Supplementary Material for
“FRoG-MOT: Fast and Robust Generic Multiple-Object Tracking by IoU and Motion-State Associations”

Takuya Ogawa[1] Takashi Shibata[1] Hoshinori Hosoi[1]
[1]NEC Corporation

takuya_ogawa@nec.com t.shibata@ieee.org t.hosoi@nec.com

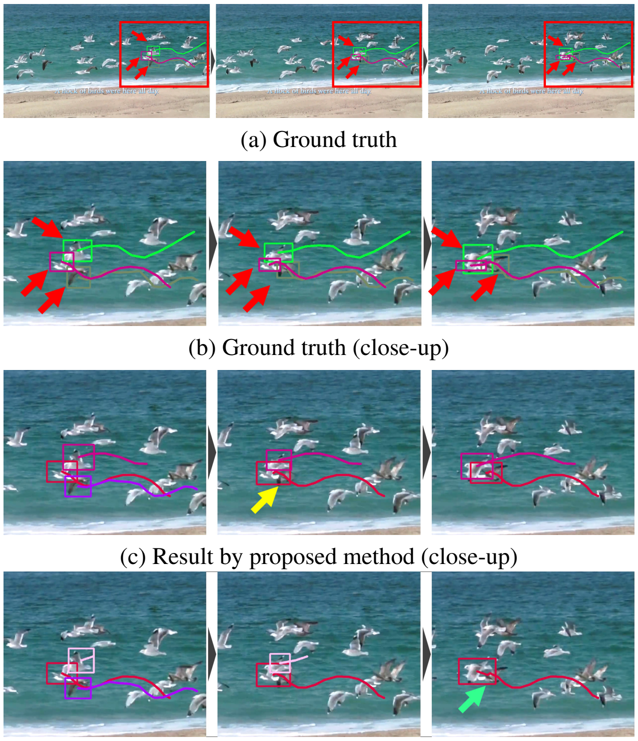
Abstract

This is the supplementary material for “FRoG-MOT: Fast and Robust Generic Multiple-Object Tracking by IoU and Motion-State Associations” (our main paper). We provide discussions of the proposed method’s limitations, additional experimental results, analysis of computational efficiency, and details of the datasets used in our experiments.

A. Additional Results

A.1. Limitations of our proposed method

In this section, we discuss the limitations of the proposed method. As described in Sec. 5.2 in the original paper, when detection performance is poor by target occlusion, this detection performance degrades tracking performance. Such object detection failures due to occlusion are particularly likely to occur when targets are very dense or for long periods of target tracking. Examples of specific sequences are shown in Figure 1. As shown in Fig. 1(a) and (b), three birds overlap in these sequences. (Please see the red arrows.) Note that for clarity, all but these three trajectories are not shown in Fig. 1. As shown in Fig. 1(c), the occluding of these three birds in the proposed method results in an undetected ID for one target. (Please see yellow arrows.) The proposed method fails to detect the object when there is excessive occlusion, and thus, the proposed method fails to track the occluded object. Note that the conventional method, ByteTrack [10], fails to track these two of the three targets, as shown in Fig. 1(d), even though it uses the same detector [3] as the proposed method. (Please see green arrows). These differences show the superiority of our proposed method over the existing method, even in such a challenging scenario. Note that the performance of the proposed method can be further enhanced by improving the performance of object detection, as described in Table 5 of our original paper. Improving the performance of object detection is essential for tracking-by-detection methods, including the proposed method.



(a) Ground truth

(b) Ground truth (close-up)

(c) Result by proposed method (close-up)

(d) Result by ByteTrack [10] (close-up)

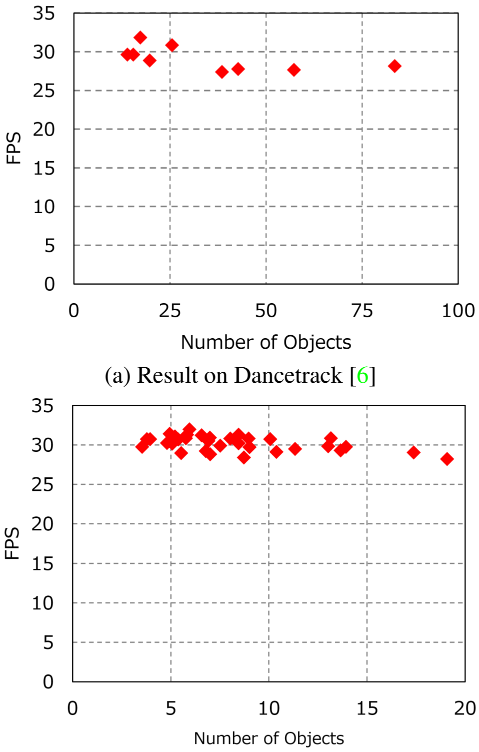
Figure 1. Limitations of our proposed method and ByteTrack [10]. Please zoom in for more details of trajectories for each box.

A.2. Analysis of Computational Efficiency

We provide a detailed analysis of the computational efficiency of our proposed method. Table 1 shows the relationship between processing speed and each data set’s average number of objects. This table shows that our proposed method is fast enough to allow real-time processing in all datasets (i.e., MOT17, Danstrack, and GMOT-40). The re-

Table 1. Relationship between the number of objects and FPS.

Dataset [6]	# Objects	Proc. Speed [FPS]
Dancetrack [6]	9	30.2
GMOT-Split101	35	29.5
MOT17 [5]	96	28.2



(a) Result on Dancetrack [6]

(b) Result on Results on GMOT-40 [1]

Figure 2. Relationship between the number of objects and processing speed for each sequence of Dancetrack and GMOT-40.

lationship between processing time and the average number of objects in each sequence of GMOT-40 [1][1] and Dancetrack [6] is shown in Fig. 2. As shown in Fig. 2, the increase in processing time with increasing average number of objects is slight. The reason for this is that the tracking process of the proposed method is a simple implementation; thus, the computation time is very small compared to the detection process in the previous stage. In fact, in the proposed method, the ratio of the computational time of the detection process (i.e., YOLOX [3]) to that of our tracking process is, on average, 79.5% and 20.5%, respectively. Those results suggest that the computational speed of our proposed method can be further improved by speeding up the detection process.

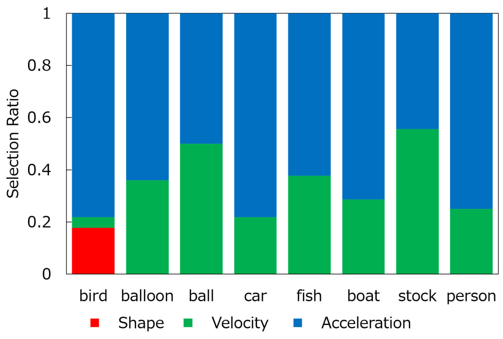


Figure 3. Selection ratio for each sequence on GMOT-Split101 [1]

A.3. Additional Analysis of Motion State Variable Selection

We discussed the effectiveness of motion state variable selection in Sec. 5.3 of our original paper. In this section, we further analyze the effectiveness of motion state variable selection in more detail. We evaluated the selection ratio of which motion state variable was used in each frame of GMOT-40 [1] and Dancetrack [6]. Figures 3 and 4 show the evaluation results for GMOT-40 [1] and Dancetrack [6]. In this analysis, we evaluated the selection ratio using the ground-truth detection results to remove the harmful effects of detection failures and analyze the properties of the proposed tracking algorithm. Figure 4 shows that acceleration was often used in all sequences, followed by velocity and shape in the Dancetrack dataset. These results are reasonable because the subject is a person, and the velocity and the shape rarely change rapidly during the dance. On the other hand, as shown in Figure 3, in GMOT-40, velocity and shape are used more frequently than in the dance track. In other words, velocity and shape are essential in addition to acceleration for general object tracking. In particular, the shape is an important motion state variable in sequences such as birds because their shape changes significantly due to the flapping of their wings [2].

Note that although appearance change was also interestingly considered as a candidate, we employ velocity, acceleration, and bounding box as motion state variables due to the poor features extracted from small objects and the heavy computation for real-time processing.

A.4. Additional Results on DanceTrack [6]

In Section 5.2 in our original paper, we analyzed the performance of our method using the DanceTrack dataset, a large-scale dataset of 2D MOT. Table 4 of our original paper shows that our proposed method is effective than Byte-

[1] Note that to evaluate the computational efficiency using large-scale data, we performed this computation time measurement using all sequences of GMOT-40.

[2] If the target is very slow (e.g., a helicopter taken at a long distance in the ”airplane” sequence), the motion state does not change rapidly. In this case, the selection of the motion state did not occur in the proposed method, so the selection ratio cannot be defined and is not shown in Fig. 3.

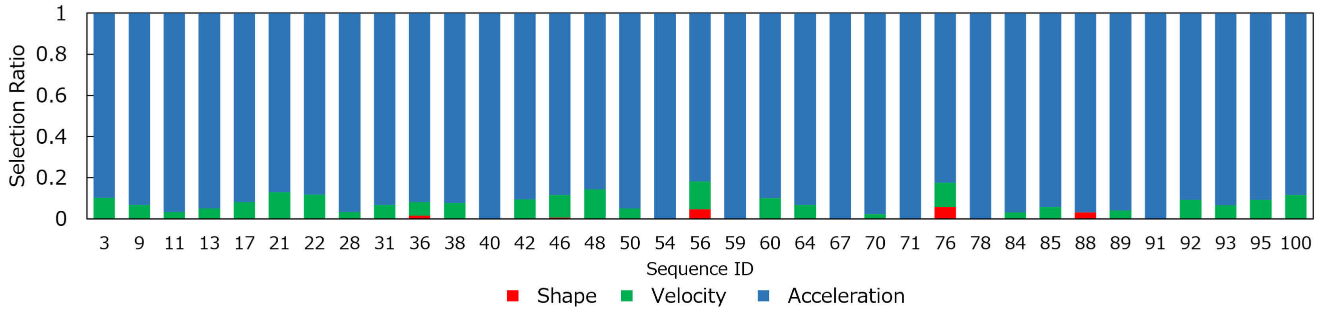


Figure 4. Selection ratio for each sequence on Dancetrack [6].

Table 2. Analysis using Dancetrack test-set [6]. Note that the results of ByteTrack† are taken from those published on Dancetrack's GitHub page. The results of ByteTrack†† are fair evaluation settings with our proposed method using the same detection algorithm and its parameters.

Methods	IDF1↑	HOTA↑	AssA↑	DetA↑	MOTA↑
CenterTrack [11]	35.7	41.8	22.6	78.1	86.8
ByteTrack† [10]	51.9	47.1	31.5	70.5	88.2
ByteTrack†† [10]	48.9	44.6	28.5	70.1	87.7
Ours	53.2	46.8	31.3	70.3	88.2

Track [10] in terms of IDF1 on this data set. In this supplemental, the evaluation data in the test set is also shown in Table 2. Note that the ByteTrack† results are taken from the GitHub page of the DanceTrack Project Page [3]. Furthermore, the results of ByteTrack†† are a fair evaluation setting with our proposed method using the same detection algorithm and its hyper-parameters. Table 2 also shows that the performance of our method on the DanceTrack test set is similar to or superior to that of ByteTrack in terms of IDF1 [4]. Those results suggest the high versatility of our proposed method.

Note that this paper focuses primarily on 2D generic MOT. Various datasets and evaluations [2, 7–9] for 3D object tracking have also been published recently. Studying the extending our approach to three-dimensional generic MOT is a future work.

A.5. Additional Visual Comparisons

Figure 5 shows the additional tracking examples (i.e., the trajectory of each target) of the ground truth, ByteTrack, and our proposed method in the last frame of GMOT-Split101, respectively. Comparing the results of the ground truth, ByteTrack, and the proposed method shown in the first column of this figure (i.e., airplane), it can be seen that ByteTrack and the proposed method can accurately track each target. However, as described in Sec. A.7, ByteTrack fails to track the generic object in bird and fish sequences, similar to other existing methods focusing on tracking person crowds in MOT benchmarks. The visual comparisons of those bird and fish sequences shown in the second and third columns show that the trajectories are inaccurate and often missing in ByteTrack (see the white dotted boxes). In contrast, the proposed method can robustly track those objects using motion state associations. As shown in the fourth and last columns, our proposed method is robust in tracking many vehicles and balls (see the green dotted boxes). Those results demonstrate the high versatility of the proposed method for generic MOT tasks. We have submitted the supplementary video showing the additional tracking results of our proposed method on GMOT-Split101.

A.6. Additional Results on MOT17 [5]

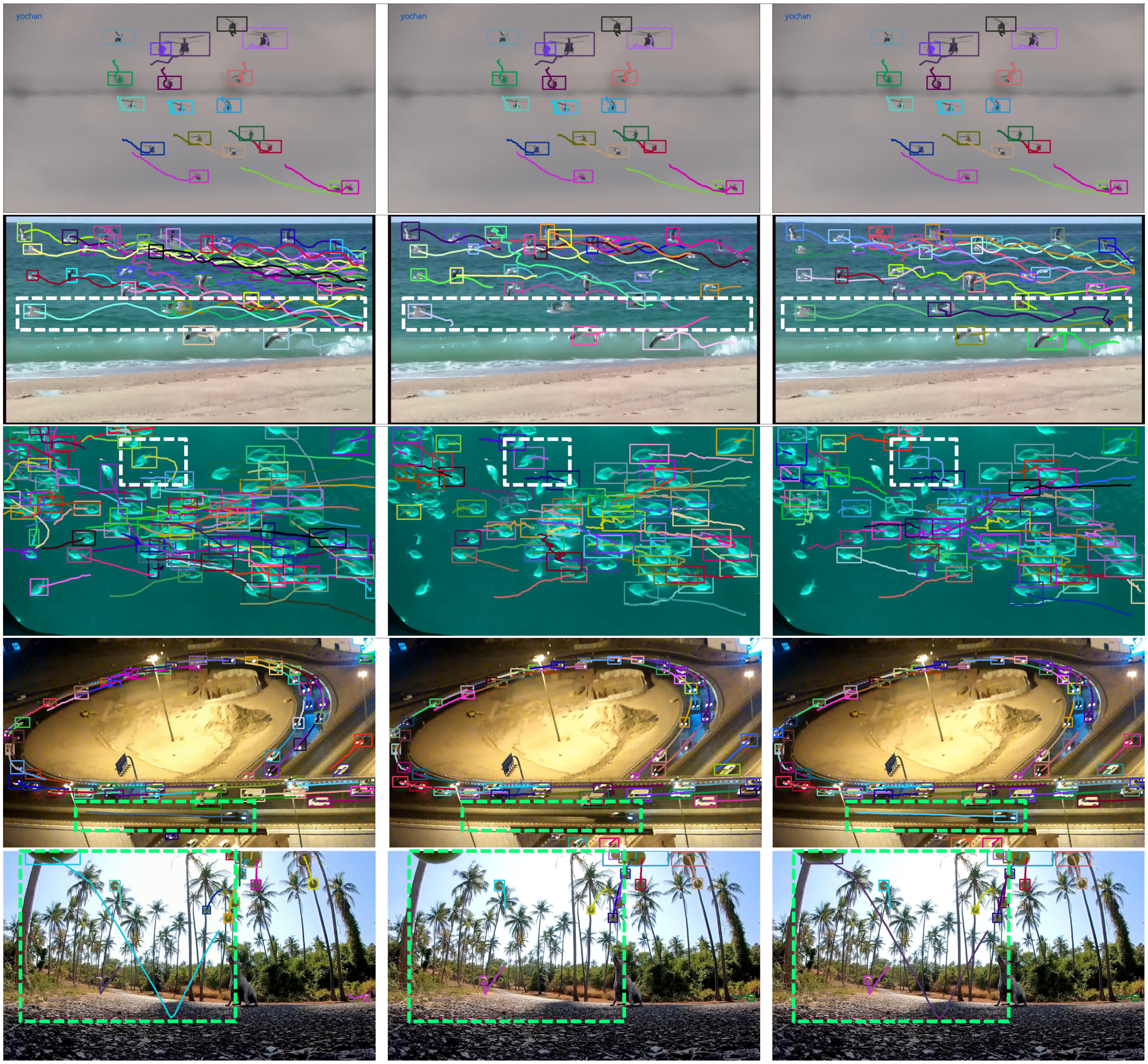
In our original paper, we present the overall results on the MOT17-val-half dataset in Table 3 of our original paper. We show additional results for each dataset sequence in Tab. 3. Here, the best/second results among the proposed and existing methods are shown in bold/underlined, respectively. In the MOT17-02 sequence, it can be observed that our proposed method significantly improved with an IDF1 score of 80.1, while the other methods had scores below 50. In particular, the proposed method significantly improved over the other methods for MOT17-05, MOT17-09, and MOT17-11 sequences. These sequences are challenging for target tracking due to low camera angles, increased occlusions, and multiple crossing trajectories. Considering each target's motion, the proposed method can handles occlusions and crossings, leading to improved tracking continuity in these challenging sequences.

A.7. Additional Results on GMOT-Split101

Our original paper presents the overall results on the GMOT-Split101 dataset in Table 2 of our original paper. We show additional results for each dataset sequence in Tab. 4. Here, the best/second results among the proposed and existing methods are shown in bold/underlined, respectively.

[3] https://github.com/DanceTrack/DanceTrack

[4] Please note that the hyper-parameters of our method in this analysis are the same as those of the experiment of our original paper.



(a) Ground truth (b) ByteTrack [10] (c) Ours

Figure 5. Additional visual comparisons on GMOT-Split101. Please zoom in for more details of trajectories for each box.

When examining the results for each sequence, the proposed method consistently achieves high scores for all ID metrics (IDF1, IDP, and IDR), demonstrating strong tracking continuity across all target types. For challenging sequences such as bird and fish datasets, the performances of all existing methods, including ByteTrack [10] are poor because it is difficult to track those targets continually due to the sudden motion changes. In contrast, our proposed method outperforms ByteTrack [10], which only uses IoU for the association, by 4.0 points in the bird sequence and 5.0 points in the fish sequence, respectively. This result suggests that the proposed method can effectively handle target changes using target-specific motion states. The design of the proposed method focuses on those challenging scenes where general objects are densely arranged, and the motion of each object changes abruptly. On the other hand, for sequences such as airplane and balloon, where the appearance changes little, motion is mostly linear, and the target is sparsely located, the performance of Trackformer [4] is comparable or better. Note that as described in our orig-

Table 3. MOT17-val-half benchmark results in each sequence.The best/Second results are shown in **bold**/underline.

(a) CenterTrack [11]

	IDF1↑	IDP↑	IDR↑	Rec.↑	Prec.↑	MT↑	IDS↓	FM↓	MOTA↑	MOTP↑
MOT17-02	36.7	57.2	27.1	42.0	88.8	10	100	102	35.6	17.8
MOT17-04	77.6	83.0	72.9	86.3	98.3	50	143	158	84.3	17.3
MOT17-05	60.3	76.6	49.7	62.8	96.7	21	52	49	59.0	17.4
MOT17-09	60.7	73.4	51.8	70.1	**99.3**	13	40	34	68.2	16.6
MOT17-10	54.7	61.5	49.3	68.3	85.3	13	95	141	54.9	**23.3**
MOT17-11	57.4	68.0	49.6	67.1	91.8	11	28	30	60.5	12.8
MOT17-13	61.2	64.0	58.7	75.5	82.5	22	70	74	57.2	**22.3**
OVERALL	64.2	74.2	56.6	71.6	94.1	41	528	588	66.1	**17.9**

(b) Trackformer [4]

	IDF1↑	IDP↑	IDR↑	Rec.↑	Prec.↑	MT↑	IDS↓	FM↓	MOTA↑	MOTP↑
MOT17-02	39.8	65.2	28.6	42.4	96.6	10	80	89	40.1	15.4
MOT17-04	82.6	**90.6**	76.0	83.1	**99.1**	46	67	**58**	82.1	12.9
MOT17-05	69.1	80.4	60.6	73.2	**97.1**	29	43	47	69.7	17.7
MOT17-09	65.6	78.7	56.3	69.7	97.5	13	17	**15**	67.4	11.4
MOT17-10	**68.6**	**75.5**	62.9	78.0	**93.6**	17	**38**	**104**	72.0	20.5
MOT17-11	67.1	77.6	59.1	75.0	**98.4**	19	19	35	**73.3**	11.6
MOT17-13	**81.6**	**86.5**	**77.3**	**85.6**	**95.9**	**31**	81	69	**79.4**	19.1
OVERALL	71.5	83.4	62.5	73.2	**97.7**	49	345	**417**	70.8	14.6

(c) ByteTrack [10]

	IDF1↑	IDP↑	IDR↑	Rec.↑	Prec.↑	MT↑	IDS↓	FM↓	MOTA↑	MOTP↑
MOT17-02	48.8	58.1	42.1	61.6	85.0	15	85	181	49.8	**20.0**
MOT17-04	**89.7**	89.0	**90.5**	**94.2**	92.8	**60**	29	79	**86.8**	14.9
MOT17-05	73.4	81.9	66.5	76.8	94.7	34	**17**	**36**	72.0	**18.0**
MOT17-09	75.7	82.8	69.7	**82.8**	98.4	17	**10**	30	**81.1**	17.7
MOT17-10	67.9	73.3	**63.2**	**78.6**	91.2	16	**38**	110	70.4	22.7
MOT17-11	77.4	83.3	72.2	79.4	91.6	23	**11**	27	72.0	15.0
MOT17-13	73.8	78.9	69.4	79.1	89.9	25	16	42	69.7	22.0
OVERALL	77.0	81.1	73.2	82.7	91.6	56	206	505	74.7	17.1

(d) Ours

	IDF1↑	IDP↑	IDR↑	Rec.↑	Prec.↑	MT↑	IDS↓	FM↓	MOTA↑	MOTP↑
MOT17-02	**80.1**	**87.6**	**73.7**	**83.8**	**99.6**	**16**	**13**	**24**	**83.0**	15.5
MOT17-04	68.6	76.9	62.0	75.6	93.7	16	**26**	87	70.0	**21.9**
MOT17-05	**90.8**	**91.3**	**90.2**	**93.5**	94.6	**61**	18	91	**88.1**	12.8
MOT17-09	**78.2**	**84.4**	**72.8**	81.9	95.0	**37**	25	48	76.8	**18.0**
MOT17-10	58.0	67.9	50.5	64.5	86.7	17	62	179	53.9	19.6
MOT17-11	**83.8**	**92.8**	**76.4**	**80.1**	97.3	**29**	14	**20**	77.5	**19.7**
MOT17-13	67.5	70.1	65.0	81.4	87.8	22	**12**	**28**	69.8	12.8
OVERALL	**79.2**	**83.9**	**75.0**	**83.2**	93.1	**58**	**170**	477	**76.7**	15.5

inal paper, the processing time of our proposed method overwhelmingly outperforms those transformer-based approaches.

B. Details of GMOT-Split101 Dataset

We provide a detailed explanation of the GMOT-Split101 dataset used in the main paper. GMOT-Split101 was prepared based on the GMOT-40 dataset. While GMOT-40 is a high-quality dataset containing various types of objects in crowded scenes, using it for MOT evaluation was challenging due to the lack of separation into training and test sets for each class. To address these issues, we separated the data into training and test sets for each class based on the four sequences available per class, as shown in Tab. 5. For example, in the proposed method and ByteTrack [10], the training datasets are used to train the parameters of YOLOX [3]. Note that the training protocol for YOLOX followed the original paper [3] and the public implementation.

Our GMOT-Split101 dataset is organized from the existing dataset GMOT-40 to improve two points: data imbalance and domain shift. To prevent data imbalance due to longer test sequences, we standardized the sequence length by adding 100 tracking frames to the initial frame, resulting in 101 frames. For classes with limited training data, ad-

Table 4. Evaluation results of each method for each sequence in the GMOT-Split101 benchmark.

(a) CenterTrack [11]

Sequence	IDF1↑	IDP↑	IDR↑	Rec.↑	Prec.↑	MOTA↑	MOTP↑
airplane	98.8	98.7	**98.9**	**98.9**	98.7	97.6	0.20
ball	73.8	75.7	72.0	92.1	96.8	87.5	0.10
balloon	83.7	82.3	85.1	86.6	83.7	69.6	0.14
bird	41.7	55.5	33.4	54.2	**90.0**	39.3	0.24
boat	87.5	89.5	85.7	90.3	94.3	84.5	0.18
car	87.8	87.6	87.9	90.9	90.6	81.3	0.17
fish	52.3	59.0	46.9	59.1	74.3	37.1	**0.32**
person	88.2	89.0	87.5	98.2	**99.9**	97.7	0.13
stock	87.3	91.7	83.4	89.3	**98.2**	87.1	0.17
OVERALL	72.8	78.0	68.3	77.6	88.6	65.8	0.21

(b) Trackformer [4]

Sequence	IDF1↑	IDP↑	IDR↑	Rec.↑	Prec.↑	MOTA↑	MOTP↑
airplane	**99.0**	**99.1**	**98.9**	**98.9**	**99.1**	**98.0**	**0.22**
ball	83.5	85.0	82.1	**95.6**	**98.9**	**93.5**	**0.15**
balloon	**92.4**	**91.0**	**93.9**	**95.0**	**92.0**	**86.6**	**0.20**
bird	58.0	63.7	53.3	73.9	88.2	**59.8**	0.24
boat	86.0	91.2	81.4	86.7	**97.2**	83.9	**0.20**
car	90.4	90.5	90.3	92.1	92.3	84.4	**0.21**
fish	57.4	**69.7**	48.8	57.1	**81.3**	**49.8**	0.25
person	83.6	84.8	82.4	96.4	99.3	95.0	**0.17**
stock	91.2	94.0	88.6	92.0	97.5	89.3	**0.20**
OVERALL	78.5	**84.0**	73.7	81.1	**92.5**	**73.6**	**0.21**

(c) ByteTrack [10]

Sequence	IDF1↑	IDP↑	IDR↑	Rec.↑	Prec.↑	MOTA↑	MOTP↑
airplane	96.2	96.2	96.2	96.2	96.2	92.4	0.20
ball	92.7	93.6	91.9	93.0	94.8	87.7	0.13
balloon	86.9	81.2	93.5	93.5	81.2	72.0	0.16
bird	59.8	64.6	55.6	72.1	83.8	53.0	0.31
boat	93.5	93.9	93.0	**94.4**	95.3	89.6	0.14
car	93.3	91.6	**94.9**	**95.8**	92.4	87.8	0.17
fish	55.1	61.1	50.2	**65.6**	79.8	46.6	0.26
person	98.7	98.4	99.0	**99.5**	99.0	98.3	0.12
stock	92.6	93.0	92.1	**95.7**	96.7	**92.3**	0.16
OVERALL	78.8	81.3	76.5	83.8	89.2	72.3	0.20

(d) Ours

Sequence	IDF1↑	IDP↑	IDR↑	Rec.↑	Prec.↑	MOTA↑	MOTP↑
airplane	96.2	96.2	96.2	96.2	96.2	92.4	0.20
ball	**92.8**	**93.7**	**91.9**	93.0	94.8	87.8	0.13
balloon	88.4	83.7	93.6	93.6	83.7	75.4	0.16
bird	**63.8**	**66.7**	**61.1**	**74.7**	83.2	54.7	**0.31**
boat	**94.7**	**95.8**	**93.7**	94.0	96.1	**90.0**	0.14
car	**93.4**	**91.9**	94.7	95.6	**92.8**	**88.1**	0.17
fish	**60.1**	67.4	**54.2**	65.2	81.2	48.0	0.26
person	**99.3**	**99.4**	**99.1**	99.1	99.4	**98.5**	0.12
stock	**93.5**	**94.1**	**92.9**	95.5	96.8	92.2	0.16
OVERALL	**80.8**	83.7	**78.2**	**83.9**	89.8	73.2	0.20

Table 5. Composition of the sequence elected by GMOT-Split101

Class	For train	For test	Non-use
airplane	1,2	2(rest)	3,4
ball	2,3,4	3(rest)	1
balloon	2,4	4(rest)	1,3
bird	1,2	2(rest)	3,4
boat	1,2	4	3
car	1,2,4	1(rest)	3
fish	2,3,4	1	-
person	2,3	3(rest)	1,4
stock	1,3	1(rest)	2,4

ditional frames from the test set were included in the training set (indicated by ”rest” in Tab. 5, it does not use for a test). Some classes in the dataset are composed entirely of sequences with similar domains, where scenes within each class are similar. For some other classes, all sequences have vastly different domains, including different object types, making it difficult to evaluate the tracking performance fairly. We only used sequences with similar or related domains for both the training and test sets to prevent evaluation biases resulting from significant domain shifts within each class. Other sequences with significant domain shifts were excluded from the original dataset (as indicated by ”Non-use” in Tab. 5). For instance, we included only the sequences containing flying birds for the bird class because walking birds sequence had vastly different domains.

References

[1] Hexin Bai, Wensheng Cheng, Peng Chu, Juehuan Liu, Kai Zhang, and Haibin Ling. Gmot-40: A benchmark for generic multiple object tracking. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, pages 6719–6728, 2021. 2

[2] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *IEEE Conf. Comput. Vis. Pattern Recog.(CVPR)*, pages 11621–11631, 2020. 3

[3] Zheng Ge, Songtao Liu, Feng Wang, Zeming Li, and Jian Sun. Yolox: Exceeding yolo series in 2021. *arXiv preprint arXiv:2107.08430*, 2021. 1, 2, 5

[4] Tim Meinhardt, Alexander Kirillov, Laura Leal-Taixe, and Christoph Feichtenhofer. Trackformer: Multi-object tracking with transformers. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, pages 8844–8854, 2022. 4, 5, 6

[5] A. Milan, L. Leal-Taixé, I. Reid, S. Roth, and K. Schindler. MOT16: A benchmark for multi-object tracking. *arXiv:1603.00831 [cs]*, Mar. 2016. arXiv: 1603.00831. 2, 3

[6] Peize Sun, Jinkun Cao, Yi Jiang, Zehuan Yuan, Song Bai, Kris Kitani, and Ping Luo. Dancetrack: Multi-object tracking in uniform appearance and diverse motion. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2022. 2, 3

[7] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, et al. Scalability in perception for autonomous driving: Waymo open dataset. In *IEEE*

Conf. Comput. Vis. Pattern Recog.(CVPR), pages 2446–2454, 2020. 3

[8] Benjamin Wilson, William Qi, Tanmay Agarwal, John Lambert, Jagjeet Singh, Siddhesh Khandelwal, Bowen Pan, Ratnesh Kumar, Andrew Hartnett, Jhony Kaesemodel Pontes, et al. Argoverse 2: Next generation datasets for self-driving perception and forecasting. In *Adv. Neural Inform. Process. Syst. Datasets and Benchmarks Track*, 2021. 3

[9] Hai Wu, Wenkai Han, Chenglu Wen, Xin Li, and Cheng Wang. 3d multi-object tracking in point clouds based on prediction confidence-guided data association. *IEEE Trans. on Intell. Transportation Syst.*, 23(6):5668–5677, 2021. 3

[10] Yifu Zhang, Peize Sun, Yi Jiang, Dongdong Yu, Fucheng Weng, Zehuan Yuan, Ping Luo, Wenyu Liu, and Xinggang Wang. Bytetrack: Multi-object tracking by associating every detection box. In *Eur. Conf. Comput. Vis. (ECCV)*, pages 1–21. Springer, 2022. 1, 3, 4, 5, 6

[11] Xingyi Zhou, Vladlen Koltun, and Philipp Krähenbühl. Tracking objects as points. In *Eur. Conf. Comput. Vis. (ECCV)*, pages 474–490. Springer, 2020. 3, 5, 6