Neurocomputing 412 (2020) 244–251

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Jointly detecting and multiple people tracking by semantic and scene information

Zhixiong Pi, Huai Qin, Changxin Gao, Nong Sang *

Key Laboratory of Image Processing and Intelligent Control, School of Artificial Intelligence and Automation, Huazhong University of Science and Technology, China

ARTICLE INFO

Article history:
Received 3 October 2019
Revised 26 April 2020
Accepted 13 June 2020
Available online 2 July 2020
Communicated by Zhen Lei

Keywords:
Online multiple object tracking
Detection
Semantic information
Scene information

ABSTRACT

In this paper, we propose a new method for online multiple people tracking, which combines the detection process and the single object tracking process, and establishes the interactions between them. The detector detects objects in the still images which ignores the sequential information. Meantime, the single object tracker does not use the category semantic information during tracking. To take both the sequential and semantic information into account, we exchange information among the detector and the trackers. More specifically, the trackers deliver sequential information to the detector by providing the detector with the extra proposals. The detector supplements each tracker with the robust semantic information by using bounding box regression to modify the tracking result. Besides, the interactions also happen among the trackers through the occlusion speculation, the perspective model interpretation and the trajectory merging process. The experimental results demonstrate that the proposed algorithm performs favorably against the state-of-the-art MOT methods.

1. Introduction

Multiple object tracking (MOT) is an important task in computer vision for its potential influence on many applications, such as autonomous driving, robot navigation, and motion analysis. The goal of MOT is to estimate the locations of targets in each frame of a video and keep their specific identities consistent, to obtain their trajectories. Conventionally, the MOT task is separated into detecting and data association steps. Object patches are provided by a detector, and matched with trajectories existed in the data association step. The two steps are executed independently. In the view of modeling methods, MOT algorithms can be separated to two branches. One concerns about to simplify the MOT problem as an abstract model whose optimal solution can be obtained by some solvers. The simplified models include network-flows [44], k-partite graph [43], and graph multi-cut [38], etc. Another branch tries to model the MOT problem as complete as possible, and to derive a good local optimal solution. For modeling the complex situations, complicated energy functions [26] are defined empirically, or deep neural networks [28,37] are constructed to obtain a good solution.

The conventional MOT frameworks, which detect objects in a frame and then associate detections with the trajectories existed, depend heavily on the detection results. The wrong association happens, once the detections are incorrect or missed. In fact, a target detected in the previous frame tends to be found near its latest location in the current frame. To use this sequential information, several algorithms [39,7] utilize single object tracking methods to alleviate the influence of false or inaccuracy detections. In recent years, the single object tracking methods have improved significantly because of taking advantages of sequential information efficiently. A target can be tracked accurately by the state-of-the-art single object tracker [29,13] in most situations without the severe occlusion and the frequent target deformation. Yet unfortunately, the serious occlusion and the target deformation frequently happen in the MOT task, especially when the target is people. In such a complex scene, it performs terribly to use the single object tracking method for every target simply in the MOT task. The tracking drift and the scale bias happen easily, as shown in Fig. 1. These tracking failure situations derive from the model updating method. When updating the appearance model, the tracker uses the previous tracking results as positive examples. Then, the updated model is employed to find the target in the current frame. However, the accumulative error increases with the tracking process, which will result in the tracking failure. We argue that the essential reason for tracking drift of the single object tracker is the lack of the semantic information about the target category. The tracker always tries to find the patch similar to its appearance model the most, regardless of the target category.

* Corresponding author.
E-mail address: nsang@hust.edu.cn (N. Sang).

https://doi.org/10.1016/j.neucom.2020.06.076
0925-2312/

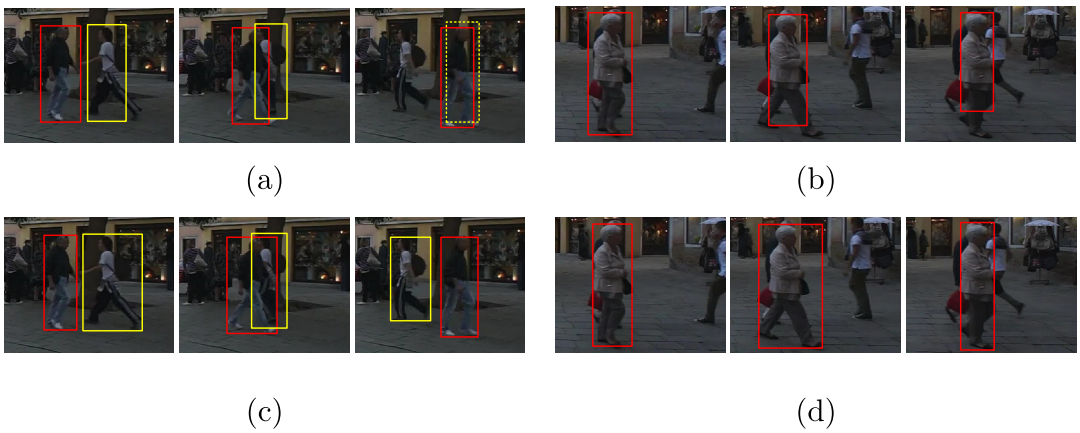

Fig. 1. Inaccuracy tracking results and modified results. The tracking drift (a) and the scale bias (b) occur when adopting CF tracking directly in the MOT task. These problems are fixed by our method (c) and (d).

Motivated by the above observations, we propose a method for the online multiple people tracking task, which unites the tracker of each target and the detector, called detector-tracker cloud. The connection is built between the detector and the trackers, as well as every tracker. In our method, trackers help the detector to find some false negatives by providing the extra proposals with the specific IDs. And the detector fine-tunes the tracking bounding boxes via the category semantic information. Exchanging information between the detector and each tracker comes from two reasons: (1) The detector can supplement the category semantic information to the tracker, and fine-tunes the bounding box more precisely, which alleviates the accumulative error. (2) The trackers can provide the detector with extra proposals to recover some false negatives, which stabilizes the target trajectories. Besides, We propose a new perspective model. Through the construction and interpretation of the perspective model, trackers influence each other and control the size of the tracking bounding boxes in a reasonable range. For data association, we separate the step into two stages. Targets who are tracked associate the detections first, which is easier and more reliable. And the rest detections are matched with lost targets after that. Then, we further drop or merge redundant trackers during the trajectory merging process.

The main contributions of our work can be summarized as:

- We propose a new method to solve online multiple people tracking task, referred to detector-tracker cloud, which combines the single object tracker and the object detector by sharing the category features and the sequential information to stabilize the tracking trajectories.
- we propose an adaptive perspective model for helping the tracker to select the correct target and keep the bounding box size in a reasonable range.
- we design a new trajectory management method which includes the re-ID based re-tracking process and the trajectory merging process, to suppress the risk of the trajectory fragment and the ID switch.

2. Related work

2.1. Multiple object tracking

Multiple object tracking algorithms can be classified into two types in general. One type of methods simplify the MOT problem to an abstract model which can be obtained the optimal solution. Typically, some methods use network-flows [44,14,5] algorithm to find global optimal trajectories association. Several other optimization algorithms are also employed to search the optimal solution, such as k-partite graph [43,8], multi-cut [38], etc. Different from the methods deriving the optimal solution via the simplified models, another type of methods construct complex models to approach to the reality, and get a good local optimal solution. For approaching the reality, more complicated energy functions [26,27] are defined for obtaining a better solution. Xiang et al. [39] infers target states with Markov decision process. Milan et al. [28] integrates data association, appearance model updating, motion predicting and trajectory management in a unified RNN architecture. Besides, deep networks also are used to encode different information like appearance, motion and interaction [34], or infer occlusion situation [7]. These previous works focus on how to get better features to distinguish different targets and overcome heavy occlusion. However, our work focus on how to build connections between detector and trackers.

2.2. Object detection

Object detection is developing rapidly, which is used in many computer vision tasks as a basic technology. Faster-RCNN architecture is one of the mainstream algorithms to solve the object detection [33], which uses a two-stage network. Another direction is detecting objects in one stage [31,25], which inputs image into a single network, and locates objects by regression directly. In our work, we use a simple Faster-RCNN as the basic detector to encode the category semantic information. It is certain that the algorithm performance will be improved by a large margin if a better detector is adopted. However, considering the object detection is not the main topic of this paper, we just use a simple architecture as the basic detector to demonstrate our idea.

2.3. Single object tracking in MOT

Recently, In MOT task, several works also try to use single object tracking methods, such as TLD [39], CNN based tracking [7], particle filter tracking [32,35] and so on. A few works just use single object tracking to generate initial short tracklets [40] or estimate a candidate area [21]. To the best of our knowledge, there is no study to focus on the interactions inside the MOT system. We argue that each single object tracker connects with others instead of keeps alone. Besides, trackers with sequential information and detector with semantic information are complementary.

3. Our approach

3.1. Overview

We propose a detector-trackers cloud system for multiple people tracking. Faster-RCNN is used as the detector to detect persons in each frame. For the tracking process, we generate a tracker for every target detected by the detector. For each tracker, the basic task is to track the object assigned to it. We employ one of the correlation filter (CF) tracking methods to handle the basic task. Considering the efficiency, we employ the Sum of Template And Pixel-wise LEarners (Staple) tracker as our basic single object tracker, which operates at 80 fps. These trackers influence each other through the occlusion speculation, the perspective model and the trajectory merging step. The pipeline is shown in the Fig. 2.

3.2. Interactions between trackers and detector

We believe that the detector and trackers can help each other in MOT task. From this base, we design the Interaction process between the detector and trackers.

3.2.1. Detector helps trackers

The single object tracker is designed for the specific instance. The lack of the category semantic information results in the less robust and precise tracking bounding box. In the MOT task, there are much more difficult situations than single object tracking, such as frequent occlusion, more distractors and more background clutters, etc. Tracking drift often occurs without the category semantic information, so we use the bounding box regression to fine-tune the tracking bounding boxes that the trackers output.

For the targets in tracked state, We divide the conventional matching task into three steps: The first step is to track every target in tracked state with the CF tracker. The second is to modify the tracking bounding boxes with the category semantic features via the bounding box regression; The third step is to match the modified tracking bounding boxes and the candidate patches from the detector, and choose the higher confidence region as the target position. The tracking bounding boxes provided by the CF trackers are not precise enough due to the lack of the category semantic information, so we use the deep network shown in the Fig. 3basic net) to extract the deep features with the category semantic information. The ROI pooling layer is employed to crop the deep features of the patches localized by the CF trackers. Then, the tracking bounding boxes are fine-tuned by the regression process to obtain the more reliable bounding boxes (see Fig. 4).

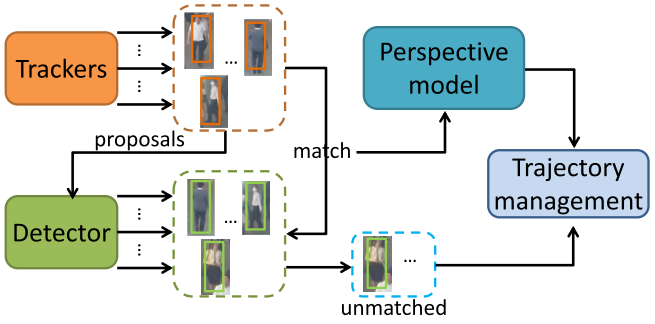

Fig. 2. Tracking pipeline. The trackers locate the targets first, and provide the tracking bounding boxes as the extra proposals of the detector. The detector fine-tunes all the original and extra proposals via the bounding box regression branch. Then, the tracking bounding boxes are matched with the detection bounding boxes. The matched boxes are sent to the perspective model for the further select. Then, the trajectory management is implemented based on all the tracked targets, unmatched detection boxes and lost targets.

3.2.2. Trackers help detector

The detector is designed to detect targets in the still images. Via learning the category semantic information, it can localize the targets with the precise bounding boxes. However, the detector is not designed for the specific instance and does not use the motion consistency. Sometimes, the trajectories are not continuous because of the targets being wrongly suppressed, if we only utilize the detector to localize the targets. It is clear that the target exists at the position near where he shows in the previous frame. Hence, we add the tracking results as extra proposals for the detector, which improves the detection quality. Notably, these extra proposals are different from normal proposals because of their id information (each of them corresponds to a specific target). What's more, they are, generally, closer to real targets than normal proposals due to the continuous tracking process. There are two advantages of the extra proposals: First, the real targets can be located easily by the detector according to them. Second, They can be used to localize the specific targets before the conventional matching process. The trackers can not only provide more accurate proposals for the detector, but also match detection bounding boxes for the targets in tracked, called the stage one match.

3.2.3. Re-tracking the lost targets

When a target is lost, the corresponding tracker tries to find it again in the next frames among the candidate patches provided by the detector, which is called the stage two match. We use three convolution layers and three fully connected layers to extract the feature vectors for the candidate patches, which is shown in Fig. 3 (detail net). Afterwards, these feature vectors are used to compute the distance metric with the feature vectors of the trajectories that are lost. We calculate and update the feature vector of a trajectory every frame if the target is tracked by the tracker. Assuming that the ith trajectory τ_i who is tracked continuously from the frame a, matches with the kth candidate patch in the frame t, we describe the updating formula as

$$x_i^{a:t} = (1-\omega) * x_i^{a:t-1} + \omega * x_k^t, \tag{1}$$

where $x_i^{a:t}$ is the feature vector of target i who is tracked continuously, x_k^t is the feature vector of an image patch matched with τ_i in the current frame, and ω is the weight calculated according to the confidence scores of the trajectory in the past and the current tracking result. If the current tracking result is credible, we set a big weight for the new feature vector to update the target feature vector. we define the ω

$$\omega = \frac{S_{i,k}^t}{S_i^{a:t-1} + S_{i,k}^t}, \tag{2}$$

$$S_{i,k}^t = P_k^{rec} P_{i,k}^{match}, \tag{3}$$

$$S_i^{a:t-1} = \frac{1}{n}\sum_{j=a}^{t-1} S_i^j, \tag{4}$$

where $S_i^{a:t-1}$ is the confidence scores of the trajectory τ_i until $t-1$ frame and $S_{i,k}^t$ is the confidence score of its current tracking result. S_i^j represents the tracking result confidence score in each frame of the trajectory being tracked continuously. P_k^{rec} is probability calculated by basic net. $P_{i,k}^{match}$ is the cosine affinity between the trajectory and detection feature vectors:

$$P_{i,k}^{match} = \frac{x_i^{a:t-1} x_k^t}{\|x_i^{a:t-1}\| \|x_k^t\|} \tag{5}$$

When the target is lost in frame t, the feature vector x_i remains unchanged and is used to calculate cosine affinities with feature vectors of candidate patches in following frames. The tracker finds its target again when the maximum cosine affinity is larger than a threshold.

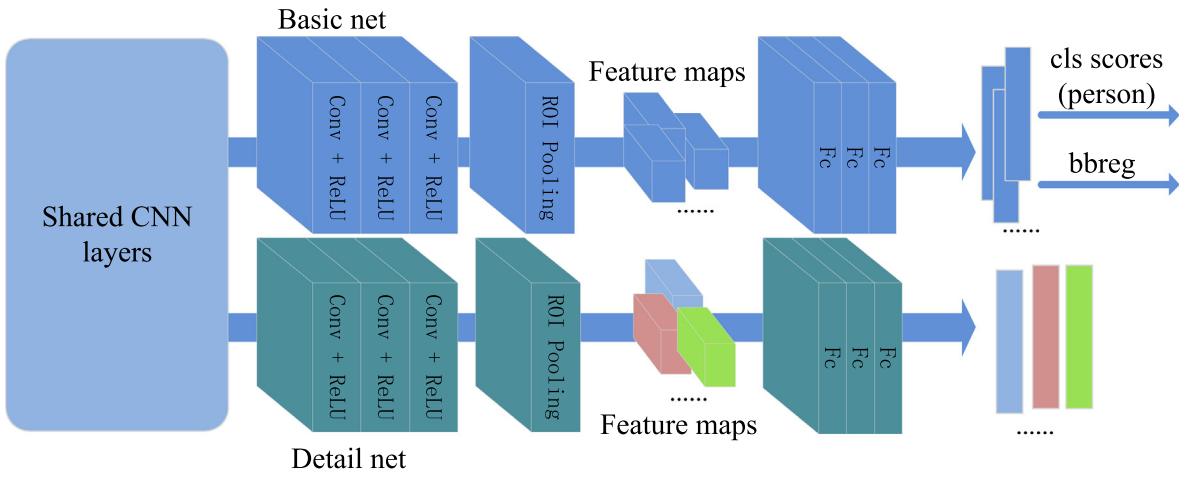

Fig. 3. Basic net and detail net. The basic net is used to encode the category semantic information and detect the people in the frames. The detail net is employed to distinguish the different persons. The parameters of the two networks are partially shared.

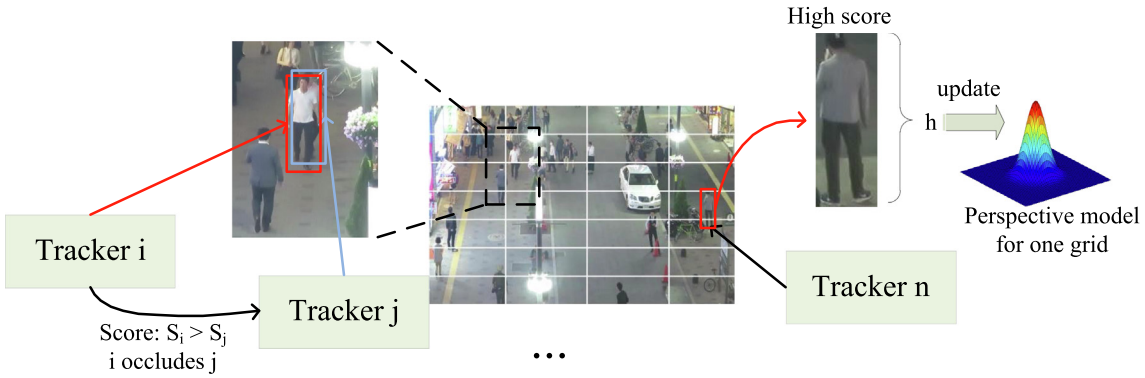

Fig. 4. Interactions between trackers. When several trackers are close to each other, these trackers speculate their occlusion situations by comparing the confidence scores of tracking results. Besides, every tracker with high confidence tracking result updates the perspective model of a grid where the bottom-center point of the bounding box is.

The stage two match is executed after the stage one match. Actually, the stage one match is easier, due to the use of the sequential information. The stage one match rules out some candidate patches, which simplifies the match process in the stage two.

3.3. Interactions between trackers

The trackers track their targets in the same scene. We argue that these trackers can share the information they obtain, which improves the whole multiple object tracking task. The trackers can speculate occlusion situations and perceive the scene that they stay in through sharing information.

3.3.1. Speculating occlusion

When some trackers get close to each other, we introduce information exchange among the subgroup of trackers. The confidence of a tracking result is not only depended by a single tracker, but also influenced by the trackers near it. The occlusion between targets happened gradually. For this reason, the tracker whose target is occluded will adapt to another target gradually with the tracking and updating process. Then, the tracking drift occurs. We define the tracking confidence of a tracker as Formula 3, and compare the tracking results of different trackers near each other. When the IOU between two tracking results is large, we consider the tracking result with lower confidence(regardless the absolute score) is occluded by the other, and the tracker loses the target.

3.3.2. Perceiving scene

We propose the perceiving scene process. Considering that the height of people is in a small range, and the viewpoint of a camera is only with the small variation, the perspective relationship in a scene can be modeled. We let the trackers perceive the perspective relationship with themselves during the tracking process. We separate the scene into some grids with the same size, and build a gaussian-like model for every grid to describe the reasonableness of the bounding boxes in the grid according to their heights. We call it perspective model. And in each grid

$$P_{pers} = e^{-\frac{(h-\mu)^2}{2*\sigma}}, \tag{6}$$

where h is the height of bounding box of a target. μ and σ are expectation and variance of the gaussian-like model, which could be various in different grids.

The perspective model is updated during the tracking process. In every frame, the tracking bounding boxes with the high confidence scores are used for updating the gaussian-like models in the grids where they are. Derived from the conjugate prior distribution, the updating formulas for μ and σ are

$$\mu := \frac{N * \mu + \sum_{i=1}^{n} h_i}{N + n}, \tag{7}$$

$$N := N + n, \tag{8}$$

$$\alpha := \alpha + \frac{n}{2}, \tag{9}$$

$$\beta := \beta + \frac{\sum_{i=1}^{n} (h_i - \mu)^2}{2}, \tag{10}$$

$$\sigma^2 := \frac{\beta}{\alpha - 1}, \tag{11}$$

where α and β are the parameters of gaussian distribution's conjugate prior distribution (inverse gamma distribution). h is the height of a bounding box. N is the total number of samples in one grid. n is the number of new samples for one grid in the current frame.

The scene information is used to modify the confidence score of the tracking results. We divide the tracking results into three parts with their confidence scores, the correct (>0.7), the incorrect (<0.2) and the undetermined. For the undetermined tracking results, we define the method to modify the confidence scores of the tracking results as follows:

$$S^m = \left(\frac{1}{1 + e^{-\eta(x-0.5)}} + 0.5 \right) S, \tag{12}$$

where S is the original tracking confidence score. η is the modifying factor. The bigger η value means the modification is more significant. It is decide that a tracking result is kept or abandoned after the modification step. These adaptive gaussian-like models tend to select the targets with the normal size and suppress those abnormal bounding boxes. The normal sizes of the different locations are variable in a scene. By using the adaptive perspective model, we control the size of the bounding boxes in a reasonable range.

3.3.3. Merging trackers

Compared with detecting a person, it needs more detail information to re-track a person by matching candidate detections with the lost targets. An example is shown in Fig. 5, a person can be found by the detector when he occluded by something or someone with only the head and a little part of body showing. However, It is hard to recognize who he is in this situation due to the lack of the detail information. In the online MOT task, the re-tracking process is executed frequently due to the targets are lost caused by occlusion. When a missed target shows up from behind the shelter gradually, the detector tends to detect him keeping one step ahead of the tracker who needs to recognize the id of the detection. It is confused for trackers to figure out that the new detections are their targets or not, when these detections are still occluded seriously (can be detected by the detector, though). In this situation, these lost targets are initialized as the new targets with high probabilities. To solve this problem, we calculate the affinities between the lost targets and the new targets. If the affinity between an old target and a new target is bigger than the merging threshold th_m, these two trackers are merged as one. The new trackers also update the trajectory feature vectors with Formula 1, and the affinity of two trackers is calculated by the cosine similarity between their trajectory feature vectors.

3.4. Trajectories management

Another problem in the MOT is to manage trajectories, including generating new trajectories and finishing invalid trajectories. A new target τ^{new} is initialized if a detection with high detection score and low overlaps with all the existed targets. To suppress false positive detections, the new target will be abandoned if it cannot be tracked successfully in the first T_{init} frames. For the target termination, We terminate the targets who are lost for a long time T_{term} or go beyond the field of the view.

4. Implementation details

The CNN used to extract features and detect objects is illustrated in Fig. 3. We employ the first 12 layers(before Conv4_1) of VGG16 as the shared CNN layers. The basic net and the detail net are with the same architecture and different parameters in the additional 3 convolution layers. Then, the basic net finishes bounding box regression and classification as the same as the standard Faster-RCNN. The detail net extracts the feature vectors of the patches, which are used to distinguish the different targets, through 3 fully connected layers. We train the shared CNN layers and the basic net with the training process of the Faster-RCNN, which detects people only. Then, the detail net is trained to classify person id with the shared CNN layers fixed. All the training images from the MOT16 training set without the additional data.

We set the parameters T_{init} and T_{term} as 3 and 200, separately. The tracker is considered as the new (may be merged with old

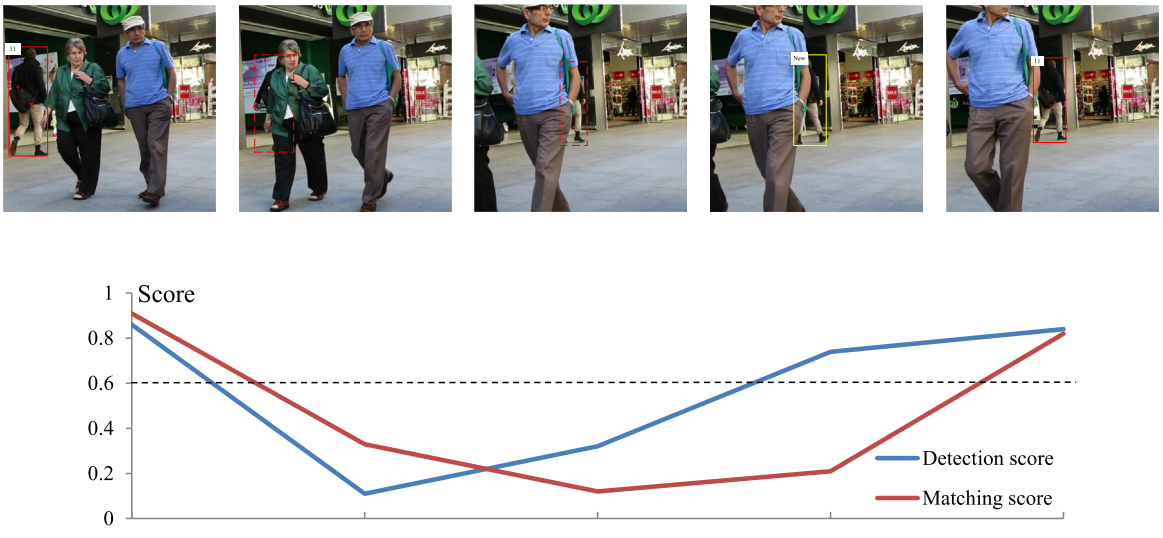

Fig. 5. Merging a wrong new target with an old one. 11th target is lost because of occlusion. When he shows up from behind the shelter, the detector gets a high detection score before the 11th tracker gets a high matching score.

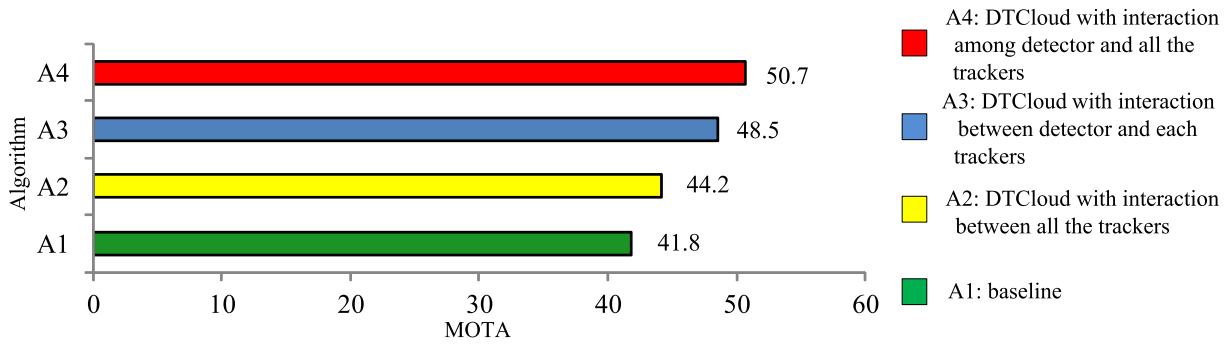

Fig. 6. The performance analysis of our algorithm on training set of MOT16 in terms of MOTA.

trackers), if its trajectory is shorter than T_{new} which is set as 5. The merging threshold th_m is 0.6. When being tracked, a target is lost if the confidence score of the tracking result is lower than 0.6. In the stage one match, a detection and a tracking result are matched with $>$0.5 IoU. In the stage two match, a lost target and a detection are matched when the affinity score is larger than 0.6. In the perspective model, we initialize α and β with 2 and 900. η is set as 1.5.

5. Experimental results

5.1. Datasets

Our online MOT algorithm is evaluated in the public available MOT16 and MOT17 benchmarks. These two benchmarks contain 14 video sequences (7 for training and 7 for testing) in unconstrained environments. Situations like camera shaking, crowded environment, and different perspectives are included. We train our CNN architecture with the training sequences and compare the performance with various state-of-the-art MOT algorithms in the testing set. Then, we do ablation studies in the MOT16 training set.

5.2. Evaluation metrics

The widely used CLEAR MOT metrics [4] are adopted to evaluate the performance of multiple object tracking algorithm. These include Multiple Object Tracking Accuracy (MOTA), Multiple Object Tracking Precision (MOTP), the total number of False Positives (FP), the total number of False Negatives (FN) and the total number of ID Switches (IDS). The metrics defined in [24] are also used, which is composed by Mostly Track targets(MT, percentage of ground truth objects who trajectories are covered by the tracking output for at least 80%), Mostly Lost targets (ML, percentage of ground truth objects who trajectories are covered by the tracking output less than 20%), and the total number of times a trajectory is Fragmented (Frag).

5.3. Ablation studies

We analyze the influence of each part of our framework in this section. The baseline is doing detection and data association separately without any interaction between the detector and trackers, which is noted by A1. A2 represents to connect the detector and each tracker, but without interaction between every two trackers. A3 represents to add interaction between trackers based on the baseline. And A4 is the whole algorithm.

The performance shown in Fig. 6 is evaluated with MOT16 training set according to MOTA metric, which is a good estimation for the overall performance of MOT algorithms. A2 is better than the baseline A1, which demonstrates the effect of interaction between detector and each tracker.That is to say, it is meaningful to deliver semantic information from detector to trackers and supplement sequential information from trackers to detector. The better performance of A3 compared to A1 shows that connecting trackers is also useful, which proves the effectiveness of understanding the environment. Algorithm A4 with all interaction paths obtains the best performance with 8.6% improvement in MOTA compared with the baseline A1, which demonstrates the effectiveness of sharing information between the detector and the trackers instead of doing detection and data association separately.

5.4. Evaluation on testing set

For the comparisons with the other algorithms, we use the public detection results provided by the benchmarks. Our algorithm, named *DTCloud* is evaluated in the MOT16 and MOT17 bench-

Table 1
Tracking performance on the testing sequences of MOT16 benchmark. The results are divided into two groups, i.e. online and offline. The best results of online and offline methods are denoted by red and blue, respectively. The symbol '↑' means the higher is better and ↓ means the lower is better.

Method	Mode	MOTA↑	MOTP↑	MT↑	ML↓	FP↓	FN↓
DP_NMS[30]	offline	26.2%	76.3%	4.1%	67.5%	**3689**	130557
SMOT[9]	offline	29.7%	75.2%	5.3%	47.7%	17426	107552
CEM[26]	offline	33.2%	75.8%	7.8%	54.4%	6837	114322
LINF1[10]	offline	41.1%	74.8%	11.6%	51.3%	7896	99224
QuadCNN[37]	offline	44.1%	76.4%	14.6%	44.9%	6388	94775
MHT_DAM[19]	offline	45.8%	76.3%	16.2%	43.2%	6412	91758
NOMT[6]	offline	46.4%	76.6%	**18.3%**	41.4%	9753	87565
NLLMPa[23]	offline	47.6%	78.5%	17.0%	40.4%	5844	89093
LMP[38]	offline	**48.8%**	**79.0%**	18.2%	**40.1%**	6654	**86245**
oICF[18]	online	43.2%	74.3%	11.3%	48.5%	6651	96515
STAM16[7]	online	46.0%	74.9%	14.6%	43.6%	6895	91117
AMIR[34]	online	47.2%	**75.8%**	14.0%	41.6%	**2681**	92856
Ours	online	**49.0%**	75.6%	**15.8%**	**37.9%**	4116	**87973**

Table 2
Tracking performance on the testing sequences of MOT17 benchmark. The results are divided into two groups, i.e. online and offline. The best results of online and offline methods are denoted by red and blue, respectively. The symbol '↑' means the higher is better and ↓ means the lower is better.

Method	Mode	MOTA↑	MOTP↑	MT↑	ML↓	FP↓	FN↓
LM_NN[1]	offline	45.1%	**78.9%**	14.8%	46.2%	**10834**	296451
MHT_bLSTM[20]	offline	47.5%	77.5%	18.2%	41.7%	25981	268042
TLMHT[36]	offline	50.6%	77.6%	17.6%	43.4%	22213	255030
JCC[17]	offline	51.2%	75.9%	**20.9%**	37.0%	25937	247822
JBNOT[15]	offline	**52.6%**	77.1%	19.7%	**35.8%**	31572	**232659**
HISP_DAL[2]	online	45.4%	77.3%	14.8%	39.2%	21820	277473
MASS[16]	online	46.9%	76.1%	16.9%	36.3%	25733	269116
PHD_GSDL[12]	online	48.0%	77.2%	17.1%	35.6%	23199	265954
AM_ADM[22]	online	48.1%	76.7%	13.4%	39.7%	25061	265495
DMAN[45]	online	48.2%	75.7%	13.4%	39.7%	26218	263608
HAM_SADF[42]	online	48.3%	77.2%	17.1%	41.7%	20967	269038
MTDF[11]	online	49.6%	75.5%	18.9%	33.1%	37124	**241768**
STRN[41]	online	50.9%	75.6%	18.9%	33.8%	25295	249365
Tracktor++[3]	online	**53.5%**	**78.0%**	19.5%	36.6%	**12201**	248047
Ours	online	51.7%	77.4%	**19.6%**	**32.4%**	25058	247465

marks, and compared with the other state-of-the-art MOT algorithms. The performances are shown in the Tables 1 and 2, respectively.

On the MOT16 benchmark, our method achieves the best performance on the MOTA, MT, ML, and FN metrics compared to the online methods, and also outperforms the best offline methods LMP with 0.2% MOTA. As we can see from the Table 1, our method improves 1.8% MOTA compared to the second best online algorithm AMIR.

As shown in the Table 2, on the MOT17 benchmark, compared to all the online methods, our method achieves the second best performance on the MOTA and MOTP metrics, and the best performance on the MT and ML metrics.

6. Conclusions

In this paper, we propose a new algorithm with the interactions between the detector and the trackers for the online multiple people tracking task. We adopt the correlation filter tracker to provide the detector with the sequential information. We use the category semantic information to modify the tracking results of the trackers, which demonstrates the importance of the category semantic information in the MOT task. We also establish a perspective model to evaluate the reasonableness of the tracking results and control the size of the bounding boxes in a reasonable range. The ablation studies prove it is useful to exchange the information inside the MOT system. And the evaluation results on the testing set of the MOT16 and MOT17 demonstrate the effectiveness of our algorithm.

CRediT authorship contribution statement

Zhixiong Pi: Conceptualization, Methodology, Software, Formal analysis, Writing - original draft. **Huai Qin:** Validation, Visualization, Investigation. **Changxin Gao:** Writing - review & editing, Resources, Visualization. **Nong Sang:** Supervision, Project administration, Funding acquisition.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work was partially supported by the National Natural Science Foundation of China (No. 61433007 and No. 61271328).

References

[1] Maryam Babaee, Zimu Li, Gerhard Rigoll, A dual cnn–rnn for multiple people tracking, Neurocomputing 368 (2019) 69–83.
[2] Nathanael L. Baisa, Robust online multi-target visual tracking using a hisp filter with discriminative deep appearance learning. arXiv preprint arXiv:1908.03945, 2019..
[3] Philipp Bergmann, Tim Meinhardt, Laura Leal-Taixe, Tracking without bells and whistles, in: Proceedings of the IEEE International Conference on Computer Vision, 2019, pp. 941–951.
[4] Keni Bernardin, Rainer Stiefelhagen, Evaluating multiple object tracking performance: the clear mot metrics, Eurasip J. Image Video Process. 2008 (1) (2008), 246309.
[5] Asad A. Butt, Robert T. Collins, Multi-target tracking by lagrangian relaxation to min-cost network flow, in: CVPR, 2013..
[6] Wongun Choi, Near-online multi-target tracking with aggregated local flow descriptor, in: ICCV, 2015.
[7] Qi Chu, Wanli Ouyang, Hongsheng Li, Xiaogang Wang, Bin Liu, Nenghai Yu, Online multi-object tracking using cnn-based single object tracker with spatial-temporal attention mechanism, in: ICCV, 2017..
[8] Afshin Dehghan, Shayan Modiri Assari, Mubarak Shah, Gmmcp tracker: globally optimal generalized maximum multi clique problem for multiple object tracking, in: CVPR, 2015..
[9] Caglayan Dicle, Octavia I. Camps, Mario Sznaier, The way they move: tracking multiple targets with similar appearance, in: ICCV, 2013..
[10] Loïc Fagot-Bouquet, Romaric Audigier, Yoann Dhome, Frédéric Lerasle, Improving multi-frame data association with sparse representations for robust near-online multi-object tracking, in: ECCV, 2016.
[11] Fu. Zeyu, Federico Angelini, Jonathon Chambers, Syed Mohsen Naqvi, Multi-level cooperative fusion of gm-phd filters for online multiple human tracking, IEEE Trans. Multimedia 21 (9) (2019) 2277–2291.
[12] Fu. Zeyu, Pengming Feng, Federico Angelini, Jonathon Chambers, Syed Mohsen Naqvi, Particle phd filter based multiple human tracking using online group-structured dictionary learning, IEEE Access 6 (2018) 14764–14778.
[13] Hamed Kiani Galoogahi, Ashton Fagg, Simon Lucey, Learning background-aware correlation filters for visual tracking, in: ICCV, 2017..
[14] Joao F. Henriques, Caseiro Rui, Jorge Batista, Globally optimal solution to multi-object tracking with merged measurements, in: ICCV, 2011..
[15] Roberto Henschel, Yunzhe Zou, Bodo Rosenhahn, Multiple people tracking using body and joint detections, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2019, pp. 0–0.
[16] Hasith Karunasekera, Han Wang, Handuo Zhang, Multiple object tracking with attention to appearance, structure, motion and size, IEEE Access 7 (2019) 104423–104434.
[17] Margret Keuper, Siyu Tang, Bjoern Andres, Thomas Brox, Bernt Schiele, Motion segmentation & multiple object tracking by correlation co-clustering, IEEE Trans. Pattern Anal. Mach. Intell. 42 (1) (2018) 140–153.
[18] Hilke Kieritz, Stefan Becker, Wolfgang Hubner, Michael Arens, Online multi-person tracking using integral channel features, in: AVSS, 2016.
[19] Chanho Kim, Fuxin Li, Arridhana Ciptadi, James M. Rehg, Multiple hypothesis tracking revisited, in: ICCV, 2015.
[20] Chanho Kim, Fuxin Li, James M. Rehg, Multi-object tracking with neural gating using bilinear lstm, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 200–215.

[21] T. Kutschbach, E. Bochinski, V. Eiselein, T. Sikora, Sequential sensor fusion combining probability hypothesis density and kernelized correlation filters for multi-object tracking in video data, in: AVSS, 2017.
[22] Seong-Ho Lee, Myung-Yun Kim, Seung-Hwan Bae, Learning discriminative appearance models for online multi-object tracking with appearance discriminability measures, IEEE Access 6 (2018) 67316–67328.
[23] Evgeny Levinkov, Jonas Uhrig, Siyu Tang, Mohamed Omran, Eldar Insafutdinov, Alexander Kirillov, Carsten Rother, Thomas Brox, Bernt Schiele, Bjoern Andres, Joint graph decomposition & node labeling: problem, algorithms, applications, CVPR (2017).
[24] Yuan Li, Chang Huang, R. Nevatia, Learning to associate: hybridboosted multi-target tracker for crowded scene, CVPR (2009).
[25] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng Yang Fu, Alexander C. Berg, Ssd: Single shot multibox detector, in: ECCV, 2016..
[26] A. Milan, S. Roth, K. Schindler, Continuous energy minimization for multitarget tracking, TPAMI 36 (1) (2014) 58–72.
[27] A. Milan, K. Schindler, S. Roth, Multi-target tracking by discrete-continuous energy minimization, TPAMI 38 (10) (2016) 2054.
[28] Anton Milan, Seyed Hamid Rezatofighi, Anthony Dick, Ian Reid, Konrad Schindler, Online multi-target tracking using recurrent neural networks, in: AAAI, 2016.
[29] Hyeonseob Nam, Bohyung Han, Learning multi-domain convolutional neural networks for visual tracking, CoRR, 2015.
[30] Hamed Pirsiavash, Deva Ramanan, Charless C. Fowlkes, Globally-optimal greedy algorithms for tracking a variable number of objects, in: CVPR, 2011..
[31] Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi, You only look once: unified, real-time object detection, in: CVPR, 2016..
[32] Fabian Reichlin, Fabian Reichlin, Bastian Leibe, Esther Koller-Meier, Luc Van Gool, Online multiperson tracking-by-detection from a single, uncalibrated camera, TPAMI 33 (9) (2011) 1820–1833.
[33] Shaoqing Ren, Kaiming He, Ross Girshick, Jian Sun, Faster r-cnn: towards real-time object detection with region proposal networks, NIPS (2015).
[34] Amir Sadeghian, Alexandre Alahi, Silvio Savarese, Tracking the untrackable: Learning to track multiple cues with long-term dependencies, in: ICCV, 2017..
[35] Ricardo Sanchez-Matilla, Fabio Poiesi, Andrea Cavallaro, Online multi-target tracking with strong and weak detections, in: ECCV, 2016.
[36] Hao Sheng, Jiahui Chen, Yang Zhang, Wei Ke, Zhang Xiong, Jingyi Yu, Iterative multiple hypothesis tracking with tracklet-level association, IEEE Trans. Circ. Syst. Video Technol. 29 (12) (2018) 3660–3672.
[37] Jeany Son, Mooyeol Baek, Minsu Cho, Bohyung Han, Multi-object tracking with quadruplet convolutional neural networks, in: CVPR, 2017..
[38] Siyu Tang, Mykhaylo Andriluka, Bjoern Andres, Bernt Schiele, Multiple people tracking by lifted multicut and person re-identification, in: CVPR, 2017..
[39] Y. Xiang, A. Alahi, S. Savarese, Learning to track: online multi-object tracking by decision making, in: ICCV, 2015.
[40] Junliang Xing, Haizhou Ai, Shihong Lao, Multi-object tracking through occlusions by local tracklets filtering and global tracklets association with detection responses, in: CVPR, 2009..
[41] Jiarui Xu, Yue Cao, Zheng Zhang, Han Hu, Spatial-temporal relation networks for multi-object tracking, in: Proceedings of the IEEE International Conference on Computer Vision, 2019, pp. 3988–3998.
[42] Young-chul Yoon, Abhijeet Boragule, Young-min Song, Kwangjin Yoon, Moongu Jeon, Online multi-object tracking with historical appearance matching and scene adaptive detection filtering, in: 2018 15th IEEE International Conference on Advanced Video and Signal based Surveillance (AVSS), 2018, IEEE, pp. 1–6.
[43] Amir Roshan Zamir, Afshin Dehghan, Mubarak Shah, Gmcp-tracker: Global multi-object tracking using generalized minimum clique graphs, in: ECCV, 2012..
[44] Li Zhang, Yuan Li, Ramakant Nevatia, Global data association for multi-object tracking using network flows, in: CVPR, 2008.
[45] Ji Zhu, Hua Yang, Nian Liu, Minyoung Kim, Wenjun Zhang, Ming-Hsuan Yang, Online multi-object tracking with dual matching attention networks, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 366–382.

Zhixiong Pi received the B.S. degree in College of Electrical and Information Engineering from Hunan University in 2016. He is currently pursuing the Ph.D. degree in pattern recognition and intelligent systems from Huazhong University of Science and Technology. His research interests include visual tracking, object detection and computer vision.

Huai Qin received the B.S. degree in pattern recognition and intelligent systems from Huazhong University of Science and Technology in 2016. His research interests include computer vision, visual object tracking and surveillance video analysis.

Changxin Gao received the Ph.D. degree in pattern recognition and intelligent systems from Huazhong University of Science and Technology in 2010. He is currently an associate professor with the School of Artificial Intelligence and Automation, Huazhong University of Science and Technology. His research interests are pattern recognition and surveillance video analysis.

Nong Sang graduated from Huazhong University of Science and Technology and received his B.E. degree in computer science and engineering in 1990, M.S. degree in pattern recognition and intelligent control in 1993, and PhD degree in pattern recognition and intelligent systems in 2000. He is currently a professor at the School of Artificial Intelligence and Automation, Huazhong University of Science and Technology, Wuhan, China. His research interests include object detection and recognition, object tracking, image/video semantic segmentation, intelligent processing and analysis of surveillance videos.