Recommended Paper

# Improving the Efficiency in Multiple Object Tracking by Tracker Switching According to Occlusion States

Bo Chen[1,a) ] Muhammad Alfian Amrizal[2,b)] Satoru Izumi[3,c)] Toru Abe[1,4,d)]
Takuo Suganuma[1,4,e)]

Received: January 7, 2021, Accepted: July 7, 2021

**Abstract:** The objective of multiple object tracking (MOT) is to locate the positions of multiple objects in a video, maintain their identities, and obtain their individual trajectories. One of the most crucial challenges of MOT is how to handle object occlusion effectively. The existing methods try to treat different occlusion situations with the same architecture, which leads to limits on performance. More specifically some trackers can achieve fast speed (frame per second) but cannot handle occlusion effectively, while other trackers can handle occlusion properly but are expensive due to costly computational resources. The proposed method estimates the occlusion states of objects, applies different trackers according to the estimation results, and finally combines the results of different trackers. This method can reduce unnecessary computational costs, and consequently can improve the efficiency of the high accuracy trackers. We evaluate the effectiveness of our proposal for different scenarios.

**Keywords:** multiple object tracking, occlusion

## 1. Introduction

With the rapid development of vision technology, videos have become a popular media in many applications, and therefore video processing requirements have significantly increased. In many applications, such as visual surveillance [1], human-computer interaction [2], and virtual reality [3], it is important to understand the movement of objects in a video. For this purpose, object tracking technology has been extensively developed in the past decades. Multiple Object Tracking (MOT) is a kind of video processing technology whose objective is to locate the position of multiple objects in a video, maintain their identities and obtain their trajectories [4].

The two basic strategies of existing MOT methods are Detection-Free-Tracking and Tracking-By-Detection. In the Detection-Free-Tracking method, the beginning position of objects must be given, which requires much work and is hard to actually implement. In the Tracking-By-Detection method, a detector is used to automatically discover the objects, which makes it more practical and is gathering a lot of recent attention.

The progress of "Tracking-By-Detection" based MOT methods involves two basic components: a detector and a tracker. The task of the detector is to traverse each frame and give the position and categories (classes) of objects, meanwhile, the task of the tracker is to compare objects of different frames and find the correspondence among them by using the following information such as appearance, motion, exclusion, occlusion, etc.

The challenges of MOT include initialization and termination of tracks, similar appearance, frequent occlusion, and interactions among objects [4]. The occlusion issue might be the most critical challenge in MOT. During occlusion, all or part of an object is covered by "front" objects, resulting in much work or making it impossible to find a correspondence between objects from different frames. In this situation, the trajectory of the object might be interrupted, and the identity of the object might change (ID-switch), leading to failure of tracking.

In the past decades, many MOT methods have been proposed [5], [6], [7], [8]. Based on the ability of handling occlusion, these works could be classified into high-speed and high-accuracy types. In high-speed methods, there is no occlusion handling component and the architecture is often very simple, which makes them efficient for tracking objects without occlusion, but unreliable for occluded ones. For high-accuracy methods, the complex architecture of occlusion handling is designed, which makes them robust for occlusion, but inevitably requires more computational resources, even for objects without occlusion.

Recently, mobile platforms are becoming popular, and many computer vision applications have been applied to them [9]. Such

[1] Graduate School of Information Sciences, Tohoku University, Sendai, Miyagi 980–8579, Japan
[2] Department of Computer Sciences and Electronics, Universitas Gadjah Mada, Bulaksumur Yogyakarta 55281, Indonesia
[3] National Institute of Technology, Sendai College, Sendai, Miyagi 989–3128, Japan
[4] Cyberscience Center,Tohoku University, Sendai, Miyagi 980–8578, Japan
[a)] chenbo@ci.cc.tohoku.ac.jp
[b)] muhammad.alfian.amrizal@ugm.ac.id
[c)] izumi@sendai-nct.ac.jp
[d)] beto@tohoku.ac.jp
[e)] suganuma@tohoku.ac.jp

The preliminary version of this paper was published at IPSJ SIGDPS Technical Reports, December 2019. The paper was recommended to be submitted to Journal of Information Processing (JIP) by the chief examiner of SIGDPS.

platforms have strictly limited computational resources, and thus computer vision applications must be developed so that they can run very efficiently on them. For this reason, the efficiency of computer vision methods has attracted more attention. In this research, we try to explore a way to achieve a balance between the accuracy and speed of MOT methods. Due to the complexity of objects (millions of categories), to simplify the research, we focus on pedestrian tracking.

The basic strategy of the proposed method is to adopt the most suitable trackers depending on the situation. We classified the states of pedestrians into non-occluded and occluded states. For each state, we assign the most suitable tracker to it. We check the change of states and deploy corresponding trackers to suit the new occlusion state. We evaluated the efficiency of existing trackers for different occlusion states. To demonstrate the effectiveness of the proposed method, we tested it on different scenarios, from less-occluded to highly-occluded ones.

The contributions of our research are:

- a method for estimating the occlusion states of objects is proposed, and
- a method for switching trackers according to object occlusion states is proposed.

We can reduce unnecessary computational costs in MOT and then improve its efficiency by separating the occlusions of objects into different states and applying different trackers according to the occlusion states. Furthermore, we discuss the proper trackers that can be used in our proposal and evaluate the performance of our proposal with them.

## 2. Related Works

### 2.1 Occlusion in MOT

Occlusions might represent the most crucial challenge in MOT. There are several different types of occlusions for pedestrians in MOT [10].

Pedestrians may be occluded by obstacles including buildings, vehicles, and vegetation, etc. In this situation, pedestrians may be completely covered by obstacles, which are generally bigger than the size of pedestrians, and these pedestrians are invisible to the tracker. When a pedestrian completely disappears, the identity of the pedestrian usually cannot be kept for a short period until the pedestrian reappears. If it disappeared for a very long period, the general operation is to stop tracking for this particular pedestrian.

The second possibility of occlusion is that pedestrians are occluded with each other, or inter-occlusion of pedestrians. In this situation, if there are relatively few pedestrians, pedestrians covered by others would reappear rapidly, and the tracker could find correspondence of occluded pedestrians by visible parts. On the other hand, if the scenario is highly crowded, it is very common that most pedestrians are occluded heavily, which makes it the most challenging situation.

The third situation is “self-occlusion” or “intra-occlusion”. When a pedestrian is walking, some parts of the body would appear and disappear regularly. The detector and tracker must therefore have the ability to ignore this interference and maintain robust detection and tracking.

In this research, for simplicity, we mainly focus on the second type of occlustion or mainly inter-occlusion of pedestrians since it is the hardest occlusion scenario to deal with. If the detector and tracker could obtain adequate performance under this scenario, then it would also maintain its effectiveness in other scenarios.

### 2.2 Tracker of MOT

Existing trackers can be categorized into online and offline types. The online type only utilizes information from the previous and current frames, and the tracking result is unchangeable for previous frames. Otherwise, the offline tracker relies on past and future frame information. In general, the offline tracker is more stable than the online tracker.

Based on their occlusion handling capability, trackers can be categorized into high-speed trackers and high-accuracy trackers.

High-speed trackers generally don’t handle occlusion issues well especially in the case of heavily occluded objects. In some high-speed trackers, there is no occlusion handling approach, such as IOU tracker [7], which only utilized Intersection over Union (IoU) of bounding boxes as the affinity between objects. Some other trackers do have an occlusion handling approach like SORT tracker [8] that utilizes a Kalman filter [11] for motion prediction but is still inadequate for handling occlusion issues. The reason that these trackers can handle slightly occluded objects is that in the tracking-by-detection approach, the slightly occluded objects still have a high detection confidence score and can therefore be treated as objects without occlusions so these trackers can handle light occlusions even without a sophisticated occlusion approach. But if the occlusion is too heavy, the detection confidence score will drop significantly and these trackers cannot handle this situtation. Because of their simple architecture, these trackers generally run faster than high-accuracy trackers.

High accuracy trackers generally have a complex architecture for handling occlusion issues. Recently, various types of trackers utilizing deep neural networks have been proposed. Some trackers only apply detectors based on deep neural networks, and then accomplish tracking based on these detection results. Another way is to exchange the handcrafted tracking methods for deep ones. In Deep SORT [5], they try to find occluded objects by deep appearance features. With the help of visual information, they can do re-identification for reappearing pedestrians and can therefore acquire more robust results than a SORT tracker. In Ref. [12], a LSTM [13] net is trained to handle long-term corresponding objects. Finally, there are trackers integrating deep detection and deep-tracking utilizing an end-to-end network for the entire framework such as the Tracktor tracker [14]. These trackers have complex architectures, as well as utilizing deep networks, and require more computational resources compare to high-speed trackers. These trackers can handle heavy occlusion situations better than high-speed trackers or namely can achieve higher accuracy but if the occlusion is too large causing the target to be nearly invisible then it is hard to find affinity for these heavily occluded objects. Recovery of the occluded objects is therefore necessary when these objects reappear after occlusion.

### 2.3 Strategies of Occlusion Handling

Strategis of currently existing methods can be categoried into

three types.

The first strategy is "Part-to-whole" and is based on the fact that during occusion some parts of the objects are still visible. The tracker can therefore find correspondence by utilizing these parts. In Ref. [15], they divide a holistic object into several parts, and the consistency is obtained by integrating the similarity between corresponding parts. If a part is occluded, the affinity among the occluded part and non-occluded part is lower than usual, so the tracker ignores it and only integrates the visible parts. In Ref. [16], which is a particular method for humans, they adopted the Deformable Part Models (DPM) detector, which used Histograms of Oriented Gradients (HOG) feature to generate a model for different human body parts which are deformable and then create HOG feature map for the input image and match the human model and the feature map to find pedestrians. In modern deep learning based methods such as Deep SORT [5], Tracktor [14], a re-identification network is trained to find affinity between images of the same target which are suitable for finding affinity between occluded and non-occluded objects. This method can therefore be utilized to recover objects after occlusion.

Another strategy is "Hypothesize-and-test." It treats the occlusion issue in MOT as a case of application of statistics. In Ref. [17], the occlusion hypothesis is generated based on an occluded pair of observations, where the characteristics of them are close and on a similar scale. The occlusion is a distraction in the hypothesis. In the test process, the hypothesis of observation and original states are input to a cost-flow framework, and MAP is evaluated to obtain an optimal solution. In Refs. [18] and [19], occlusion patterns are used to assist the detector, where different hypotheses are generated from the synthesis of two objects with different levels and patterns, and then the detector is trained on these hypotheses.

The third strategy is "Buffer-and-recover." This strategy buffers the observations and states of the object before occlusion and recovers the states after occlusion based on buffered observations and states. In Ref. [20], when an occlusion happens, the trajectory is maintained during up to 15 frames, the possible trajectory predicted if the object reappears, and the predicted and possible trajectories are linked and the identity maintained. In Ref. [21], during occlusion the observation mode is activated until enough observations are obtained and a hypothesis generated to explain the observation. Modern deep learning based methods such as Deep SORT [5], Tracktor [14] also utilized the same strategy to suspend the tracking if they can't find the target, and for objects that satisfy particular requirements, the re-identification is applied to recover the original object.

## 3. Proposal

### 3.1 Assumption of Proposal

As previously mentioned, the research objective of our work is pedestrian tracking. For this purpose, the assumption of this research is given as described here. Each frame of the video involves multiple pedestrians, only inter-occlusion of pedestrians will be considered. The proposed tracker will then be applied to these pedestrian candidates and outputs the trajectories of them.

### 3.2 Overview

The basic strategy of our proposal is the combination of the merits of two types of trackers: A high-speed tracker solves non-occluded tracking, and a high-accuracy tracker handles occluded tracking. In this way, we reduce redundant computational costs for non-occluded pedestrians and also improve the efficiency of the entire tracking progress.

**Figure 1** shows an overview of the proposed method. For input video, firstly a pedestrian detector is applied on each frame, to give positions of pedestrians. Then we check the relationship between pedestrians, label the occlusion states (Non-occluded/Occluded). Then we assign the most suitable tracker for pedestrians. We also trace any variations in occlusion states and promptly exchange the tracker if the occlusion state changes. Finally, the tracklets (i.e., short trajectories of tracked objects) of both trackers are merged to obtain the global trajectory of pedestrians.

### 3.3 Pedestrian Detection

A pedestrian detector is utilized for each video frame. The out-

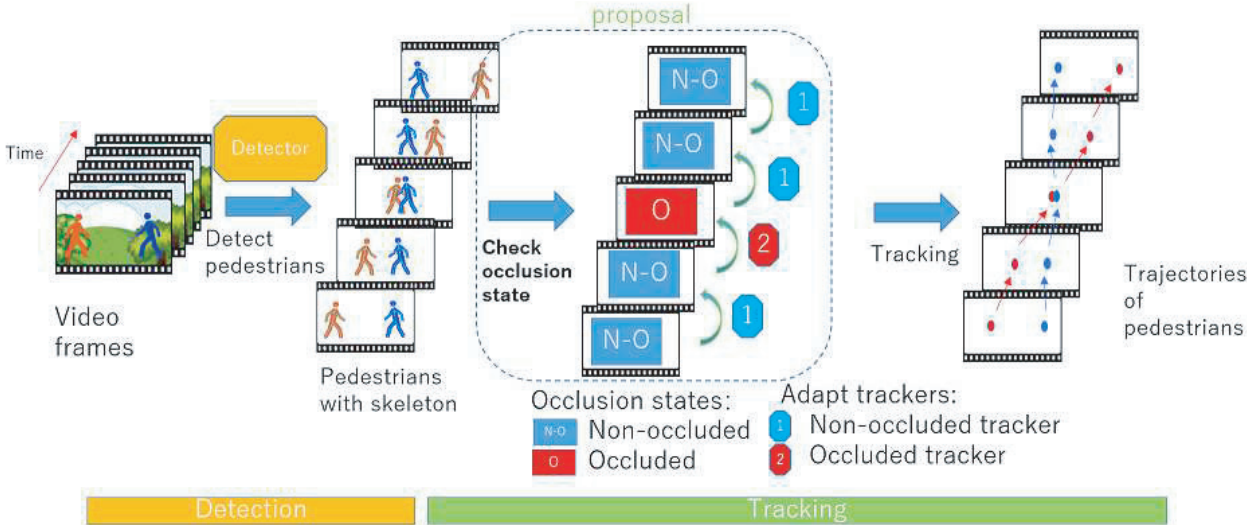


**Fig. 1** Overview of the proposed method.


**Fig. 2** Different outputs from pedestrian detectors. (a) bounding box (b) skeleton (c) mask.

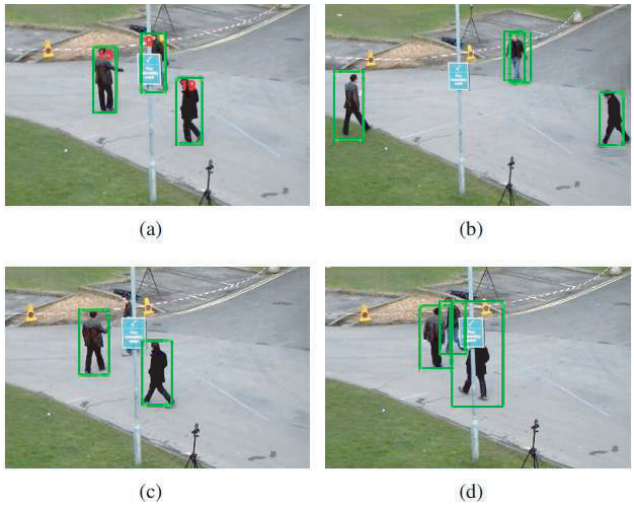


**Fig. 3** Ground truth and different detection issues: (a) ground truth: exact bounding boxes for all pedestrians (b) detection issue: multiple bounding boxes for middle pedestrian (c) detection issue: cannot detect middle occluded pedestrian (d) detection issue: larger bounding box for right pedestrian.

put from pedestrian detectors will differ depending on how pedestrians are expressed. **Figure 2** shows examples of different types of detectors. Some detectors utilized a bounding box to show the pedestrian region, which is a rectangle around the pedestrian. Another way is a skeleton, which is several line segments that connect joints of different body parts. A more precise way is a mask, which only involves pixels that belong to the body. Our proposal is based on occlusion detection for the pedestrians, so the detector must can detect the partially occluded pedestrians, and the output must be bounding box.

The ideal result of detection should be accurate: each bounding box cover just one pedestrian and even if the pedestrian is occluded, the detector can still search for it by the visible parts, estimate the occluded parts and provide proper coordinates for all four vertices of the bounding box as a ground truth as shown in **Fig. 3** (a) shows. The result of modern detectors may involve many mistakes such as multiple detection results for one pedestrian in Fig. 3 (b), pedestrians missed due to occlusion in Fig. 3 (c), and unsuitable bounding box sizes in Fig. 3 (d). These issues should be properly handled in the detection processing. We filter out unreliable detection results by setting detection threshold $T_d$ but pedestrians with higher confidence than $T_d$ will be considered.

### 3.4 Check Occlusion States

**Figure 4** shows different occlusion situations of objects and

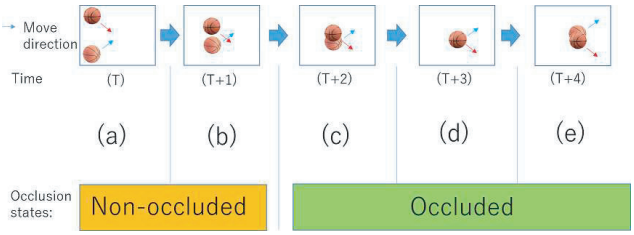


**Fig. 4** Different situations of objects interaction and our definition of occlusion states: (a) Non-occluded (b) Non-occluded (c) Partial-occluded (d) Full-occluded (e) Partial-occluded. (c) (d) (e) are "Occluded" in our definition.

our definition of occlusion states. In Fig. 4 (a)(b), two balls move close to each other, without overlapping and we classify this case as a "Non-occluded" state. In Fig. 4 (c), (d), (e), the "front" ball overlapped the "back" ball partially or completely. Rregardless of how much they overlap and what position the object is located, we classify these cases to "Occluded" state.

With this definition, we can check the occlusion states of pedestrians by the relationship of the bounding box for different detection or namely for bounding box based detection, if the bounding boxes of two pedestrians are crossed, they are occluded with each other, otherwise, they are non-occluded. For skeleton detection, we can create a bounding box from the skeleton and then apply the IoU based method. For mask detection, we can also generate a bounding box from the range of the mask, and then use IoU to check occlusion states. Different types of detectors have different occlusion detection methods, if we can check the occlusion state more accurately, the accuracy of the proposal will be better. However, more sophisticated occlusion detection method can check the occlusion state accurately, its checking speed is slow and then it will last slower speed of proposal.

We utilize Intersection over Union (IoU) to express the quantitative relationship between bounding boxes. Given two bounding boxes A and B, the IoU of them could be expressed as the following equation:

$$IoU(A, B) = \frac{A \cap B}{A \cup B} \tag{1}$$

The range of IoU is [0,1]. If two bounding boxes have no contact, the IoU is 0. If two bounding boxes completely overlap, the IoU is 1.

If there are more than two pedestrians in the frame image, the IoU is calculated for each pair of them, therefore the calculation is the same as when there are two pedestrians. If a pedestrian is occluded by more than two pedestrians, the IoU of the pedestrian is the ratio between the union of occluded parts of the pedestrian's bounding box and the entire area of the pedestrian's bounding box.

The detection issues should be handled appropriately. We try to solve the "multiple detection" issue by setting the threshold $T_o$ of IoU between two bounding boxes in the same frame. The threshold is set to 0 for ground truth, and the threshold should be set to higher than 0 for unstable detection, but making it too large will lower the occlusion detection capability.

Our target is to balance between accuracy and speed of trackers, the speed of occlusion detection should be fast enough to reduce extra computational costs in addition to the trackers.

### 3.5 Switching Trackers

There are two trackers in our proposal, a high speed tracker and a high accuracy tracker. As previously mentioned, the high speed tracker is faster but with a lower ability to handle occlusions, and the high accuracy tracker is slower but with a higher ability to handle occlusions. The switching of the two trackers must be efficient to achieve our objective.

In the video frames $T$ and $T+1$, we first check the occlusion states of pedestrians that have higher detection confidence scores than $T_d$, and then record the state of each pedestrian.

The high-speed tracker is utilized if the occlusion states of corresponding pedestrians from the two frames are both "occluded", the high-accuracy tracker is utilized if they are both "occluded" or the occlusion states change from "non-occluded" to "occluded", otherwise the high-speed tracker is utilized.

The matched detection results in frame $T+1$ will be removed, resulting in unmatched detection results. These detection results can be matched with previously labeled "missed" pedestrians by the re-identification approach. Finally, the remaining detection results are treated as new pedestrians and we create new tracklets for them. If the "missed" pedestrians match some conditions, such as excessively long time after suspended tracking, we will stop the tracking.

### 3.6 Combination of Trajectories

By switching of trackers for different situations, the tracklets of pedestrians from different trackers finally compose the complete trajectories of all objects.

## 4. Experiment

### 4.1 Evaluation of Occlusion Detection Method

To evaluate the effectiveness of our occlusion detection method in Section 3.4, we tested it on the public benchmark dataset.

We choose the MOT Challenge 2015 [22] training set as our test dataset, which is a widely used benchmark for MOT. It contains images of different scenarios, with different occlusion intensities. This dataset includes pedestrian detection results and their ground truth (positions and identities) as bounding boxes. The pedestrian detection results are obtained by the Aggregated Channel Features (ACF) detector. In the experiments, tracking methods refer to those pedestrian detection results in the dataset instead of detecting every pedestrian in each frame image, and consequently only tracking processes are evaluated. Detailed data is shown in the **Table 1**.

MOTA is a widely used metric in the evaluation of MOT methods, and the definition of MOTA is:

$$MOTA = 1 - \frac{\sum_t (FN_t + FP_t + IDSW_t)}{\sum_t GT_t} \quad (2)$$

Where $t$ is frame index, FN is false negative, FP is false positive, IDSW is ID switch, and GT is ground truth objects. Generally, the MOTA is expressed as a percentage ranging (-∞,100].

To evaluate the occlusion intensities of these sequences, we define the occlusion ratio $Or$ of each sequence $s$ as:

$$Or(s) = \frac{amount\ of\ occluded\ bbox}{total\ amount\ of\ bbox} \quad (3)$$

The $Or$ of each sequence is shown in the last column of Table 1.

We applied our occlusion detection method on these sequences, with different thresholds of IoU, which are used to control whether the two bounding boxes will be treated as the same person or not. **Figure 5** shows the result. It shows a comparison of results with threshold (from 0.1 to 0.9) and without threshold (None).

**Table 1** Detail of MOT2015 dataset.

| Training sequences | | | | | |
|---|---|---|---|---|---|
| Name | Resolution | Frames | Trajectories | Boxes | Or |
| PETS09-S2L1 | 768 × 576 | 795 | 19 | 4,476 | 0.19 |
| KITTI-13 | 1,242 × 375 | 340 | 42 | 762 | 0.37 |
| TUD-Stadtmitte | 640 × 480 | 179 | 10 | 1,156 | 0.50 |
| ETH-Bahnhof | 640 × 480 | 1,000 | 171 | 5,415 | 0.54 |
| ADL-Rundle-8 | 1,920 × 1,080 | 654 | 28 | 6,783 | 0.54 |
| KITTI-17 | 1,242 × 370 | 145 | 9 | 683 | 0.56 |
| ETH-Sunnyday | 640 × 480 | 354 | 30 | 1,858 | 0.63 |
| TUD-Campus | 640 × 480 | 71 | 8 | 359 | 0.72 |
| Venice-2 | 1,920 × 1,080 | 600 | 26 | 7,141 | 0.82 |
| ADL-Rundle-6 | 1,920 × 1,080 | 525 | 24 | 5,009 | 0.82 |

### 4.2 Evaluation of Tracker Switch Strategy

To evaluate the efficiency of our proposal of tracker switch, we tested the strategy on existing trackers.

The setup for our experiment is shown in **Table 2**. Details of the data are the same as the previous experiment.

In our assumption, the high-speed tracker should be as fast as possible, and the high-accuracy should be extremely stable for occlusion. Unfortunately, state-of-the-art trackers are unsatisfactory and implementing them requires a vast amount of work. Therefore we checked existing open-sourced trackers and selected suitable ones for our experiment.

The two trackers we used in the experiment are handcrafted. The high-speed tracker is an iou-tracker [7]. The IOU tracker is assume to have high fps. When the object's movement between adjacent frames is very small, the tracker just searches the best match. i.e., the highest IoU of bounding boxes from the adjacent frames, then the best match is updated to tracklet. If the length of

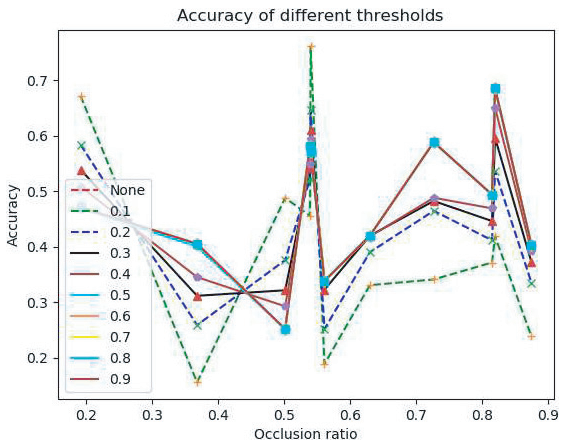


**Fig. 5** Result of of different occlusion detection thresholds.

**Table 2** Experiment setup.

| | |
|---|---|
| CPU | Intel Core i7-8700 |
| Memory | 16 GB |
| Occlusion-weak tracker | iou-tracker [7] |
| Occlusion-robust tracker | SORT-tracker [5] |

**Table 3** Features of two types trackers.

| Types | FPS | MOTA |
|---|---|---|
| High-speed | higher | lower |
| High-accuracy | lower | higher |

**Table 4** Validation of two trackers.

| | MOTA | fps |
|---|---|---|
| iou-tracker | 58.3% | 12,083 |
| SORT-tracker | 67.0% | 380.3 |

a tracklet is larger than the threshold, then the tracklet is performing successful tracking. If a tracklet cannot find a match with the current detection, it is terminated. If detection cannot be matched with the previous tracklet, the tracker will assign a new tracklet for it.

The high-accuracy tracker is a more complex tracker called SORT-tracker [8]. We should point out that if we compare our algorithm to the state-of-the-art trackers, our algorithm is not the ideal choice since it does not have a very good occlusion handling algorithm. We choose this tracker due to its similar strategy of tracking with the iou-tracker. It is also based on the IoU between the previous frame and current frames' objects, and chooses objects with a larger IoU than the pre-defined threshold.

The difference between iou-tracker and SORT-tracker is the latter has a prediction part, which is achieved by Kalman filter [11]. It predicts the future position future position of bounding box based on past frames, and then associates previously predicted tracklets and current detection by linear assignment of IoU. It is more complex than the iou-tracker, which makes it much slower. The iou tracker and SORT tracker both use IoU to calculate the similarity of bounding boxes, and the only difference is the Kalman filter in SORT, so the switch between the two trackers is equivalent to whether or not a Kalman filter is used. The iou tracker is treated as the baseline. If the bounding boxes are occluded, the Kalman filter will be applied to predict the future position, and the predicted position is used to match the new detection. In this way, we can switch the trackers efficiently, thereby achieving a balance between accuracy and speed. The iou tracker and SORT tracker both use IoU to calculate the similarity of bounding boxes, and the only difference is that the Kalman filter is used in SORT. Thus the only difference between the two trackers is basically whether they use the Kalman filter or not. The iou tracker is treated as the baseline, and if the bounding boxes are occluded, the Kalman filter will be applied to predict the future position, and the predicted position is used to match the new detection.

The features of two types trackers are shown in **Table 3**.

We validated the two trackers on the MOT dataset 2015, and **Table 4** shows the result.

From accuracy and speed results we can know whether the proposed method can achieve a balance between the iou tracker and the SORT tracker, as we expected. The accuracy is not as high as the SORT tracker because the iou tracker is still used for simpler situations and without prediction of motion it may not be able to find the precise future position.

From the result of the evaluation, we confirmed the iou-tracker has very high performance on bounding box based tracking (over

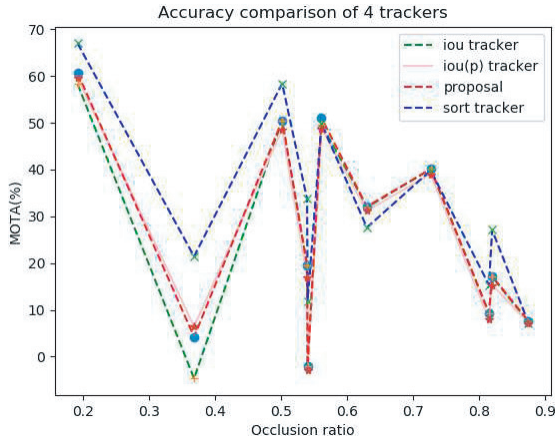


**Fig. 6** Result of accuracy comparison of 4 methods.

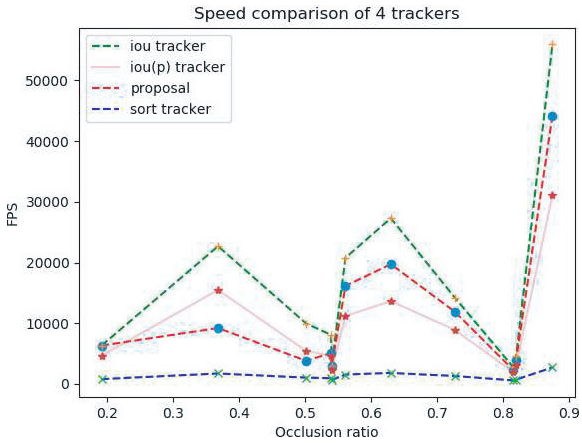


**Fig. 7** Result of speed comparison of 4 methods.

10,000 fps), but with lower MOTA. On the other hand, the SORT-tracker has a relatively slower speed (less than 1,000 fps) but has a higher MOTA. These attributes of the two trackers are what we expected.

Our strategy is to switch between different types of trackers. Due to the difficulty of combining the iou-tracker and SORT-tracker (they work in very different ways), we modified the iou-tracker to add the more robust feature such as used in SORT-tracker, or namely we use the uniform motion model to predict the future position of a pedestrian, which is based on the average speed of past frames. This method is simpler than the Kalman filter method, which gives us more possibilities for evaluating the effect of different tracker speeds. We name this tracker the "iou(p)-tracker". We combined the iou-tracker and iou(p)-tracker with our strategy, which switches different trackers according to the occlusion states of pedestrians. If it is non-occluded, we apply the iou-tracker, if it is occluded, we apply the iou(p)-tracker. We name this method "proposal." We test all the above trackers (SORT-tracker, iou-tracker, iou(p)-tracker, proposal) on dataset MOT 2015. As the experimental results, the accuracy (MOTA) is shown in **Fig. 6**, the calculation is shown in **Fig. 7**.

### 4.3 Result and Discussion

From the result of occlusion detection with different thresholds Fig. 5, we noticed that threshold 0.4 could achieve the best accuracy in highly-occluded sequences (0.6–0.9), and the result

**Table 5** Relative accuracy and speed of trackers.

| tracker | Relative accuracy | Relative speed |
|---|---|---|
| SORT | 1 | 1 |
| iou(p) | 0.829 | 95.5 |
| proposal | 0.82 | 116 |
| iou | 0.793 | 171 |

is very unstable. This result shows the proposed method needs improvement to handle unstable pedestrian detection results.

From the accuracy result Fig. 6, the most complex SORT-tracker could achieve the highest accuracy in most scenarios, and the simplest iou-tracker is the worst, and the iou(p)-tracker could achieve slightly higher accuracy compared to our proposal for lower occlusion ratio scenarios. In high occlusion ratio situation, all trackers obtained similar results, and the accuracy significantly decreased. This result shows these trackers cannot deal with highly occluded situations, but for low occlusion ratio scenarios, our proposed tracker could achieve accuracy similar to the iou(p)-tracker which demonstrates the effectiveness of this strategy. The reason is that the uniform motion model of iou(p) tracker is suitable to datasets where most pedestrian movements are relatively simple.

Figure 7 shows the speed comparison. For all scenarios, the simplest iou-tracker is the fastest tracker, and the most complex SORT-tracker is the slowest. The modified iou(p)-tracker and proposal rank intermediately. For lower occlusion ratio scenarios (0.2–0.5), due to the occlusion detection cost, the iou(p)-tracker is more efficient. However, in the higher occlusion ratio situation (0.6–0.9), our proposal reduces the computational cost significantly. This result indicates that the occlusion detection method must be efficient enough to offset the extra cost itself, thus the proposal could achieve the expected performance and effectiveness.

From the results, we now know that iou(p) accuracy is similar but neither of them is as good as SORT. The speed of our proposal is faster than iou(p), they both much faster than the SORT tracker.

To evaluate the efficiency of the trackers, we calculated the relative accuracy and relative speed of trackers. The results are shown in **Table 5**. From the results, we know that the accuracy of iou(p) is similar to the proposal, but neither of them is as good as the SORT tracker. We also know that the speed of the proposal is faster than iou(p), these are both much faster that the SORT tracker.

The two results demonstrate that it is possible to improve the efficiency of trackers by combining fast-weak and slow-robust trackers. The current experiments tested only simple trackers, and we believe our strategy can be more effective on more complex trackers.

## 5. Conclusion and Future Work

We propose a simple strategy to improve the efficiency of existing tracker methods. We combine different trackers, classify the objects by different occlusion states, and apply the most suitable tracker to them. We test our proposal on existing datasets, and evaluate it under different situations.

In the future, we will try to utilize more stable features in our tracker, such as the intensity of illumination, optical flow, etc. In the current stage we tested handcrafted methods, and the future work will involve deep learning algorithms. We will test this strategy on more complex trackers to evaluate the effectiveness.

### References


[1] Wang, X.: Intelligent multi-camera video surveillance: A review, *Pattern Recognit. Lett.*, Vol.34, No.1, pp.3–19 (2013).

[2] Candamo, J., Shreve, M., Goldgof, D.B., Sapper, D.B. and Kasturi, R.: Understanding transit scenes: A survey on human behavior-recognition algorithms, *IEEE Trans. Intell. Transp. Syst.*, Vol.11, No.1, pp.206–224 (2010).

[3] Uchiyama, H. and Marchand, E.: Object detection and pose tracking for augmented reality: Recent approaches, *18th Korea-Japan Joint Workshop Front. Comput. Vision* (2012).

[4] Luo, W., Xing, J., Milan, A., Zhang, X., Liu, W. and Kim, T.-K.: Multiple object tracking: A literature review, *Artif. Intell.*, Vol.293, pp.1–23 (2021).

[5] Wojke, N., Bewley, A. and Paulus, D.: Simple online and realtime tracking with a deep association metric, *IEEE Int. Conf. Image Process.*, pp.3645–3649 (2017).

[6] Sadeghian, A., Alahi, A. and Savarese, S.: Tracking the untrackable: Learning to track multiple cues with long-term dependencies, *IEEE Int. Conf. Comput. Vision*, pp.300–311 (2017).

[7] Bochinski, E., Eiselein, V. and Sikora, T.: High-speed tracking-by-detection without using image information, *14th IEEE Int. Conf. Adv. Video Signal Based Surv.*, pp.1–6 (2017).

[8] Bewley, A., Ge, Z., Ott, L., Ramos, F. and Upcroft, B.: Simple online and realtime tracking, *IEEE Int. Conf. Image Process.*, pp.3464–3468 (2016).

[9] Tan, M., Chen, B., Pang, R., Vasudevan, V., Sandler, M., Howard, A. and Le, Q.V.: MnasNet: Platform-aware neural architecture search for mobile, *IEEE/CVF Conf. Comput. Vision Pattern Recognit.*, pp.2820–2828 (2019).

[10] Meshgi, K. and Ishii, S.: The state-of-the-art in handling occlusions for visual object tracking, *IEICE Trans. Inf. Syst.*, Vol.E98-D, No.7, pp.1260–1274 (2015).

[11] Reid, D.B.: An algorithm for tracking multiple targets, *IEEE Trans. Autom. Control*, Vol.24, No.6, pp.843–854 (1979).

[12] Girdhar, R., Gkioxari, G., Torresani, L., Paluri, M. and Tran, D.: Detect-and-track: Efficient pose estimation in videos, *IEEE/CVF Conf. Comput. Vision Pattern Recognit.*, pp.350–359 (2018).

[13] Hochreiter, S. and Schmidhuber, J.: Long short-term memory, *Neural Comput.*, Vol.9, No.8, pp.1735–1780 (1997).

[14] Bergmann, P., Meinhardt, T. and Leal-Taixé, L.: Tracking without bells and whistles, *IEEE/CVF Int. Conf. Comput. Vision*, pp.941–951 (2019).

[15] Hu, W., Li, X., Luo, W., Zhang, X., Maybank, S. and Zhang, Z.: Single and multiple object tracking using log-Euclidean Riemannian subspace and block-division appearance model, *IEEE Trans. Pattern Anal. Mach. Intell.*, Vol.34, No.12, pp.2420–2440 (2012).

[16] Izadinia, H., Saleemi, I., Li, W. and Shah, M.: $(MP)^2T$: Multiple people multiple parts tracker, *Eur. Conf. Comput. Vision*, pp.100–114 (2012).

[17] Zhang, L., Li, Y. and Nevatia, R.: Global data association for multi-object tracking using network flows, *IEEE Conf. Comput. Vision Pattern Recognit.*, pp.1–8 (2008).

[18] Tang, S., Andriluka, M. and Schiele, B.: Detection and tracking of occluded people, *Int. J. Comput. Vision*, Vol.110, No.1, pp.58–69 (2014).

[19] Tang, S., Andriluka, M., Milan, A., Schindler, K., Roth, S. and Schiele, B.: Learning people detectors for tracking in crowded scenes, *IEEE Int. Conf. Comput. Vision*, pp.1049–1056 (2013).

[20] Mitzel, D., Horbert, E., Ess, A. and Leibe, B.: Multi-person tracking with sparse detection and continuous segmentation, *Eur. Conf. Comput. Vision*, pp.397–410 (2010).

[21] Ryoo, M.S. and Aggarwal, J.K.: Observe-and-explain: A new approach for multiple hypotheses tracking of humans and objects, *IEEE Conf. Comput. Vision Pattern Recognit.*, pp.1–8 (2008).

[22] Leal-Taixé, L., Milan, A., Reid, I., Roth, S. and Schindler, K.: MOTChallenge 2015: Towards a benchmark for multi-target tracking, arXiv preprint arXiv:1504.01942 (2015).


### Editor's Recommendation

This paper proposed a method to obtain the trajectories by multi object tracking (MOT) in a video. A crucial challenge is on how to efficiently handle the occlusion, the paper reduced its

effect of occlusion by two different strategies selected by whether the target is hidden or not. The evaluations showed publicly available datasets under various congestion conditions and compare existing studies to show the usefulness of the proposed method. This paper is recommended because it is expected to be applied in more realistic environments and practical efforts are expected in the future.

(Chief examiner of SIGDPS Atsushi Tagami)

**Bo Chen** received his B.S. degree from University of Science and Technology Beijing in 2011 and his M.S. degree from Tohoku University in 2016. He is currently pursuing the Ph.D. degree with Tohoku University. His research interest is computer vision and deep learning.

**Muhammad Alfian Amrizal** received his B.E. degree in mechanical engineering and his M.S. and Ph.D. degrees in information sciences from Tohoku University, in 2012, 2014, and 2017, respectively. He is currently a Lecturer with Department of Computer Sciences and Electronics, Universitas Gadjah Mada. His research interests are in the area of high-performance computing (HPC), including the dependability of HPC systems, novel fault tolerance techniques, performance modeling, and optimization.

**Satoru Izumi** received his M.S. and Ph.D. degrees from Tohoku University, Japan, in 2009 and 2012, respectively. He is currently an Associate Professor with National Institute of Technology, Sendai College. He received the IEEE GCCE 2013 Excellent Paper Award. His main research interests include semantic Web, ontology engineering, green ICT, and software defined networking. He is a member of the IEICE and the IPSJ.

**Toru Abe** received his M.E. and Ph.D. degrees in information engineering from Tohoku University, in 1987 and 1990, respectively. From 1990 to 1993, he was a Research Associate with the Education Center for Information Processing, Tohoku University. From 1993 to 2001, he was an Associate Professor with the Graduate School of Information Science, Japan Advanced Institute of Science and Technology. He is currently an Associate Professor with the Cyberscience Center, Tohoku University. His research interests include image processing and knowledge engineering.

**Takuo Suganuma** received his M.S. and Ph.D. degrees in engineering from the Chiba Institute of Technology, Japan, in 1994 and 1997, respectively. He is currently a Professor and the Director with the Cyberscience Center, Tohoku University, Japan. His main research interests include agent-based computing, flexible network middleware, network management, symbiotic computing, green ICT, and Disaster-resistant communications. He is a member of the IEICE and the IPSJ. He received the UIC-07 Outstanding Paper Award in 2007.