VariabilityTrack: Multi-Object Tracking with Variable Speed Object Movement

Run Luo\(^1\), JinLin Wei\(^1\), Qiao Lin\(^1\)
Huazhong University of Science and Technology

Abstract

Multi-object tracking (MOT) aims at estimating bounding boxes and identities of objects in videos. Most methods can be roughly classified as tracking-by-detection and joint-detection-association paradigms. Although the latter has elicited more attention and demonstrates comparable performance relative than the former, we claim that the tracking-by-detection paradigm is still the optimal solution in terms of tracking accuracy, such as ByteTrack\(^{[70]}\), which achieves 80.3 MOTA, 77.3 IDF1 and 63.1 HOTA on the test set of MOT17 with 30 FPS running speed on a single V100 GPU. However, under complex perspectives such as vehicle and UAV acceleration, the performance of such a tracker using uniform Kalman filter will be greatly affected, resulting in tracking loss. In this paper, we propose a variable speed Kalman filter algorithm based on environmental feedback and improve the matching process, which can greatly improve the tracking effect in complex variable speed scenes while maintaining high tracking accuracy in relatively static scenes. Eventually, higher MOTA and IDF1 results can be achieved on MOT17 test set than ByteTrack\(^{[70]}\).

1. Introduction

Tracking by detection is the most effective paradigm for multi-object tracking (MOT) in current, such as ByteTrack\(^{[70]}\), which achieves 80.3 MOTA, 77.3 IDF1 and 63.1 HOTA on the test set of MOT17 with 30 FPS running speed on a single V100 GPU. That’s amazing! ByteTrack\(^{[70]}\) uses a simple and effective strategy — assuming that objects are moving at a uniform speed and matching the low score detection boxes to find the obscured objects. When the Angle of video is static or compare the stability, uniform motion hypothesis can be simple and accurate prediction of motion of the object, but when the video from the point of view of the change is more intense, such as the video taken during driving or unmanned aerial vehicle (UAV) movement, the object will have a sudden violent movement, constant speed in a simple hypothesis cannot accurately predict object trajectory. At the same time, a simple IOU matching strategy will result in a large number of object tracking failures.

Figure 2 (a) and (b) show this problem. In frame \(t - 2\), we have four different tracklets as two tracklets near and two tracklets far away. However, from frame \(t - 1\) to frame \(t + 2\) when close objects and perspectives are close to each other, the speed changes dramatically, the yellow tracklet at the front fails to match the target track because the IOU between the predicted box of uniform Kalman filter and the real detection box are too small. Meanwhile, the yellow tracklet’s box will remain several frames. The blue tracklet will suffer the same influence in frame \(t + 2\) and mismatch the remaining yellow tracklet’s box, resulting in ID switching. In this cycle, a large number of lost boxes will remain on both sides of the Angle of view, causing a chain reaction e.g., the result can be seen in Figure 2 (b). To the best of our knowledge, very few SOTA methods use variable Kalman filter, including ByteTrack\(^{[70]}\), to solve this problem.

In this paper, we propose a variable speed Kalman filter algorithm based on the change of motion perspective and process improvement to increase the cost of similar tracklet to reduce the matching times of prediction box and detection box, so as to solve the problem that uniform kalman filter in variable speed perspective cannot solve. As shown in Figure 2 (c), When the speed of the tracklet changes dramatically due to the perspective position in the front, it will produce a very large acceleration. In this case, the accelerated Kalman filter will be used to solve this problem well, while the acceleration of the target with the perspective position in the back is very small, and the uniform Kalman filter can be used to accurately predict the tracklet. Therefore, when the target acceleration is large, we use the accelerated Kalman filter, and when the target acceleration is small, we use the uniform Kalman filter.

All of our work is an improvement on ByteTrack\(^{[70]}\). If you want to evaluate the generalization ability of our proposed method, you can apply it to 9 different state-of-the-art trackers like ByteTrack\(^{[70]}\), including the Re-ID-based ones.
Figure 1. MOTA-IDF1-FPS comparisons of different trackers. The horizontal axis is FPS (running speed), the vertical axis is MOTA, and the radius of circle is IDF1. For the perspective of complex motion, variable speed Kalman filter is used to improve the tracking effect and our achieve better MOTA performance than ByteTrack[70], while reducing the number of matches and achieving better IDF1. However, it increases the amount of computation, so FPS decreases. Details are given in Table 1.

[58, 72, 27, 38], motion-based ones [75, 60, 39], chain-based one [39] and attention-based ones [48, 66]. Compared to ByteTrack[70], we achieve improvements on almost all the metrics including MOTA, IDF1 score and ID switches. We increase the MOTA of ByteTrack [75] from 80.3 to 80.4, IDF1 from 77.3 to 77.5 and decrease the IDs from 2196 to 2175 on the test set of MOT17 [75].

Towards pushing forwards the state-of-the-art performance of MOT, we propose a simple and strong tracker, named VariabilityTrack, which use variable Kalman filter based on the ByteTrack[70].

Our proposed method is the first work that based on the excellent performance of ByteTrack[70] and simply uses accelerated Kalman filter to solve the prediction weakness of ByteTrack[70] in accelerated scenarios, which can simply improve its robustness.

2. Related Work

Object detection and identity association are two key stages of multi-object tracking. Detection estimates the bounding boxes and association obtains the identities.
Figure 2. Comparison of tracking effect between uniform Kalman filter and accelerated Kalman filter in accelerated scene. (a) shows prediction and matching problems of uniform Kalman filter in accelerated scenarios. (b) shows the tracklets obtained by uniform Kalman filter. The same box color represents the same identity. (c) shows the tracklets obtained by our method. The red is the acceleration of the object in current frame.

2.1. Object Detection in MOT

Object detection is one of the most active topics in computer vision and it is the basis of multi-object tracking. The MOT17 set [36] provides detection results obtained by popular detectors such as DPM [17], Faster R-CNN [41] and SDP [64]. A large number of methods [61, 12, 2, 10, 77, 7, 23] focus on improving the tracking performance based on these given detection results. The association ability of these methods can be fairly compared.

**Tracking by detection.** With the rapid development of object detection [41, 22, 40, 29, 8, 18, 49, 47], more and more methods begin to use more powerful detectors to obtain higher tracking performance. The one-stage object detector RetinaNet [29] begin to be used by several methods such as [32, 39]. CenterNet [76] is the most popular detector used by most methods [75, 72, 60, 74, 57, 52, 55] for its simplicity and efficiency. The YOLO series detectors [40, 6] are also used by a large number of methods [58, 27, 28, 13] for its excellent balance of accuracy and speed. Most of these methods directly use the detection boxes on a single image for tracking.

However, the number of missing detection boxes and very low scoring detection boxes begin to increase when occlusion or motion blur happens in the video sequence, as is pointed out by video object detection methods [51, 34]. Therefore, the information of the previous frames are usually leveraged to enhance the video detection performance.

**Detection by tracking.** Tracking can also adopted to help obtain more accurate detection boxes. Some methods [44, 77, 12, 11, 13, 10] use single object tracking (SOT) [4] or Kalman filter [24] to predict the location of the tracklets in the following frame and fuse the predicted boxes with the detection boxes to enhance the detection results. Other methods [73, 28] use tracked boxes in the previous frames to enhance feature representation of the following frame. Recently, Transformer-based [53, 15, 56, 31] detectors [9, 78] are used by several methods [48, 35, 66] for its strong ability to propagate boxes between frames.

After obtaining the detection boxes by various detectors, most MOT methods [58, 72, 38, 32, 27, 60, 48] only keep the high score detection boxes by a threshold, i.e. 0.5, and use those boxes as the input of identity association. This is because the low
score detection boxes contain many backgrounds which harm the tracking performance. However, ByteTrack\cite{70} observe that many occluded objects can be correctly detected but have low scores. To reduce missing detection boxes and keep the persistence of trajectories, ByteTrack\cite{70} keep all the detection boxes and associate across every of them. So ByteTrack\cite{70} has done a phenomenal job.

2.2. Identity Association

Identity association is the core of multi-object tracking, which first computes the similarity between tracklets and detection boxes and then matches them according to the similarity.

**Similarity metrics.** Location, motion and appearance are useful cues for association. SORT \cite{5} combines location and motion cues in a very simple way. It first uses Kalman filter \cite{24} to predict the location of the tracklets in the new frame and then computes the IoU between the detection boxes and the predicted boxes as the similarity. Some recent methods \cite{75, 48, 60} design networks to learn object motions and achieve more robust results in cases of large camera motion or low frame rate. Location and motion similarity are accurate in the short-range matching. Appearance similarity are helpful in the long-range matching. An object can be re-identified using appearance similarity after being occluded for a long period of time. Appearance similarity can be measured by the cosine similarity of the Re-ID features. DeepSORT \cite{59} adopts a stand-alone Re-ID model to extract appearance features from the detection boxes. Recently, joint detection and Re-ID models \cite{58, 72, 27, 32, 71, 38} becomes more and more popular because of their simplicity and efficiency.

**Matching strategy.** After similarity computation, matching strategy assigns identities to the objects. This can be done by Hungarian Algorithm \cite{25} or greedy assignment \cite{75}. SORT \cite{5} matches the detection boxes to the tracklets by once matching. DeepSORT \cite{59} proposes a cascaded matching strategy which first matches the detection boxes to the most recent tracklets and then to the lost ones. MOTDT \cite{10} first uses appearance similarity to match and then use the IoU similarity to match the unmatched tracklets. QuasiDense \cite{38} turns the appearance similarity into probability by a bi-directional softmax operation and uses a nearest neighbor search to accomplish matching. Attention mechanism \cite{53} can directly propagate boxes between frames and perform association implicitly. Recent methods such as \cite{35, 66} propose track queries to find the location of the tracked objects in the following frames. The matching is implicitly performed in the attention interaction process.

All these methods including ByteTrack\cite{70}, which use uniform Kalman filter to predict box in next frame, focus on how to design better association methods. However, we believe that correct and reasonable prediction determines the upper bound of identity association and we focus on how to have a better prediction strategy in complex movement scenarios.

3. Variability

We propose a simple, effective and generic movement prediction method, Variability. Different from previous methods \cite{58, 72, 27, 38} which use uniform Kalman filter to obtain predicted boxes, we first associate the high score detection boxes to the tracklets. Some tracklets get unmatched because it does not match to an appropriate high score detection box, which usually happens when occlusion, motion blur or size changing occurs.

We then associate the low score detection boxes and these unmatched tracklets to recover the objects in low score detection boxes and filter out background like ByteTrack\cite{70}, simultaneously. The pseudo-code of Variability is shown in Algorithm 1.

The input of Variability is a video sequence $V$, along with an object detector $\text{Det}$ and two Kalman filter $\text{KF, V-KF}$. We also set four thresholds $\tau_{\text{high}}, \tau_{\text{low}}, \tau_a$, and $\epsilon$. $\tau_{\text{high}}$ and $\tau_{\text{low}}$ are the detection score thresholds, $\tau_a$ is the acceleration threshold and $\epsilon$ is the tracking score threshold. The output of BYTE is the tracks $T$ of the video and each track contains the bounding box and identity of the object in each frame.

For each frame in the video, we predict the detection boxes and scores using the detector $\text{Det}$. We separate all the detection boxes into two parts $D_{\text{high}}$ and $D_{\text{low}}$ according to the detection score thresholds $\tau_{\text{high}}$ and $\tau_{\text{low}}$. For the detection boxes whose scores are higher than $\tau_{\text{high}}$, we put them into the high score detection boxes $D_{\text{high}}$. For those whose scores range from $\tau_{\text{low}}$ to $\tau_{\text{high}}$, we put them into the low score detection boxes $D_{\text{low}}$ (line 3 to 13 in Algorithm 1).

After separating the low score detection boxes and the high score detection boxes, we use Kalman filter $\text{KF}$ and $\text{V-KF}$ to predict the new locations of each track in $T$ according to the acceleration threshold. For the tracks whose accelerations are higher than $\tau_a$, we use Variability Kalman Filter $\text{V-KF}$ to predict the new locations. For whose accelerations are lower than $\tau_a$, we use uniform Kalman Filter $\text{KF}$ to predict the new locations (line 14 to 21 in Algorithm 1).

The first association is performed between the high score detection boxes $D_{\text{high}}$ and all the tracks $T$ (including the lost tracks $T_{\text{lost}}$). The similarity is computed by the IoU between the detection boxes $D_{\text{high}}$ and the predicted box of tracks $T$. The IoU distance between the $T$ was calculated to find out the tracks that were close to each other. For these tracks, ID
Algorithm 1: Pseudo-code of Variability based on Byte.

**Input:** A video sequence $V$; object detector $\text{Det}$; Kalman Filter $KF$; Variable Kalman Filter $\text{V-KF}$; acceleration threshold $\tau_a$; detection score threshold $\tau_{\text{high}}$, $\tau_{\text{low}}$; tracking score threshold $\epsilon$

**Output:** Tracks $T$ of the video

1. **Initialization:** $T \leftarrow \emptyset$
2. **for** frame $f_k$ in $V$ **do**
   3. /* predict detection boxes & scores */
   4. $D_k \leftarrow \text{Det}(f_k)$
   5. $D_{\text{high}} \leftarrow \emptyset$
   6. $D_{\text{low}} \leftarrow \emptyset$
   7. **for** $d$ in $D_k$ **do**
   8. if $d$.score $> \tau_{\text{high}}$ then
   9. $D_{\text{high}} \leftarrow D_{\text{high}} \cup \{d\}$
   10. end
   11. else if $d$.score $> \tau_{\text{low}}$ then
   12. $D_{\text{low}} \leftarrow D_{\text{low}} \cup \{d\}$
   13. end
   14. end
   15. /* predict new locations of tracks, different Kalman Filters are used according to the acceleration of the object */
   16. **for** $t$ in $T$ **do**
   17. if $t$.a $> \tau_a$ then
   18. $t \leftarrow \text{V-KF}(t)$
   19. end
   20. else if $t$.a $< \tau_a$ then
   21. $t \leftarrow KF(t)$
   22. end
   23. end
   24. /* first association */
   25. Associate $T$ and $D_{\text{high}}$ using IoU distance
   26. The IOU distance between $T$ and $T$ was calculated and the IoU distance of $D_{\text{high}}$ and $T$ with a far distance were appropriately increased
   27. $T_{\text{remain}} \leftarrow$ remaining object boxes from $D_{\text{high}}$
   28. $T_{\text{remain}} \leftarrow$ remaining tracks from $T$
   29. /* second association */
   30. Associate $T_{\text{remain}}$ and $D_{\text{low}}$ using IoU distance
   31. The IOU distance between $T_{\text{remain}}$ and $T_{\text{remain}}$ was calculated and the IoU distance of $D_{\text{low}}$ and $T_{\text{remain}}$ with a far distance were appropriately increased
   32. $T_{\text{re}-\text{remain}} \leftarrow$ remaining tracks from $T_{\text{remain}}$
   33. /* delete unmatched tracks */
   34. $T \leftarrow T \setminus T_{\text{re}-\text{remain}}$
   35. /* initialize new tracks */
   36. **for** $d$ in $D_{\text{remain}}$ **do**
   37. if $d$.score $> \epsilon$ then
   38. $T \leftarrow T \cup \{d\}$
   39. end
   40. end
   41. Return: $T$

Track rebirth [59, 75] is not shown in the algorithm for simplicity. In green is the key of our method.

switching was more likely to occur. For the tracks that were far apart, we appropriately increased the IoU distance between them and $D_{\text{high}}$, so as to reduce the number of matching. Then, we use Hungarian Algorithm [25] to finish the matching based on the similarity. In particular, if the IoU between the detection box and the tracklet box is smaller than 0.2, we reject the matching. We keep the unmatched detection boxed in $D_{\text{remain}}$ and the unmatched tracks in $T_{\text{remain}}$ (line 22 to 25 in Algorithm 1).

Variability is highly flexible like Byte and can be compatible to other different association methods. For example, when Variability is combined with DeepSORT [59], Re-ID feature is added into *first association* in Algorithm 1, others are the same. you can apply Variability like BYTE to 9 different state-of-the-art trackers and will help improve the
| Tracker            | MOTA↑ | IDFI↑ | HOTA↑ | MT↑ | ML↓ | FP↓ | FN↓ | ID↓ | FPS↑ |
|--------------------|-------|-------|-------|-----|-----|-----|-----|-----|------|
| DAN [50]           | 52.4  | 49.5  | 39.3  | 21.4%| 30.7%|25423|234592| 8431| <3.9 |
| Tube_TK [37]       | 63.0  | 58.6  | 48.0  | 31.2%| 19.9%|27060|177483| 4137| 3.0  |
| MOTR [66]          | 65.1  | 66.4  | -     | 33.0%| 25.2%|45486|149307| 2049| -    |
| Chained-Tracker [39]| 66.6  | 57.4  | 49.0  | 37.8%| 18.5%|22284|160491| 5529| 6.8  |
| CenterTrack [75]   | 67.8  | 64.7  | 52.2  | 34.6%| 24.6%|18498|160332| 3038| 17.5 |
| QuasiDense [38]    | 68.7  | 66.3  | 53.9  | 40.6%| 21.9%|26589|146643| 3378| 20.3 |
| TraDes [60]        | 69.1  | 63.9  | 52.7  | 36.4%| 21.5%|20892|150060| 3555| 17.5 |
| MAT [21]           | 69.5  | 63.1  | 53.8  | 43.8%| 18.9%|30660|138741| 2844| 9.0  |
| SOTMOT [74]        | 71.0  | 71.9  | -     | 42.7%| 15.3%|39537|118983| 5184| 16.0 |
| TransCenter [62]   | 73.2  | 62.2  | 54.5  | 40.8%| 18.5%|23112|123738| 4614| 1.0  |
| GSDT [57]          | 73.2  | 66.5  | 55.2  | 41.7%| 17.5%|26397|120666| 3891| 4.9  |
| Semi-TCL [26]      | 73.3  | 73.2  | 59.8  | 41.8%| 18.7%|22944|124980| 2790| -    |
| FairMOT [72]       | 73.7  | 72.3  | 59.3  | 43.2%| 17.3%|27507|117477| 3303| 25.9 |
| RelationTrack [65] | 73.8  | 74.7  | 61.0  | 41.7%| 23.2%|27999|118623|1374| 8.5  |
| PermaTrackPr [52]  | 73.8  | 68.9  | 55.5  | 43.8%| 17.2%|28998|115104|3699| 11.9 |
| CSTrack [27]       | 74.9  | 72.6  | 59.3  | 41.5%| 17.5%|23847|114303|3567| 15.8 |
| TransTrack [48]    | 75.2  | 63.5  | 54.1  | 55.3%| 10.2%|50157|86442|3603| 10.0 |
| FUFE [45]          | 76.2  | 68.0  | 57.9  | 51.1%| 13.6%|32796|98475|3237| 6.8  |
| SiamMOT [28]       | 76.3  | 72.3  | -     | 44.8%| 15.5%|-    | -    | -   | 12.8 |
| CorrTracker [55]   | 76.5  | 73.6  | 60.7  | 47.6%| 12.7%|29808|99510|3369| 15.6 |
| TransMOT [13]      | 76.7  | 75.1  | 61.7  | 51.0%| 16.4%|36231|93150|2346| 9.6  |
| ReMOT [63]         | 77.0  | 72.0  | 59.7  | 51.7%| 13.8%|32024|93612|2853| 1.8  |
| ByteTrack [63]     | 80.3  | 77.3  | 63.1  | 53.2%| 14.5%|25491|83721|2196| 29.6 |
| **VariabilityTrack** (ours) | **80.4** | **77.5** | **63.2** | **53.2%** | **14.5%** | **25381** | **83611** | **2175** | **28.3** |

Table 1. Comparison of the state-of-the-art methods under the “private detector” protocol on MOT17 test set. The best results are shown in **bold**. MOT17 contains rich scenes and half of the sequences are captured with camera motion. ByteTrack[70] ranks 1st among all the trackers on the leaderboard of MOT17 and outperforms the second one ReMOT by a large margin on almost all the metrics. It also has the highest running speed among all the trackers.

4. VariabilityTrack

we use high-performance detector YOLOX [20] like ByteTrack[70] with our prediction method Variability.

YOLOX switches the YOLO series detectors [40, 6] to an anchor-free manner and conduct other advanced detection techniques, including decoupled heads, strong data augmentations, such as Mosaic [6] and Mixup [67], and effective label assignment strategy SimOTA [19] to achieve state-of-the-art performance on object detection.

The backbone network is the same as YOLOv5 [1] which adopts an advanced CSPNet [54] backbone and an additional PAN [30] head. There are two decoupled heads after the backbone network, one for regression and the other for classification. An additional IoU-aware branch is added to the regression head to predict the IoU between the predicted box and the ground

---

effect in accelerated scenarios

The second association is performed between the low score detection boxes $D_{low}$ and the remaining tracks $T_{remain}$ after the first association. Same process as first association. We keep the unmatched tracks in $T_{re-remain}$ and just delete all the unmatched low score detection boxes, since we view them as background. (line 26 to 28 in Algorithm 1).

Finally, we initialize new tracks from the unmatched high score detection boxes $D_{remain}$ after the first association. For each detection box in $D_{remain}$, if its detection score is higher than $\epsilon$ and exists for two consecutive frames, we initialize a new track (line 29 to 34 in Algorithm 1).

The output of each individual frame is the bounding boxes and identities of the tracks $T$ in the current frame. Note that we do not output the boxes and identities of $T_{lost}$. 

---

4. VariabilityTrack

we use high-performance detector YOLOX [20] like ByteTrack[70] with our prediction method Variability.

YOLOX switches the YOLO series detectors [40, 6] to an anchor-free manner and conduct other advanced detection techniques, including decoupled heads, strong data augmentations, such as Mosaic [6] and Mixup [67], and effective label assignment strategy SimOTA [19] to achieve state-of-the-art performance on object detection.

The backbone network is the same as YOLOv5 [1] which adopts an advanced CSPNet [54] backbone and an additional PAN [30] head. There are two decoupled heads after the backbone network, one for regression and the other for classification. An additional IoU-aware branch is added to the regression head to predict the IoU between the predicted box and the ground
| Tracker           | MOTA↑ | IDF1↑ | HOTA↑ | MT↑ | ML↓ | FP↓ | FN↓ | IDs↓ | FPS  |
|------------------|-------|-------|-------|-----|-----|-----|-----|-----|------|
| MLT [69]         | 48.9  | 54.6  | 43.2  | 30.9%| 22.1%|45660|216803|2187 |3.7   |
| FairMOT [72]     | 61.8  | 67.3  | 54.6  | 68.8%| 7.6% |103440|88901 |5243 |13.2  |
| TransCenter [62] | 61.9  | 50.4  | -     | 49.4%| 15.5%|45895 |146347|4653 |1.0   |
| TransTrack [48]  | 65.0  | 59.4  | 48.5  | 50.1%| 13.4%|27197 |150197|3608 |7.2   |
| CorrTracker [55] | 65.2  | 69.1  | -     | 66.4%| 8.9% |79429 |95855 |5183 |8.5   |
| Semi-TCL [26]    | 65.2  | 70.1  | 55.3  | 61.3%| 10.5%|61209 |114709|4139 |-     |
| CTrack [27]      | 66.6  | 68.6  | 54.0  | 50.4%| 15.5%|25404 |144358|3196 |4.5   |
| GSST [57]        | 67.1  | 67.5  | 53.6  | 53.1%| 13.2%|31913 |135409|3131 |0.9   |
| SiamMOT [28]     | 67.1  | 69.1  | -     | 49.0%| 16.3%| -   | -   | -   |4.3   |
| RelationTrack [65]| 67.2  | 70.5  | 56.5  | 62.2%| 8.9% |61134 |104597|4243 |2.7   |
| SOTMOT [74]      | 68.6  | 71.4  | -     | 64.9%| 9.7% |57064 |101154|4209 |8.5   |
| ByteTrack [74]   | 77.8  | 75.2  | 61.3  | 69.2%| 9.5% |26249 |87594 |1223 |17.5  |
| VariabilityTrack (ours) | 77.8  | 75.2  | 61.3  | 69.2%| 9.5% |26249 |87594 |1223 |17.5  |

Table 2. Comparison of the state-of-the-art methods under the “private detector” protocol on MOT20 test set. The best results are shown in bold. The scenes in MOT20 are much more crowded than those in MOT17. ByteTrack [70] ranks 1st among all the trackers on the leaderboard of MOT20 and outperforms the second one SOTMOT by a large margin on all the metrics. It also has the highest running speed among all the trackers.

| Method       | w/ Re-ID | MOTA↑ | IDF1↑ | IDs↓ | FPS  |
|--------------|----------|-------|-------|------|------|
| SORT         |          | 74.6  | 76.9  | 291  | 30.1 |
| DeepSORT     | ✓        | 75.4  | 77.2  | 239  | 13.5 |
| MOTDT        | ✓        | 75.8  | 77.6  | 273  | 11.1 |
| BYTE         |          | 76.6  | 79.3  | 159  | 29.6 |
| Variability (ours) |      | 76.7  | 79.4  | 123  | 28.4 |

Table 3. Comparison of different identity association methods on the MOT17 validation set. The best results are shown in bold.

truth box. The regression head directly predicts four values in each location in the feature map, i.e., two offsets in terms of the left-top corner of the grid, and the height and width of the predicted box. The regression head is supervised by GIoU loss [42] and the classification and IoU heads are supervised by the binary cross entropy loss.

5. Experiments

5.1. Setting

Datasets. We evaluate Variability and VariabilityTrack on MOT17 [36] and MOT20 [14] datasets. Both datasets contain training sets and test sets, without validation sets. For ablation studies, we use the first half of each video in the training set of MOT17 for training and the last half for validation following [75]. We train on the combination of CrowdHuman dataset [46] and MOT17 half training set following [75, 48, 66, 60]. We add Cityperson [68] and ETHZ [16] for training following [58, 72, 27] when testing on the test set of MOT17.

Metrics. We use the CLEAR metrics [3], including MOTA, FP, FN, IDs, etc., IDF1 [43] and HOTA [33] to evaluate different aspects of the tracking performance. MOTA is computed based on FP, FN and IDs. Considering the amount of FP and FN are larger than IDs, MOTA focuses more on the detection performance. IDF1 evaluates the identity preservation ability and focus more on the association performance. HOTA is a very recently proposed metric which explicitly balances the effect of performing accurate detection, association and localization.

MOT17. ByteTrack [70] ranks 1st among all the trackers on the leaderboard of MOT17. Not only does it achieve the best accuracy (i.e. 80.3 MOTA, 77.3 IDF1 and 63.1 HOTA), but also runs with highest running speed (30 FPS). Based on ByteTrack [70], we improved its tracking performance ability in accelerated motion perspective and named the new method VariabilityTrack. Due to the selective use of accelerated Kalman filter for more accurate prediction, its lower bound
MOT20. Compared with MOT17, MOT20 has much more crowded scenarios and occlusion cases. The average number of pedestrians in an image is 170 in the test set of MOT20. ByteTrack \cite{70} also ranks 1st among all the trackers on the leaderboard of MOT20 and outperforms other state-of-the-art trackers by a large margin on almost all the metrics. For example, it increases MOTA from 68.6 to 77.8, IDF1 from 71.4 to 75.2 and decreases IDs by 71\% from 4209 to 1223. It is worth noting that ByteTrack \cite{70} achieves extremely low identity switches, which further indicates that associating every detection boxes is very effective under occlusion cases. Since MOT20 has no motion perspective, our performance is the same as ByteTrack \cite{70}.

6. Conclusion

We propose a very simple and effective algorithm that can select variable speed Kalman filter according to the feedback of complex motion scenes, which can improve the algorithm that uses uniform speed Kalman filter algorithm for prediction and association matching like ByteTrack\cite{70}, and can significantly improve its tracking ability under the perspective of driving and UAV. This also brings us to think whether we can design a neural network to dynamically select a more personalized Kalman filter for each tracking target in the future, so as to make target trajectory prediction and accuracy, so as to improve the tracking effect in different complex scenes.

References

\begin{enumerate}
\item Yolov5. \url{https://github.com/ultralytics/yolov5}, 2020.
\item P. Bergmann, T. Meinhardt, and L. Leal-Taixé. Tracking without bells and whistles. In ICCV, pages 941–951, 2019.
\item K. Bernardin and R. Stiefelhagen. Evaluating multiple object tracking performance: the clear mot metrics. EURASIP Journal on Image and Video Processing, 2008:1–10, 2008.
\item L. Bertinetto, J. Valmadre, J. F. Henriques, A. Vedaldi, and P. H. Torr. Fully-convolutional siamese networks for object tracking. In European conference on computer vision, pages 850–865. Springer, 2016.
\item A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft. Simple online and realtime tracking. In ICIP, pages 3464–3468. IEEE, 2016.
\item A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao. Yolov4: Optimal speed and accuracy of object detection. \textit{arXiv preprint arXiv:2004.10934}, 2020.
\item G. Brasó and L. Leal-Taixé. Learning a neural solver for multiple object tracking. In \textit{Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition}, pages 6247–6257, 2020.
\item Z. Cai and N. Vasconcelos. Cascade r-cnn: Delving into high quality object detection. In CVPR, pages 6154–6162, 2018.
\item N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko. End-to-end object detection with transformers. In European Conference on Computer Vision, pages 213–229. Springer, 2020.
\item L. Chen, H. Ai, Z. Zhuang, and C. Shang. Real-time multiple people tracking with deeply learned candidate selection and person re-identification. In 2018 IEEE International Conference on Multimedia and Expo (ICME), pages 1–6. IEEE, 2018.
\item P. Chu, H. Fan, C. C. Tan, and H. Ling. Online multi-object tracking with instance-aware tracker and dynamic model refreshment. In 2019 IEEE Winter Conference on Applications of Computer Vision (WACV), pages 161–170. IEEE, 2019.
\item P. Chu and H. Ling. Famnet: Joint learning of feature, affinity and multi-dimensional assignment for online multiple object tracking. In ICCV, pages 6172–6181, 2019.
\item P. Chu, J. Wang, Q. You, H. Ling, and Z. Liu. Transmot: Spatial-temporal graph transformer for multiple object tracking. \textit{arXiv preprint arXiv:2104.00194}, 2021.
\item P. Dendorfer, H. Rezatofighi, A. Milan, J. Shi, D. Cremers, I. Reid, S. Roth, K. Schindler, and L. Leal-Taixé. Mot20: A benchmark for multi object tracking in crowded scenes. \textit{arXiv preprint arXiv:2003.09003}, 2020.
\item A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. \textit{arXiv preprint arXiv:2010.11929}, 2020.
\item A. Ess, B. Leibe, K. Schindler, and L. Van Gool. A mobile vision system for robust multi-person tracking. In CVPR, pages 1–8. IEEE, 2008.
\item P. Felzenszwalb, D. McAllester, and D. Ramanan. A discriminatively trained, multiscale, deformable part model. In CVPR, pages 1–8. IEEE, 2008.
\item J. Fu, L. Zong, Y. Li, K. Li, B. Yang, and X. Liu. Model adaption object detection system for robot. In 2020 39th Chinese Control Conference (CCC), pages 3659–3664. IEEE, 2020.
\item Z. Ge, S. Liu, Z. Li, O. Yoshiie, and J. Sun. Ota: Optimal transport assignment for object detection. In \textit{Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition}, pages 303–312, 2021.
\item Z. Ge, S. Liu, F. Wang, Z. Li, and J. Sun. Yolox: Exceeding yolo series in 2021. \textit{arXiv preprint arXiv:2107.08430}, 2021.
\end{enumerate}
[21] S. Han, P. Huang, H. Wang, E. Yu, D. Liu, X. Pan, and J. Zhao. Mat: Motion-aware multi-object tracking. arXiv preprint arXiv:2009.04792, 2020.
[22] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. In ICCV, pages 2961–2969, 2017.
[23] A. Hornakova, R. Henschel, B. Rosenhahn, and P. Swoboda. Lifted disjoint paths with application in multiple object tracking. In International Conference on Machine Learning, pages 4364–4375. PMLR, 2020.
[24] R. E. Kalman. A new approach to linear filtering and prediction problems. J. Fluids Eng., 82(1):35–45, 1960.
[25] H. W. Kuhn. The hungarian method for the assignment problem. Naval research logistics quarterly, 2(1-2):83–97, 1955.
[26] W. Li, Y. Xiong, S. Yang, M. Xu, Y. Wang, and W. Xia. Semi-tcl: Semi-supervised track contrastive representation learning. arXiv preprint arXiv:2107.02396, 2021.
[27] C. Liang, Z. Zhang, X. Zhou, B. Li, X. Ye, and J. Zou. Rethinking the competition between detection and Reid in multi-object tracking. arXiv preprint arXiv:2010.12138, 2020.
[28] C. Liang, Z. Zhang, X. Zhou, B. Li, Y. Lu, and W. Hu. One more check: Making “fake background” be tracked again. arXiv preprint arXiv:2104.09441, 2021.
[29] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár. Focal loss for dense object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 8759–8768, 2018.
[30] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo. Swin transformer: Hierarchical vision transformer using shifted windows. arXiv preprint arXiv:2103.14030, 2021.
[31] Z. Lu, V. Rathod, R. Votel, and J. Huang. Retinatrack: Online single stage joint detection and tracking. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 14668–14678, 2020.
[32] J. Luiten, A. Osep, P. Dendorfer, P. Torr, A. Geiger, L. Leal-Taixé, and B. Leibe. Hota: A higher order metric for evaluating multi-object tracking. International journal of computer vision, 129(2):548–578, 2021.
[33] H. Luo, W. Xie, X. Wang, and W. Zeng. Detect or track: Towards cost-effective video object detection/tracking. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 33, pages 8803–8810, 2019.
[34] T. Meinhardt, A. Kirillov, L. Leal-Taixe, and C. Feichtenhofer. Trackformer: Multi-object tracking with transformers. arXiv preprint arXiv:2101.02702, 2021.
[35] A. Milan, L. Leal-Taixé, I. Reid, S. Roth, and K. Schindler. Mot16: A benchmark for multi-object tracking. arXiv preprint arXiv:1603.00831, 2016.
[36] B. Pang, Y. Li, Y. Zhang, M. Li, and C. Lu. Tubetk: Adopting tubes to track multi-object in a one-step training model. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 6308–6318, 2020.
[37] J. Pang, L. Qiu, X. Li, H. Chen, Q. Li, T. Darrell, and F. Yu. Quasi-dense similarity learning for multiple object tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 164–173, 2021.
[38] J. Peng, C. Wang, F. Wan, Y. Wu, Y. Wang, Y. Tai, C. Wang, J. Li, F. Huang, and Y. Fu. Chained-tracker: Chaining paired attentive regression results for end-to-end joint multiple-object detection and tracking. In European Conference on Computer Vision, pages 145–161. Springer, 2020.
[39] J. Redmon and A. Farhadi. Yolov3: An incremental improvement. arXiv preprint arXiv:1804.02767, 2018.
[40] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In Advances in neural information processing systems, pages 91–99, 2015.
[41] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, and S. Savarese. Generalized intersection over union: A metric and a loss for bounding box regression. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 658–666, 2019.
[42] E. Ristani, F. Solera, R. Zou, R. Cucchiara, and C. Tomasi. Performance measures and a data set for multi-target, multi-camera tracking. In ECCV, pages 17–35. Springer, 2016.
[43] R. Sanchez-Martilla, F. Poiesi, and A. Cavallaro. Online multi-target tracking with strong and weak detections. In ECCV, pages 84–99. Springer, 2016.
[44] C. Shan, C. Wei, B. Deng, J. Huang, X.-S. Hua, X. Cheng, and K. Liang. Tracklets predicting based adaptive graph tracking. arXiv preprint arXiv:2010.09015, 2020.
[45] S. Shao, Z. Zhao, B. Li, T. Xiao, G. Yu, X. Zhang, and J. Sun. Crowdhuman: A benchmark for detecting human in a crowd. arXiv preprint arXiv:1805.00123, 2018.
[46] P. Sun, Y. Jiang, E. Xie, W. Shao, Z. Yuan, C. Wang, and P. Luo. What makes for end-to-end object detection? In Proceedings of the 38th International Conference on Machine Learning, volume 139 of Proceedings of Machine Learning Research, pages 9934–9944. PMLR, 2021.
[47] P. Sun, Y. Jiang, R. Zhang, E. Xie, J. Cao, X. Hu, T. Kong, Z. Yuan, C. Wang, and P. Luo. Transtrack: Multiple-object tracking with transformer. arXiv preprint arXiv:2012.15460, 2020.
[48] P. Sun, R. Zhang, Y. Jiang, T. Kong, C. Xu, W. Zhan, M. Tomizuka, L. Li, Z. Yuan, C. Wang, et al. Sparse r-cnn: End-to-end object detection with learnable proposals. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 14454–14463, 2021.
C.-Y. Wang, H.-Y. M. Liao, Y.-H. Wu, P.-Y. Chen, J.-W. Hsieh, and I.-H. Yeh. Cspnet: A new backbone that can enhance learning capability of cnn. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops, pages 390–391, 2020.

Q. Wang, Y. Zheng, P. Pan, and Y. Xu. Multiple object tracking with correlation learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 3876–3886, 2021.

W. Wang, E. Xie, X. Li, D.-P. Fan, K. Song, D. Liang, T. Lu, P. Luo, and L. Shao. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. arXiv preprint arXiv:2102.12122, 2021.

Y. Wang, K. Kitani, and X. Wang. Joint object detection and multi-object tracking with graph neural networks. arXiv preprint arXiv:2006.13164, 2020.

Z. Wang, L. Zheng, Y. Liu, Y. Li, and S. Wang. Towards real-time multi-object tracking. In Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020. Proceedings, Part XI 16, pages 107–122. Springer, 2020.

N. Wojke, A. Bewley, and D. Paulus. Simple online and realtime tracking with a deep association metric. In 2017 IEEE international conference on image processing (ICIP), pages 3645–3649. IEEE, 2017.

J. Wu, J. Cao, L. Song, Y. Wang, M. Yang, and J. Yuan. Track to detect and segment: An online multi-object tracker. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 12352–12361, 2021.

J. Xu, Y. Cao, Z. Zhang, and H. Hu. Spatial-temporal relation networks for multi-object tracking. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 3898–3908, 2019.

Y. Xu, Y. Ban, G. Delorme, C. Gan, D. Rus, and X. Alameda-Pineda. Transcenter: Transformers with dense queries for multiple-object tracking. arXiv preprint arXiv:2103.15145, 2021.

F. Yang, X. Chang, S. Sakti, Y. Wu, and S. Nakamura. Remot: A model-agnostic refinement for multiple object tracking. Image and Vision Computing, 106:104091, 2021.

F. Yang, W. Choi, and Y. Lin. Exploit all the layers: Fast and accurate cnn object detector with scale dependent pooling and cascaded rejection classifiers. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 2129–2137, 2016.

E. Yu, Z. Li, S. Han, and H. Wang. Relationtrack: Relation-aware multiple object tracking with decoupled representation. arXiv preprint arXiv:2105.04322, 2021.

F. Zeng, B. Dong, T. Wang, C. Chen, X. Zhang, and Y. Wei. Motr: End-to-end multiple-object tracking with transformer. arXiv preprint arXiv:2105.03247, 2021.

H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz. mixup: Beyond empirical risk minimization. arXiv preprint arXiv:1710.09412, 2017.

S. Zhang, R. Benenson, and B. Schiele. Citypersons: A diverse dataset for pedestrian detection. In CVPR, pages 3213–3221, 2017.

Y. Zhang, H. Sheng, Y. Wu, S. Wang, W. Ke, and Z. Xiong. Multiplex labeling graph for near-online tracking in crowded scenes. IEEE Internet of Things Journal, 7(9):7892–7902, 2020.

Y. Zhang, P. Sun, Y. Jiang, D. Yu, Z. Yuan, P. Luo, W. Liu, and X. Wang. Bytetrack: Multi-object tracking by associating every detection box. arXiv preprint arXiv:2110.06864, 2021.

Y. Zhang, C. Wang, X. Wang, W. Liu, and W. Zeng. Voxeltrack: Multi-person 3d human pose estimation and tracking in the wild. arXiv preprint arXiv:2108.02452, 2021.

Y. Zhang, C. Wang, X. Wang, W. Zeng, and W. Liu. Fairmot: On the fairness of detection and re-identification in multiple object tracking. arXiv preprint arXiv:2004.01888, 2020.

Z. Zhang, D. Cheng, X. Zhu, S. Lin, and J. Dai. Integrated object detection and tracking with tracklet-conditioned detection. arXiv preprint arXiv:1811.11167, 2018.

L. Zheng, M. Tang, Y. Chen, G. Zhu, J. Wang, and H. Lu. Improving multiple object tracking with single object tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 2453–2462, 2021.

X. Zhou, V. Koltun, and P. Krähenbühl. Tracking objects as points. In European Conference on Computer Vision, pages 474–490. Springer, 2020.

X. Zhou, D. Wang, and P. Krähenbühl. Objects as points. arXiv preprint arXiv:1904.07850, 2019.

J. Zhu, H. Yang, N. Liu, M. Kim, W. Zhang, and M.-H. Yang. Online multi-object tracking with dual matching attention networks. In Proceedings of the European Conference on Computer Vision (ECCV), pages 366–382, 2018.

X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai. Deformable detr: Deformable transformers for end-to-end object detection. arXiv preprint arXiv:2010.04159, 2020.