A Cascade Matching Algorithm for Multi-object Tracking Systems Based on Detection Box Scores

Hongshen Zhao[1] and Junyong Zhai[2](✉)

[1] School of Software, Southeast University, Suzhou 215123, China
[2] School of Automation, Southeast University, Nanjing 210096, China
jyzhai@seu.edu.cn

Abstract. Tracking by detection has become the mainstream paradigm of current multi-object tracking. A threshold is set to filter out low-score detection boxes in order to reduce the impact of background false detections. However, these low-score detection boxes may contain supplementary information useful for similarity computation and target matching. This paper proposes a cascade matching algorithm based on detection box scores, which makes full use of the position information, motion information and appearance information of detection boxes. Experimental results show that the proposed matching algorithm can effectively maintain identities and achieve continuous tracking in the case of ambient light interference, motion blur and partial occlusion.

Keywords: Multi-object tracking · Information aggregation · Target matching

1 Introduction

Multi-object tracking (MOT) is a typical task in computer vision, which aims to predict the location and identity (ID) of each target in the video stream data. There are two key processes in multi-object tracking: object detection and data association. Most of the current best-performing methods adopt the tracking by detection paradigm, which first detects the locations of objects in the current frame by an object detector, then calculates the similarity between detections and targets, and finally matches detections with targets by a matching algorithm to assign IDs to each tracked target. The simple online and realtime tracking (SORT) algorithm [2] was one of the first MOT pipelines to predict the current frame trajectory position using the Kalman filter [6]. The cost matrix is calculated from the Intersection over Union (IoU) information between predictions and detections, and finally the matching is completed by the Hungarian algorithm [7]. Bytetrack [14] divides the matching process into two stages according to the scores of detection boxes. Relying on the high-accuracy detections provided by YOLOX [3], Bytetrack has reached the state-of-the-art level on multiple

Y. Jia et al. (Eds.): CISC 2022, LNEE 950, pp. 854–862, 2022.
https://doi.org/10.1007/978-981-19-6203-5_85

benchmark datasets by only utilizing the IoU information between detections and predictions. The above methods are simple in structure but highly dependent on the performance of the detector. When the target disappears completely and reappears due to occlusion, the trajectory of the target cannot be reconstructed. Some recent works [8,13] propose to crop detected image regions and feed them to re-identification (ReID) networks after resizing to extract appearance features. These ReID features are used for similarity computation between detections and trajectories. This method is effective for trajectory reconstruction because the locations of objects may change greatly after being occluded for a long time, but their appearance information is similar. It is worth noting that the apparent features are usually unreliable when the score of the detection box is low, i.e. there is severe occlusion or motion blur.

In this paper, data association is the focus of a multi-object tracking system. Firstly, the influence of position information, motion information and appearance information on the MOT system is explored through information aggregation. Then, a cascade matching algorithm is proposed based on detection box scores. Finally, experimental results show that the proposed algorithm can improve the performance of the MOT system from several aspects.

2 Methodology

2.1 Information Aggregation

Estimation Model. The displacement of each object between frames is approximately a linear uniform motion, and the noise obeys a Gaussian distribution. Each object is defined in the 8-dimensional state space as follows:

$$X = (u, v, a, h, \dot{u}, \dot{v}, \dot{a}, \dot{h})^T \tag{1}$$

where the state space contains the horizontal and vertical coordinates of the object centroid (u, v), aspect ratio a, height h, and their respective velocities. The detection bounding box location (u, v, a, h) is taken as the direct observation of the state space. During data association, detection boxes are used to update the state of each target, where the velocity components are solved optimally via a Kalman filter framework [6].

Position Information. A bounding box is used to represent the position information of each target. Let B_d and B_p denote detection boxes obtained from the object detector and prediction boxes obtained from the Kalman filter, respectively. Then the position distance matrix can be calculated as the IoU distance between each detection box and all prediction boxes of the existing targets:

$$D_{pos} = 1 - \frac{B_d \cap B_p}{B_d \cup B_p} \tag{2}$$

In practice, if the position distance between the detection box and prediction box is greater than the given hyper parameter D_{IoU}, the match is considered failed.

Motion Information. The motion information of each target is modeled using Mahalanobis distance. Mahalanobis distance is an effective method to calculate the similarity between two unknown sample sets. It is used to quantify the matching degree between predictions and detections. In practice, the squared Mahalanobis distance is generally used:

$$D_m = \left(X_d - X_p\right)^T S^{-1} \left(X_d - X_p\right) \tag{3}$$

where X_d and X_p represent the detection distribution and prediction distribution, respectively. S is the covariance matrix between the two distributions.

Appearance Information. Appearance feature descriptors r_d and r_t denote respectively the appearance information of detections and existing trajectories obtained by the feature extraction network. OSNet [15] is used for feature extraction, and the obtained feature descriptors are all 512-dimensional unit feature vectors. The characteristic cosine distance can be calculated as:

$$D_r = \min_{0<i\leq N} \left\{ 1 - r_d^T r_t^{(i)} \middle| r_t^{(i)} \in \mathcal{R}_i \right\} \tag{4}$$

where $\mathcal{R}_i$ is the appearance descriptor gallery of the i-th trajectory and N is the total number of trajectories. In practice, the match is considered failed if the cosine distance between the detection box and the trajectory is greater than the given hyper parameter D_{app}.

Combine these three kinds of information to obtain the hybrid distance matrix:

$$D = \alpha D_{pos} + \beta D_m + \left(1 - \alpha - \beta\right) D_r \tag{5}$$

where $0 \leq \alpha \leq 1$ and $0 \leq \beta \leq 1$ are hyper parameters satisfying $0 \leq \alpha + \beta \leq 1$.

2.2 A Cascade Matching Algorithm Based on Detection Box Scores

Most research works set a threshold (commonly 0.5) to prevent false detection of the background, resulting in error filtering of low-score foreground objects. To make full use of each detection box and reduce missed detections and trajectory fragments, a cascade matching algorithm is proposed based on detection box scores. The process is shown in Fig. 1, and the specific steps are as follows:

Step 1: Split detection set into D_{high} and D_{low} according to their scores. In specific, two detection score thresholds ξ_{high} and ξ_{low} are set. Detection boxes with scores above ξ_{high} are put in D_{high}, and those with scores between ξ_{high} and ξ_{low} are put in D_{low}.

Step 2: Split track set into confirmed and unconfirmed tracks. When a new track is created, it is in the initialization stage, i.e. an unconfirmed track, and is transformed into a confirmed track only when it successfully matches N_INIT times in a row.

Step 3: Perform the first matching using confirmed tracks and D_{high}. It is full cost matching, i.e., the hybrid distance described in Sect. 2.1 is used as the

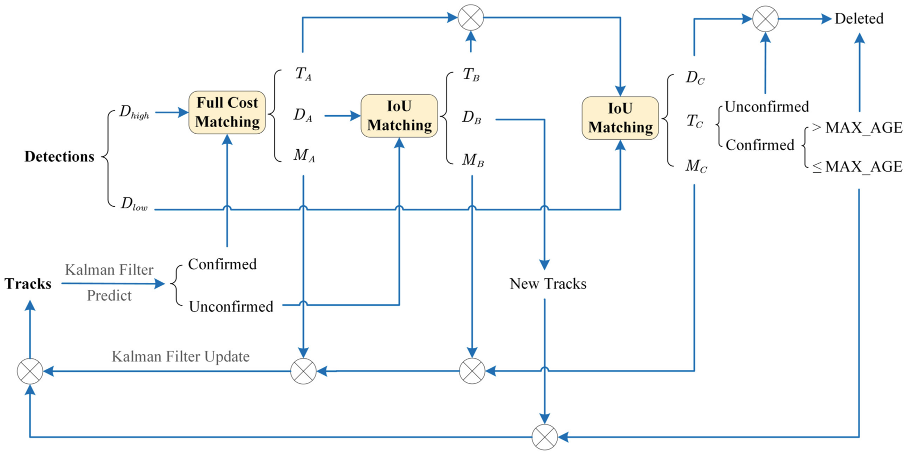


Fig. 1. A cascade matching algorithm based on detection box scores.

input of the Hungarian algorithm [7] to solve a linear sum assignment problem. There are three outputs: M_A, T_A and D_A, representing the set of successfully matched tracks, the set of unmatched tracks, and the set of unmatched detections, respectively.

Step 4: Perform IoU matching using unconfirmed tracks and the remaining high-score detections in the previous matching, i.e. D_A. The unconfirmed tracks are likely to be interference generated by false detections, and their appearance information and motion information may change drastically. Thus in this matching, only the position information of the detections and predictions is used. Specifically, when the position distance D_{pos} between detections and predictions is greater than D_{IoU}, the match is rejected. There are also three outputs: M_B, T_B and D_B.

Step 5: Rematch the remaining tracks T_A and T_B with D_{low} by IoU matching. Low-score detection boxes may contain background false positives, so the position distance threshold is lowered to $D_{IoU} - \varepsilon$. The implication is that for a low-score detection, the matching will be accepted only if the position distance between the detection and the prediction is close enough, which reduces mismatches to some extent. The outputs include M_C, T_C and D_C.

Step 6: After the above matching process, all the successfully matched tracks are obtained by combining M_A, M_B and M_C. The unmatched tracks T_C are directly deleted if they are unconfirmed. Otherwise, they are deleted only after having been unmatched continuously for MAX_AGE times. Note that for unmatched detections, only the high-score D_B is used to initialize new tracks, while D_C is directly deleted to ensure that the newly established tracks have reliable appearance information.

3 Experiments

3.1 Setting

Dataset. The MOT16 [9] dataset is adopted to verify the information aggregation method and the matching algorithm. This benchmark contains 14 challenging video sequences filmed with both static and moving cameras in unconstrained environments. Note that the MOT16 evaluation is performed on the train split using official evaluation code [5] as the test ground truth is not publicly available. However, this is not an issue as the train split is never used for training. In addition, CrowdHuman [11] and MSMT17 [12] are used as object detection dataset and person re-identification dataset, respectively.

Metrics. IDF1 [10] and CLEAR [1] metrics, including MOTA, IDs, FP, FN, and Frag, are used to evaluate the performance of the proposed methods. The description of each metric is shown in Table 1.

Table 1. Evaluation metrics used in this paper.

Metric	Description
IDF1	The ratio of correctly identified detections over the average number of ground-truth and computed detections. It is mainly concerned with the ability to maintain trajectories
MOTA	Multi-object tracking accuracy. It intuitively shows the overall performance in detecting objects and preserving trajectories
IDs	The total number of Identity Switches
FP	The total number of False Positives, i.e. false detections
FN	The total number of False Negatives, i.e. missed detections
Frag	The total number of times a trajectory is interrupted during tracking

Implementation Details. In the calculation of hybrid distance matrix, the default values of D_{app} and D_{IoU} are 0.2 and 0.7, respectively. The input image size is 736 × 1280. The default N_INIT, MAX_AGE, ξ_{high} and ξ_{low} are set to 3, 30, 0.7 and 0.2, respectively. The ReID network for feature extraction is OSNet [15] with osnet_x0_25 as the backbone and pre-trained on the MSMT17 dataset. The detector is YOLOv5 [4] with YOLOv5m as the backbone and the COCO-pretrained model as the initialized weights. The model is trained on an NVIDIA Tesla V100 GPU on the CrowdHuman dataset for 300 epochs with batch size of 64. The optimizer is SGD with weight decay of 0.0005 and momentum of 0.937. The initial learning rate is 0.01 with 3 epochs warm-up.

3.2 Ablation Studies on Information Aggregation

DeepSORT [13] is used as the baseline to study the influence of position, motion and appearance information on the MOT system. As shown in Fig. 2, when $\alpha = 0$

or $\beta = 0$, especially $\alpha + \beta = 1$, the MOTA value decreases significantly. In other words, when calculating the similarity between detections and existing tracks, compared with only considering the position information, motion information or appearance information, the method of information aggregation is beneficial to improving the accuracy of the tracking system. Table 2 shows the results of using one of position information, motion information and appearance information and performing information aggregation respectively when calculating the distance matrix. Compared with position and motion information, appearance information has advantages in improving the accuracy of the tracking system, especially in reducing the number of ID switches. This is due to its consistency across frames and the ability to reconstruct trajectories of occluded targets. In addition, after aggregating the three kinds of information, the overall performance of the tracking system is further improved, indicating the importance of information aggregation.

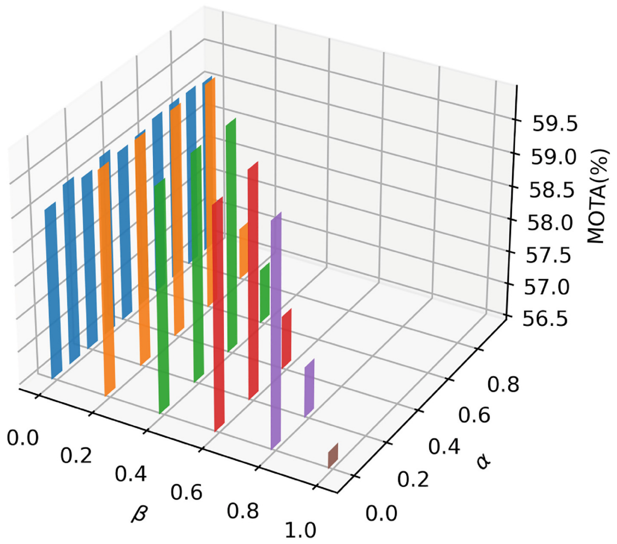


Fig. 2. The influence of information aggregation coefficients α and β.

Table 2. Comparison of results on MOT16 dataset when calculating distance matrix with different information. ↑ means higher is better while ↓ means lower is better. The best results are shown in **bold**.

Information	α	β	MOTA↑	IDF1↑	IDs↓	Frag↓
Position	1	0	45.412	40.404	2091	3251
Motion	0	1	56.755	52.273	1169	3107
Appearance	0	0	59.061	61.898	531	2357
Aggregation	0.4	0.5	**59.932**	**63.415**	**452**	**2261**

3.3 Ablation Studies on Matching Algorithm

The hybrid distance matrix is adopted as the input of the first matching, i.e., full cost matching. In specific, set $\alpha = 0.4$ and $\beta = 0.5$. For a more intuitive comparison, the improved DeepSORT algorithm is used as the strong baseline, which also takes the hybrid distance matrix as the similarity measurement. The results under different thresholds are shown in Table 3. Detections with higher scores have more reliable position, motion and appearance information, thus as ξ_{high} decreases, the number of detections in D_{high} increases, leading to an increase in TPs. Low-score detections can still provide relatively reliable position information when targets are partially occluded. Therefore, as ξ_{low} decreases, the number of retained detections increases, leading to the decrease of tracking interruptions, i.e. Frags. However, low-score detections may contain background false detections, causing an increase in FPs. Overall, the proposed matching algorithm based on detection box scores consistently outperforms the strong baseline on major metrics such as MOTA and IDF1.

Table 3. Comparison of results on the MOT16 dataset under different thresholds.

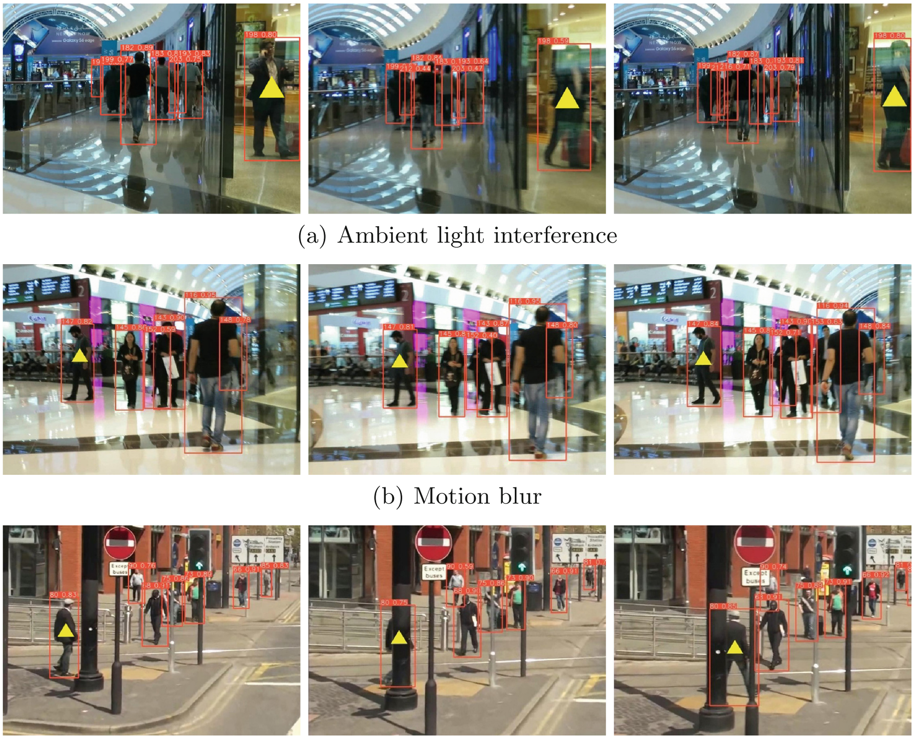
Method	ξ_{high}	ξ_{low}	MOTA↑	IDF1↑	IDs↓	Frag↓	TP↑	FP↓
DeepSORT [13]	–	–	59.871	63.387	466	2283	73968	7400
Baseline	–	–	59.932	63.415	452	2261	73998	7377
Ours	0.6	0.1	61.354	67.24	549	1442	78732	10444
	0.6	0.2	61.979	68.067	524	1507	77776	8823
	0.6	0.3	61.547	66.31	539	1601	76587	8096
	0.7	0.1	61.159	69.039	404	1175	75814	7886
	0.7	0.2	61.428	68.822	413	1238	75065	6831
	0.7	0.3	61.05	67.049	418	1269	74120	6299

Table 4. Evaluation of three proposed modules in the matching process.

Process	MOTA↑	IDF1↑	IDs↓	Frag↓
Original	61.428	68.822	413	1238
w/o full-cost matching	61.575 (+0.147)	65.747 (−3.075)	443	1165
w/o new tracks filtering	59.044 (−2.384)	64.085 (−4.737)	1012	2027
w/o D_{low} rematching	57.231 (−4.197)	62.798 (−6.024)	357	1338

Three important modules in the proposed matching algorithm are evaluated, including full-cost matching, new tracks filtering and low-score detection rematching. As shown in Table 4, when the module of full-cost matching is removed, the value of IDF1 drops significantly, and the number of ID switches increases, which means it is more difficult for the tracking system to continuously track the same target, again reflecting the importance of multiple information aggregation. When removing the new tracks filtering module, the MOTA value drops by 2.384% and the IDF1 value drops by 4.737%, indicating the effectiveness of this design. When removing the module of low-score detection rematching, the MOTA and IDF1 drop significantly by 4.197% and 6.024%, respectively, which demonstrates that low-score detections should not be ignored directly. When a target is occluded, its position information provided by the low-score detection box will be a critical supplementary.

The visualization results of the proposed matching algorithm on MOT16 test set are shown in Fig. 3. Three kinds of difficult cases common in MOT scenarios are selected. The results show that the proposed matching algorithm is robust to ambient light interference, motion blur, and partial occlusion, and can effectively maintain trajectories for continuous tracking.



(a) Ambient light interference

(b) Motion blur

(c) Partial occlusion

Fig. 3. Visualization results of the proposed matching algorithm on MOT16 test set. Targets marked with yellow triangle represent difficult cases due to ambient light interference, motion blur, or occlusion.

4 Conclusion

In this paper, position, motion and appearance information are first incorporated into the calculation of the hybrid distance matrix through information aggregation, and then a cascade matching algorithm is proposed based on detection box scores, which divides the entire matching process into three parts. The effectiveness of the proposed methods is verified by a series of ablation experiments on the MOT16 dataset.

References

1. Bernardin, K., Stiefelhagen, R.: Evaluating multiple object tracking performance: the CLEAR MOT metrics. EURASIP J. Image Video Process. **2008**(1), 1–10 (2008). https://doi.org/10.1155/2008/246309
2. Bewley, A., Ge, Z., Ott, L., Ramos, F., Upcroft, B.: Simple online and realtime tracking. In: 2016 IEEE International Conference on Image Processing (ICIP), pp. 3464–3468. IEEE (2016)
3. Ge, Z., Liu, S., Wang, F., Li, Z., Sun, J. YOLOX: exceeding yolo series in 2021. arXiv preprint arXiv:2107.08430 (2021)
4. Yolov5 [EB/OL]. https://github.com/ultralytics/yolov5
5. Trackeval [EB/OL]. https://github.com/JonathonLuiten/TrackEval
6. Kalman, R.E.: A new approach to linear filtering and prediction problems **82D**, 35–45 (1960)
7. Kuhn, H.W.: The Hungarian method for the assignment problem. Nav. Res. Logistics Q. **2**(1–2), 83–97 (1955)
8. Mahmoudi, N., Ahadi, S.M., Rahmati, M.: Multi-target tracking using CNN-based features: CNNMTT. Multimedia Tools Appl. **78**(6), 7077–7096 (2018). https://doi.org/10.1007/s11042-018-6467-6
9. Milan, A., Leal-Taixé, L., Reid, I., Roth, S., Schindler, K.: MOT16: a benchmark for multi-object tracking. arXiv preprint arXiv:1603.00831 (2016)
10. Ristani, E., Solera, F., Zou, R., Cucchiara, R., Tomasi, C.: Performance measures and a data set for multi-target, multi-camera tracking. In: Hua, G., Jégou, H. (eds.) ECCV 2016. LNCS, vol. 9914, pp. 17–35. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-48881-3_2
11. Shao, S., et al.: CrowdHuman: a benchmark for detecting human in a crowd. arXiv preprint arXiv:1805.00123 (2018)
12. Wei, L., Zhang, S., Gao, W., Tian, Q.: Person transfer GAN to bridge domain gap for person re-identification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 79–88 (2018)
13. Wojke, N., Bewley, A., Paulus, D.: Simple online and realtime tracking with a deep association metric. In: 2017 IEEE International Conference on Image Processing (ICIP), pp. 3645–3649. IEEE (2017)
14. Zhang, Y., et al.: Bytetrack: multi-object tracking by associating every detection box. arXiv preprint arXiv:2110.06864 (2021)
15. Zhou, K., Yang, Y., Cavallaro, A., Xiang, T.: Omni-scale feature learning for person re-identification. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 3702–3712 (2019)