Online Multi-object Tracking Based on Deep Learning

Zheming Sun[1], Chunjuan Bo[2(✉)], and Dong Wang[1]

[1] School of Information and Communication Engineering, Dalian University of Technology, Dalian, China
[2] School of Information and Communication Engineering, Dalian Minzu University, Dalian, China
bcj@dlnu.edu.cn

Abstract. Multi-object tracking task aims to identify and track all targets in the video. It has important applications in intelligent monitoring and other fields. Two problems can affect the accuracy of the multi-object tracking task. First, occlusion between targets will lead to interruption of tracking trajectory and switch of tracking target. Second, quality of the object detection results will directly affect the tracking accuracy. In this paper, we adopt a single-object tracking algorithm based on deep learning is introduced to solve the first problem and develop a discriminant network scoring the accuracy of detection and prediction bounding boxes to solve the second problem. The experimental results show that the proposed tracker performs better than other competing methods.

Keywords: Multi-object tracking · Deep learning · Discriminant network

1 Introduction

Multi-object tracking aims to identify the target that appears in each frame of a video sequence and the same target in a continuous sequence of images. The tracking target can be pedestrians and cars. Multi-object tracking task for pedestrians is the main direction because of its practicality. Multi-object tracking task is the fundamental principle of action recognition, behavior analysis, and other fields and plays an important role in public safety and human-computer interaction.

Multi-object tracking tasks can be divided into detection-based (DBT) [1,2] and detection-free (DFT) [3–5] tracking from the perspective of target initialization. DBT refers to the tracking based on the results of detection. First, we use the detector to determine the target in the video, followed by identity matching based on the detection results. The accuracy of the detector will remarkably affect the accuracy of multi-object tracking tasks in this mode. DFT refers to the manual labeling of the target in the first frame and tracking targets in subsequent frames. DBT is used more than DFT because the latter fails to solve the problem of a new target or disappearing target in subsequent frames.

Q. Liang et al. (Eds.): CSPS 2021, LNEE 879, pp. 1–9, 2022.
https://doi.org/10.1007/978-981-19-0386-1_1

The DeepSORT algorithm [13] is improved on the basis of the cost matrix in the Hungarian algorithm and introduces an additional cascade matching before iou matching and utilizes the appearance feature and Mahalanobis distance. Appearance features are extracted from the Re-ID network. The Mahalanobis distance is used as a constraint in the motion information because Euclidean distance ignores the calculation of the spatial domain distribution. DeepSORT first uses the Kalman Filter to predict the trajectory of the target and then uses the Hungarian algorithm to match predicted trajectories with detections in the current frame and finally updates trajectories with the Kalman Filter.

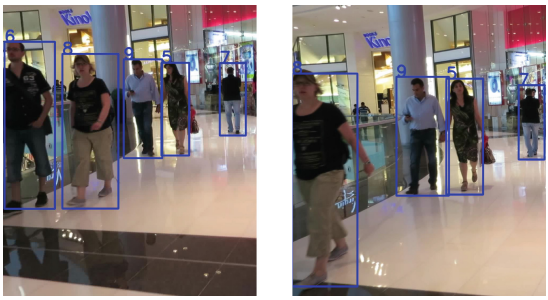


Fig. 1. Results of our algorithm on the MOT16 dataset.

Object detection algorithm based on the deep convolutional neural network (CNN) also appears after the deep CNN obtains excellent results on the classification problem. R-CNN series object detection algorithm [6–8] aims to generate and classify bounding boxes of targets. The algorithm classifies the background and foreground in the first stage by determining the anchor box of the target in the picture. The algorithm classifies targets of the rectangular anchor box in the second stage. Faster R-CNN [8] uses neural networks to generate bounding boxes of targets rather than generate candidate boxes with artificial rules to achieve end-to-end training and increase the algorithm speed.

Appearance models, including both visual features of the target and measure of similarity between targets, such as corner points [9], color histograms [10], optical flow [11], gradient features [12], are often used in multi-object tracking tasks. Each feature has its advantages and disadvantages and is sometimes ineffective. The simple color histogram can easily calculate similarity. However, it only obtains statistics but loses pixel location information. Corner points are effective for transformation within the plane but ineffective when it comes to occlusion and out-of-plane. Gradient features positively affect light changes but fail to solve deformation and occlusion. A deep feature extraction network is added in this study to replace manually designed and general visual features.

In this paper, an online multi-object tracking algorithm, including single-object tracker, discriminant network, and deep feature extraction network, is

designed. First, a single-object tracker is adopted to predict the target trajectory and deal with occlusion and background complexity. Second, a discriminant network is designed and trained to choose the more accurate candidate box generated by the detector and tracker.

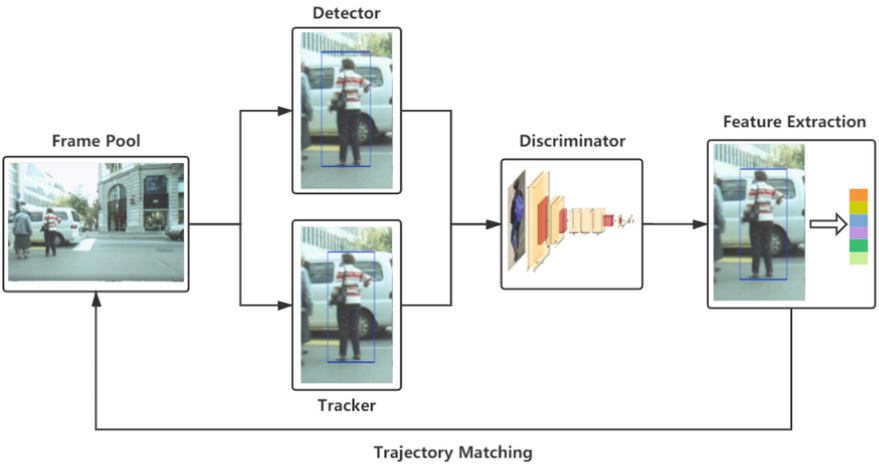


Fig. 2. Flow chart of our algorithm.

2 Proposed Method

First, the target position is determined with a detector. Second, we extract deep features of each target and use the tracker to predict the target trajectory. Finally, we use the discriminant network to score bounding boxes given by the tracker and detector and then match the candidate box with the target trajectory. The flow chart of our algorithm is shown in Fig. 2. The following sections are described separately.

2.1 Detection and Feature Extraction Based on Deep Learning

Multi-object tracking is based on visual detection. We start with detecting all targets that appear in each frame before subsequent processing. Faster R-CNN is used as the target detector in this study. Limitations of using only the motion information to track the target in the multi-object tracking task are as follows. The ineffective motion information processing for addressing variable motion, change of movement direction, and occlusion leads to low tracking accuracy. To this end, adding the appearance information of the target can improve the identification and matching of the target and this facilitate the tracking of the target. To be specific, we adopt ResNet-50 as the backbone to extract deep visual features. The last fully connected layer is removed from the ResNet-50 network structure to obtain a final output of a 2048-dimensional vector, which highly

refines the appearance information for matching between pedestrian targets. We use the Market-1501 dataset to train the ResNet-50 network and obtain the deep feature extraction network. Market-1501 includes 1501 pedestrians taken by six cameras and 32668 pedestrian bounding boxes, which is very commonly used for person re-identification.

2.2 Single-Object Tracker

The single-object tracking algorithm has progressed considerably in recent years. The single-object tracker has a remarkable effect in dealing with target deformation and occlusion. A single-object tracking algorithm is introduced in this study to solve the ID switch problem caused by the occlusion between targets in the multi-object tracking task and predict the pedestrian trajectory.

Traditional tracking algorithms, such as Kalman filter and particle filter, can successfully solve the tracking problem in simple scenarios. However, such algorithms perform poorly in complex scenarios, such as multi-object tracking. A single-object tracking algorithm based on deep learning, such as SiamMask, can successfully track objects in complex scenes.

The SiamMask algorithm [15] can perform both video target tracking and video target segmentation tasks simultaneously. The algorithm achieves target segmentation by adding a mask branch to the full convolutional Siamese neural network for target tracking, and enhances the loss to optimize the network. Once trained, the SiamMask algorithm can achieve real-time category-independent target tracking and segmentation. This model is simple, versatile, fast, and outperforms other tracking methods.

The SiamMask algorithm is used to predict the position of the target in the new frame and obtain the trajectory prediction results of each target in the current frame. At the same time, the detection algorithm is used in this study to detect the target in the current frame to obtain the detected bounding box. We then utilize the matching algorithm between bounding boxes obtained by the tracker and the detector. The matching results are used for updating the target trajectory.

2.3 Discriminator

The detector can find the new target in the scene in a timely manner in the multi-object tracking task. However, the detection quality may be poor in case of cluttered background, occlusion, and interaction between targets. The single-object tracking algorithm is suitable for these problems. Selecting the more accurate finding between the detection and tracking results will affect the accuracy of the task. To this end, a discriminant model is proposed to score the confidence of detection and tracking boxes. The result with a higher score is used to update the target trajectory. The structure of the discriminant network is shown in Fig. 3.

The network consists of eight convolutional layers and two fully connected layers. The input size of the pedestrian picture is unified to 200×100. Convolutional layers aim to extract general visual features of pedestrians and are

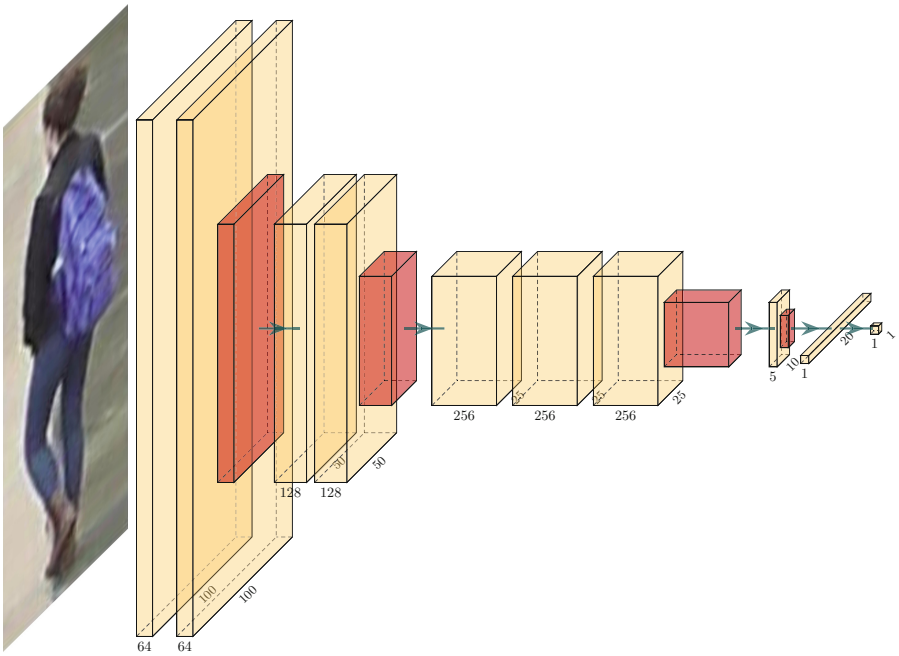


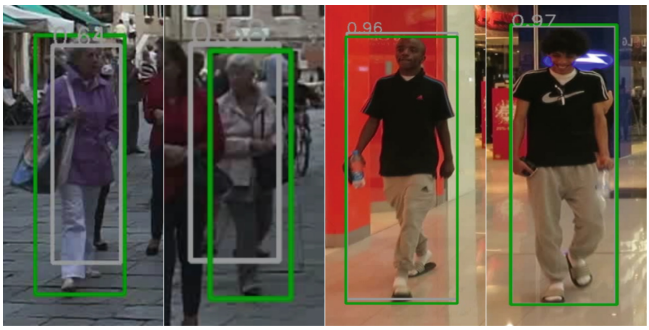
Fig. 3. Structure of the discriminant network that contains eight convolutional and two fully connected layers. The input size is 200×100 and the output is a score.

connected to fully connected layers to output the score. We collect the pedestrian bounding box as the training set and use the iou value between the bounding box and groundtruth as the score of the bounding box. The large iou value between the bounding box and groundtruth indicates a high score. The discriminator can score the input pedestrian bounding box after training. The bounding box is considered satisfactory if its score is high. Figure 4 shows the process of collecting the training set.

3 Experiment

3.1 Dataset

MOT16, a dataset for pedestrian tracking that was proposed in 2016, is used to measure detection and tracking methods for multi-object tracking [14]. Evaluation indicators of the MOT16 dataset include MOTA, MOTP, ID switch (IDS), false positive (FP), and false negative (FN). MOTA evaluates all object matching errors during the tracking process, which contains false negative, false positive, and ID switch. This indicator measures the performance of matching targets and maintaining the trajectory during tracking. MOTA values are less than 1 but



(a) iou=0.64 (b) iou=0.58 (c) iou=0.96 (d) iou=0.97

Fig. 4. Process of collecting training samples. The green bounding box is the groundtruth. The gray bounding box is generated by the detector and the tracker, with the iou of the two bounding boxes as the score.

Table 1. Comparison of our tracker with other algorithms on the MOT16 dataset.

Method	MOTA	MOTP
EAMTT [16]	38.8	75.1
TBD [17]	33.7	76.5
JPDA [18]	26.2	76.3
Ours	44.4	76.8

can become negative when the number of errors in the tracking is larger than the total number of groundtruths. MOTA can be expressed as follows:

$$MOTA = 1 - \sum(FN + FP + IDS) \Big/ \sum GT. \tag{1}$$

MOTP is used to quantify the positioning accuracy of the detector and contains nearly zero unrelated information to the actual performance of the tracker. The MOT16 dataset uses the overlap rate of the bounding box as the measurement.

$$MOTP = \sum_{i,t} d_{i,t} \Big/ \sum_{t} c_t, \tag{2}$$

where d is the average measurement distance between the detection target i and the groundtruth assigned to it and c is the number of successful matches in frame t.

3.2 Results

We test our algorithm on the MOT16 dataset. Figure 1 shows a portion of the results of our algorithm. Table 1 presents the comparison between our algorithm

Table 2. Ablation study of our tracker.

Method	MOTA	MOTP
Baseline	35.3	76.6
Baseline+SOT	44.0	77.0
Baseline+SOT+DIS	44.4	76.8

and other approaches. The satisfactory performance of the algorithm is indicated by the high values of performance indicators MOTA and MOTP. A comparison experiment (in Table 2) is conducted to explore the effects of using a single-object tracker to predict the target location and the accuracy of the discriminator.

The first line in Table 2 is our baseline, which uses the faster R-CNN detector to detect the target. This baseline extracts deep appearance features of the detected target and uses the Kalman filter to predict the possible position of the target in the next frame. We then match predicted and detected positions using the Hungarian algorithm and update the trajectory.

The second line in Table 2 is used to replace the Kalman filter with the single-object tracker, SiamMask, on the basis of the baseline algorithm. Using a deep-learning-based single-object tracker is suitable for predicting the target position because the Kalman filter performs poorly under complex motion conditions. We then match the tracking and detection results and update trajectories using the detection bounding box. The results improved significantly.

The third line in Table 2 is used to add the single-object tracker SiamMask and the discriminator to the baseline. We obtain the target detection and target prediction results of the current frame. We then use the discriminator to score the two kinds of bounding boxes and the one with the higher score is used to update the target trajectory. The total accuracy of MOTA improves after adding the discriminator.

4 Conclusion

This paper presents a novel algorithm for multi-object tracking. A single-object tracker is added to solve the multi-object tracking problem, predict the target trajectory, and reduce the impact of target occlusion in tracking. In addition, a deep discriminant network is proposed to improve the accuracy by selecting the candidate box with higher confidence between tracking and detecting results to update the trajectory. The experimental results show that our tracker achieves performs better than other competing methods.

Acknowledgement. This work was supported in part by the National Natural Science Foundation of China under Grant 61806037, in part by the Natural Science Foundation of Liaoning Province under Grant 2019-MS067, and in part by the Youth Technology Star Project of Dalian City under Grant 2018RQ57.

References

1. Bose, B., Wang, X., Grimson, E.: Multi-class object tracking algorithm that handles fragmentation and grouping. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8 (2007)
2. Song, B., Jeng, T.-Y., Staudt, E., Roy-Chowdhury, A.K.: A stochastic graph evolution framework for robust multi-target tracking. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) Computer Vision – ECCV 2010: 11th European Conference on Computer Vision, Heraklion, Crete, Greece, 5–11 September 2010, Proceedings, Part I, pp. 605–619. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-15549-9_44
3. Hu, W., Li, X., Luo, W., Zhang, X., Maybank, S., Zhang, Z.: Single and multiple object tracking using log-euclidean Riemannian subspace and block-division appearance model. IEEE Trans. Pattern Anal. Mach. Intell. **34**(12), 2420–2440 (2012)
4. Zhang, L., van der Maaten, L.: Structure preserving object tracking. In: 2013 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1838–1845 (2013)
5. Yang, M., Yu, T., Wu, Y.: Game-theoretic multiple target tracking. In: 2007 IEEE International Conference on Computer Vision, pp. 1–8 (2007)
6. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition, pp. 580–587 (2014)
7. Girshick, R.: Fast R-CNN. In: 2015 IEEE International Conference on Computer Vision, pp. 1440–1448 (2015)
8. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. In: 2015 Advances in Neural Information Processing Systems, pp. 91–99 (2015)
9. Brostow, G., Cipolla, R.: Unsupervised Bayesian detection of independent motion in crowds. In: 2006 IEEE Conference on Computer Vision and Pattern Recognition, pp. 594–601 (2006)
10. Okuma, K., Taleghani, A., de Freitas, N., Little, J.J., Lowe, D.G.: A boosted particle filter: multitarget detection and tracking. In: Pajdla, T., Matas, J. (eds.) ECCV 2004. LNCS, vol. 3021, pp. 28–39. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-24670-1_3
11. Ali, S., Shah, M.: Floor fields for tracking in high density crowd scenes. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) Computer Vision – ECCV 2008: 10th European Conference on Computer Vision, Marseille, France, 12–18 October 2008, Proceedings, Part II, pp. 1–14. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-88688-4_1
12. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: 2005 IEEE Conference on Computer Vision and Pattern Recognition, pp. 886–893 (2005)
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, pp. 3645–3649 (2017)
14. Milan, A., Leal-Taixé, L., Reid, I., Roth, S., Schindler, K.: MOT16: a benchmark for multi-object tracking. arXiv preprint arXiv:1603.00831 (2016)
15. Wang, Q., Zhang, L., Bertinetto, L., Hu, W., Torr, P.: Fast online object tracking and segmentation: a unifying approach. In: 2019 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1328–1338 (2019)

16. Sanchez-Matilla, R., Poiesi, F., Cavallaro, A.: Online multi-target tracking with strong and weak detections. In: Hua, G., Jégou, H. (eds.) ECCV 2016. LNCS, vol. 9914, pp. 84–99. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-48881-3_7
17. Geiger, A., Lauer, M., Wojek, C., Stiller, C., Urtasun, R.: 3D traffic scene understanding from movable platforms. IEEE Trans. Pattern Anal. Mach. Intell. **36**(5), 1012–1025 (2014)
18. Rezatofighi, H., Milan, A., Zhang, Z., Shi, Q., Dick, A., Reid, I.: Joint probabilistic data association revisited. In: 2015 IEEE International Conference on Computer Vision, pp. 3047–3055 (2015)