, one must note that re-identification methods are trained in a way that doesn't favor any framerate in particular, while motion trackers are conditioned on previous frames. CenterTrack has some sort of robustness due to being trained on randomly spaced frames but the performance hinges on the nature of the motion (the simpler the better). Moreover, the Re-ID branch in DeepSORT was trained using perfect detections which could translate to a bias of them working better on ground truth detections.

5 Conclusion

In this article, we compared the performance of several state-of-the-art object trackers under different framerates. We observed that all methods suffer from a drop in performance, as the framerate is reduced. Our results indicate that certain re-identification based methods as well as some motion estimation methods are the most robust to low framerate and to pronounced changes in appearance, but performance is still lackluster. This indicates that much work remains to be done to solve the problem of MOT for low-framerate videos.

Acknowledgments This work was performed using HPC resources from the "Mésocentre" computing center of CentraleSupélec and ENS Paris-Saclay supported by CNRS and Région Île-de-France (http://mesocentre.centralesupelec.fr/).

References

[1] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft. Simple online and realtime tracking. In *ICIP*. IEEE, 2016.

[2] P. Dendorfer, H. Rezatofighi, A. Milan, J. Shi, D. Cremers, I. Reid, S. Roth, K. Schindler, and L. Leal-Taixé. Mot20: A benchmark for multi object tracking in crowded scenes. *arXiv preprint arXiv:2003.09003*, 2020.

[3] Z. Ge, S. Liu, F. Wang, Z. Li, and J. Sun. Yolox: Exceeding yolo series in 2021. *arXiv preprint arXiv:2107.08430*, 2021.

[4] J. Hayashida and R. Bise. Cell tracking with deep learning for cell detection and motion estimation in low-framerate. In *MICCAI*. Springer, 2019.

[5] H. W. Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955.

[6] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia. Path aggregation network for instance segmentation. In *CVPR*, 2018.

[7] H. Luo, Y. Gu, X. Liao, S. Lai, and W. Jiang. Bag of tricks and a strong baseline for deep person re-identification. In *CVPR Workshops*, 2019.

[8] A. Milan, L. Leal-Taixé, I. Reid, S. Roth, and K. Schindler. Mot16: A benchmark for multi-object tracking. *arXiv preprint arXiv:1603.00831*, 2016.

[9] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *NeurIPS*, 28, 2015.

[10] S. Shao, Z. Zhao, B. Li, T. Xiao, G. Yu, X. Zhang, and J. Sun. Crowdhuman: A benchmark for detecting human in a crowd. *arXiv preprint arXiv:1805.00123*, 2018.

[11] R. Stiefelhagen, K. Bernardin, R. Bowers, J. Garofolo, D. Mostefa, and P. Soundararajan. The clear 2006 evaluation. In *International Evaluation Workshops CLEAR*. Springer, 2006.

[12] P. Sun, J. Cao, Y. Jiang, R. Zhang, E. Xie, Z. Yuan, C. Wang, and P. Luo. Transtrack: Multiple object tracking with transformer. *arXiv preprint arXiv:2012.15460*, 2020.

[13] C.-Y. Wang, H.-Y. M. Liao, Y.-H. Wu, P.-Y. Chen, J.-W. Hsieh, and I.-H. Yeh. Cspnet: A new backbone that can enhance learning capability of cnn. In *CVPR workshops*, 2020.

[14] Y. Wang, K. Kitani, and X. Weng. Joint object detection and multi-object tracking with graph neural networks. In *ICRA*. IEEE, 2021.

[15] N. Wojke, A. Bewley, and D. Paulus. Simple online and realtime tracking with a deep association metric. In *ICIP*. IEEE, 2017.

[16] J. Wu, J. Cao, L. Song, Y. Wang, M. Yang, and J. Yuan. Track to detect and segment: An online multi-object tracker. In *CVPR*, June 2021.

[17] Y. Zhang, P. Sun, Y. Jiang, D. Yu, Z. Yuan, P. Luo, W. Liu, and X. Wang. Bytetrack: Multi-object tracking by associating every detection box. *arXiv preprint arXiv:2110.06864*, 2021.

[18] Y. Zhang, C. Wang, X. Wang, W. Zeng, and W. Liu. Fairmot: On the fairness of detection and re-identification in multiple object tracking. *IJCV*, 129(11), 2021.

[19] L. Zheng, L. Shen, L. Tian, S. Wang, J. Wang, and Q. Tian. Scalable person re-identification: A benchmark. In *ICCV*, 2015.

[20] X. Zhou, V. Koltun, and P. Krähenbühl. Tracking objects as points. In *ECCV*. Springer, 2020.

[21] X. Zhou, D. Wang, and P. Krähenbühl. Objects as points. *arXiv preprint arXiv:1904.07850*, 2019.

[22] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*, 2020.