Modelling Ambiguous Assignments for Multi-Person Tracking in Crowds

Daniel Stadler\textsuperscript{1,2,3} \hspace{1em} Jürgen Beyerer\textsuperscript{2,1,3}

\textsuperscript{1}Karlsruhe Institute of Technology \hspace{1em} \textsuperscript{2}Fraunhofer IOSB \hspace{1em} \textsuperscript{3}Fraunhofer Center for Machine Learning

\{daniel.stadler, juergen.beyerer\}@iosb.fraunhofer.de

Abstract

Multi-person tracking is often solved with a tracking-by-detection approach that matches all tracks and detections simultaneously based on a distance matrix. In crowded scenes, ambiguous situations with similar track-detection distances occur, which leads to wrong assignments. To mitigate this problem, we propose a new association method that separately treats such difficult situations by modelling ambiguous assignments based on the differences in the distance matrix. Depending on the numbers of tracks and detections, for which the assignment task is determined ambiguous, different strategies to resolve these ambiguous situations are proposed. To further enhance the performance of our tracking framework, we introduce a camera motion-aware interpolation technique and make an adaptation to the motion model, which improves identity preservation. The effectiveness of our approach is demonstrated through extensive ablative experiments with different detection models. Moreover, the superiority w.r.t. other trackers is shown on the challenging MOT17 and MOT20 datasets, where state-of-the-art results are obtained.

1. Introduction

Multi-person tracking (MPT) demands the localization and identification of all targets throughout a video sequence and is a basic component for several applications like human activity detection or surveillance related tasks.

The predominant methodology to solve the MPT problem is the tracking-by-detection paradigm \cite{4, 5, 24, 38, 42, 48, 50, 51}. An object detector is applied in every frame of a sequence and the generated detections are associated to the current tracks based on a distance measure. For example, the Intersection over Union (IoU) of detection and track box is often used. Mostly, the Hungarian method \cite{18} is leveraged for solving the assignment problem.

While the association task is easy if targets are far away from each other, the assignment of detections to tracks can be ambiguous in crowded scenes, where persons have similar spatial positions. We therefore argue that it is promising to treat those ambiguous situations separately and develop an association method which explicitly models ambiguous assignments by looking more closely at the distance matrix of all possible track-detection matches.

We determine ambiguous situations by introducing a similarity constraint, which indicates for two possible track-detection matches whether they are similar, and scan the distance matrix for all similar assignments. Depending on the numbers of detections and tracks that lead to similar assignments, \textit{i.e.}, whether there are more tracks or more detections, we propose different strategies to resolve the ambiguous situations. For example, we find that one simple yet effective method of handling so-called ambiguous detections (there are less detections than tracks) is to simply delete those detections and propagate the involved tracks with the motion model until there are clear matches again. The reason for the success of this deletion strategy lies in the fact that ambiguous detection boxes are often located between two tracks covering parts of both targets due to the inaccuracy of the detectors in crowded scenes. For the other kind of ambiguous situations, \textit{i.e.}, there are more detections than tracks which lead to ambiguous assignments, the best investigated method is an initialization strategy, that suppresses duplicate detections by demanding a target to be detected multiple consecutive frames to start a new track.

After resolving all ambiguous situations, the standard assignment process applying the Hungarian method is performed for the remaining tracks and detections. Note that the proposed association method with the ambiguous assignments modelling needs only a distance matrix and a predefined similarity threshold as input. Therefore, it can be included in any tracking-by-detection based approach, independent from how the distance matrix is calculated.

Besides the approach of modelling and resolving ambiguous assignments, we find that trackers from literature either include an interpolation mechanism for occluded track boxes and neglect the influence of moving cameras \cite{28}, or apply a camera motion compensation (CMC) model \cite{2}. To take advantage of both modules, we develop a camera motion-aware interpolation technique that transforms start and end box of the interpolation into a common...
frame using the transformation matrices coming from the CMC model before the interpolation is performed.

We observe another problem caused by inaccurate detection boxes right before track inactivation due to occlusion: The change of box size, in particular of box height, is often overestimated in the motion model which makes propagated inactive track boxes shrink or grow too fast. As a consequence, re-identification after occlusion fails and identity conservation is harmed. To counteract this, we make an adaptation in the motion model that preserves the height of inactive track boxes during propagation.

The main contributions of our work are summarized in the following:

- We propose a novel association method for tracking-by-detection based approaches, that models ambiguous assignments by searching for possible track-detection matches with similar distances.
- Depending on the numbers of tracks and detections leading to ambiguous assignments, we investigate different strategies to resolve the ambiguous situations.
- A camera motion-aware interpolation technique is introduced and an adaptation to the motion model is made to further improve the tracking performance.

2. Related Work

Tracking-by-detection. Most of the multi-person tracking (MPT) approaches from the literature follow the tracking-by-detection paradigm \([4, 5, 24, 38, 42, 48, 50, 51]\), which splits the MPT task into two sub-problems: detection and association. With many detection models publicly available, most of the MPT research aims at improving the association task, for example, by designing advanced distance measures \([11, 36, 40, 51]\). One of the most used distance measures for assigning detections to tracks is based on IoU as in the SORT framework \([4]\). The further development DeepSORT \([48]\) additionally uses visual information of objects extracting appearance features with a separate convolutional neural network, which is also done in many other works \([24, 42, 51]\). For motion prediction of targets, often a Kalman filter is used in MPT. When additionally the camera is moving, another important component next to the targets motion model is the compensation of camera motion \([2]\), which makes the estimated position of propagated track boxes more accurate. Besides position, motion, and appearance information, human poses can be leveraged in MPT \([42, 50]\). Moreover, instead of only extracting information of single objects, some tracking methods also consider the relations between the targets \([24, 39, 51]\).

While the design of advanced distance measures, which combine information of several object cues \([11, 36, 40, 51]\), lead to an improved association performance, there still remain ambiguous situations, in that the distance measures give no clear picture about which of the possible track-detection matches are correct. Some approaches follow a hierarchical association scheme \([1, 37, 44]\), first matching detection boxes to short tracklets and afterwards matching on the tracklets level, to mitigate this issue. However, those methods cannot be processed in an online manner, which makes them unsuitable for real-time applications. In contrast, we propose a new online association method, that models and resolves ambiguous assignments needing only the current distance matrix of track-detection pairs as input.

Handling ambiguities. A common strategy to handle ambiguous situations in the association process is to follow a multiple hypothesis tracking (MHT) approach \([17, 19, 53]\), where multiple association hypotheses are maintained for several time steps to find the optimal solution. However, MHT often comes with a high computational complexity which increases exponentially with the number of considered time steps. As already mentioned, another idea to overcome ambiguities is to pursue a hierarchical association scheme \([1, 37, 44]\), in which first short high confidence tracklets are generated before they are merged to longer trajectories. In \([29]\), split-merge conditions are introduced to deal with missing detections of occluded tracks. The special feature of our ambiguous assignments modelling is that ambiguities are determined with the distance matrix of possible track-detection matches and therefore can be included in any tracking-by-detection approach.

Track interpolation. Many methods apply a simple linear interpolation of track boxes to close the gaps of recovered tracks after occlusion \([15, 28, 29, 30]\). While this works well in scenes with limited motion dynamics, the interpolated boxes are inaccurate when severe camera motion occurs. In \([6]\), single-object trackers like KCF \([14]\) or Medialflow \([16]\) are used for handling fragmented tracks. However, this brings a computational overhead and the single-object trackers also suffer from occlusion so that the interpolation might fail. The cyclic pseudo-observation trajectory filling strategy from \([12]\) incorporates camera motion in the interpolation process, however, future frames are needed for motion prediction. Thus, the interpolation can only be performed as post-processing. In contrast, our interpolation method needs no future information and is performed immediately when occluded tracks are recovered.

3. Proposed Method

We first describe our approach to find ambiguous assignments in Section 3.1. Then, different methods for resolving ambiguous assignments are proposed in Section 3.2. Furthermore, we introduce a camera motion-aware interpolation module and an adapted motion model in Section 3.3.
3.1. Modelling Ambiguous Assignments

In each time step $t$ of a tracking-by-detection based method, the generated detections $D^t = [D_1, \ldots, D_N]$ are assigned to the tracks from the previous iteration $T^{t-1} = [T_1, \ldots, T_M]$ based on a distance matrix $D \in \mathbb{R}^{M \times N}$. When targets are far away from each other and clearly visible, the assignment task is easy. However, in crowded scenes, the association of detections to tracks can become ambiguous, e.g., because of missing detections, and thus, the risk for tracking errors is high. Instead of treating all possible matches equally by applying the Hungarian method [18] on the full distance matrix $D$, we propose to handle ambiguous assignments separately, and after that, apply the Hungarian method only on a reduced distance matrix $D_{\text{clear}}$ with remaining clear assignments.

To find ambiguous situations, we first search for similar assignments by comparing the distances of possible track-detection matches and introduce a similarity threshold $\Delta$. For example, if the distances of the two best matching tracks $T_l$ and $T_k$ w.r.t. a detection $D_j$ differ by less than $\Delta$ (and both distances are below a maximum distance $d_{\text{max}}$ – here ignored for clarity), those possible matches belong to the set of similar assignments $A^s$:

$$|D[i, k] - D[j, k]| < \Delta \iff \{i, j\} \in A^s$$ (1)

$D_k$ is termed an ambiguous detection as it is not clear to which track the detection should be assigned. Similarly, there can be an ambiguous track $T_l$ if for the best matching detections $D_{m_l}$ and $D_{n_l}$ w.r.t. $T_l$ the following holds:

$$|D[l, m] - D[l, n]| < \Delta \iff \{l, m, n\} \in A^s$$ (2)

While these two examples illustrate the idea of similar assignments, note that in crowded scenes and depending on the choice of $\Delta$, similar assignments can include both multiple detections and tracks. In this case, the rows and columns of the distance matrix $D$ have to be scanned multiple times in order to get the complete set of similar assignments $A^s$.

Finally, the set of ambiguous assignments $A^a$ is the subset of $A^s$, where the numbers of detections and tracks differ. We do not keep similar assignments with equal numbers of detections and tracks as ambiguous assignments, since in such cases all detections and tracks can be matched. Thus, the relation of $A^a$ and $A^s$ can be expressed as:

$$A \in A^a \implies A \in A^s \iff |A[0]| \neq |A[1]|$$ (3)

The complete process, how ambiguous assignments are determined with the distance matrix $D$, the similarity threshold $\Delta$, and a maximum allowed distance $d_{\text{max}}$, can be found in Algorithm 1. Additionally, we provide a toy example distance matrix with highlighted similar, ambiguous, and clear assignments in Figure 1. The similar assignments with equal numbers of tracks and detections (orange) are not treated separately but resolved together with the clear assignments (teal) with the Hungarian method. Unassigned tracks are not terminated immediately but turn inactive for at most $i_{\text{max}}$ time steps and unassigned detections start new tracks. To resolve the ambiguous assignments, different

**Algorithm 1: Modelling Ambiguous Assignments**

<table>
<thead>
<tr>
<th>Input:</th>
<th>Distance matrix $D \in \mathbb{R}^{M \times N}$ of $M$ tracks and $N$ detections</th>
</tr>
</thead>
<tbody>
<tr>
<td>Similarity threshold $\Delta$, maximum distance $d_{\text{max}}$</td>
<td></td>
</tr>
<tr>
<td>Output:</td>
<td>Set of ambiguous assignments $A^a$</td>
</tr>
</tbody>
</table>

```python
A^a \leftarrow \emptyset; A^a \leftarrow \emptyset \quad // \text{sim. / amb. assignm.}
for i = 1 \ldots N \ do \ // \text{iterate over all dets}
    dets \leftarrow \emptyset; \text{tracks} \leftarrow \emptyset
    // start with best match
    dist \leftarrow \min(D[:i]) \ // \text{for comparison}
    sim_dets \leftarrow \{(i, dist)\} \ // (idx, distance)
    do \ // \text{find similar assignments}
        dets \leftarrow dets \cup \text{sim_dets}
        sim_tracks \leftarrow \emptyset
        // find similar tracks
        sim_dets \leftarrow \emptyset
        for n, dist \in \text{sim_dets} \ do
            t_idx \leftarrow \text{where}(D[:, n] - dist < \Delta)
            t_dists \leftarrow D[t_idx, n]
            for idx, d \in \text{zip}(t_idx, t_dists) \ do
                sim_tracks \leftarrow sim_tracks \cup \{(idx, d)\}
        tracks \leftarrow \text{tracks} \cup \text{sim_tracks}
        // find similar detections
        sim_dets \leftarrow \emptyset
        for m, dist \in \text{sim_tracks} \ do
            d_idx \leftarrow \text{where}(D[m, :] - dist < \Delta)
            d_dists \leftarrow D[m, d_idx]
            for idx, d \in \text{zip}(d_idx, d_dists) \ do
                sim_dets \leftarrow sim_dets \cup \{(idx, d)\}
        while \ sim_dets \ \neq \ \emptyset
        // keep idx of sim. dets / tracks
        A^s \leftarrow A^s \cup \{(t[0] \mid t \in \text{tracks}), \{d[0] \mid d \in \text{dets})\}
        // merge sets of sim. dets / tracks
        for A_k \in A^a \ do
            for A_l \in A^a \ \setminus \ \{A_k\} \ do
                if (A_k[0] \cap A_l[0]) \lor (A_k[1] \cap A_l[1]) \ then
                    A_k \leftarrow (A_k[0] \cup A_l[0], A_k[1] \cup A_l[1])
                    A^a \leftarrow A^a \ \setminus \ \{A_l\}
                    // save ambiguous assignments
                    for A \in A^a \ do
                        if n_T \neq n_D \ then
                            A^a \leftarrow A^a \ \cup \ \{A\}
```

135
### Figure 1: Illustration of a toy example distance matrix $D$ and resolving similar and ambiguous assignments $\mathcal{A}^s$ and $\mathcal{A}^a$, respectively, after the proposed ambiguous assignments modelling with $\Delta = 0.1$ and $d_{\text{max}} = 0.8$. $T_6$ is an ambiguous track (cyan) and $D_4$ is an ambiguous detection (purple). The orange colored similar assignments are not ambiguous, as the numbers of detections and tracks is equal. A clear match ($T_5, D_5$) is highlighted in teal color. Note that $D_{[2, 8]} = 0.87$ exceeds the maximum distance $d_{\text{max}}$. Thus, $T_2$ turns inactive and $D_8$ starts a new track.

<table>
<thead>
<tr>
<th></th>
<th>$D_1$</th>
<th>$D_2$</th>
<th>$D_3$</th>
<th>$D_4$</th>
<th>$D_5$</th>
<th>$D_6$</th>
<th>$D_7$</th>
<th>$D_8$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T_1$</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0.38</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>$T_2$</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0.31</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>$T_3$</td>
<td>0.92</td>
<td>1</td>
<td>1</td>
<td>0.06</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>$T_4$</td>
<td>1</td>
<td>1</td>
<td>0.11</td>
<td>1</td>
<td>1</td>
<td>0.19</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>$T_5$</td>
<td>1</td>
<td>0.29</td>
<td>1</td>
<td>1</td>
<td>0.37</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

$\mathcal{A}^s = \{([1, 4], [4]), ([3, 7], [2, 6]), ([6], [3, 7])\}$

$\mathcal{A}^a = \{([1, 4], [4]), ([6], [3, 7])\}$

The (ten best) hypotheses are updated in each time step until they can be resolved by clear matches. (3) The third strategy is to allow multiple assignments of detections so that two tracks competing for a detection can be both updated with it at the same time. In the standard association, detections are assigned to the best matching tracks with the Hungarian method and unassigned tracks turn inactive.

**Resolving ambiguous tracks ($n_D < n_T$).** Ambiguous tracks occur when multiple detections fit well to a track. This is the case for duplicate false positive detections, but also when new partly-occluded targets occur for the first time. To resolve ambiguous tracks, we investigate the following strategies. (1) Deleting the detections and inactivating the involved tracks. (2) Following a mht approach (ten best hypotheses maintained) until hypotheses can be resolved by clear matches. (3) Applying an initialization strategy in that unmatched detections start tentative tracks, which have to be confirmed in $n_{\text{active}}$ consecutive frames until activation – otherwise they are deleted. The standard procedure assigns to each track a detection with the Hungarian method and unassigned detections start new tracks.

### 3.3. Improved Interpolation and Motion Model

Many trackers apply a simple linear interpolation of track boxes when an occluded track is recovered and potential camera motion is neglected. We find that this simplification leads to bad interpolation results when the camera is moving and therefore propose a camera motion-aware interpolation technique, which makes use of a CMC model. Instead of copying the last detection box into the frame, in which the target is recovered, and generating the interpolated boxes without considering camera motion (standard linear interpolation), the following steps are performed:

1. The last detection box from frame $t_k$ is transformed with the transformation matrices $W_k, \ldots, W_m$ coming from a CMC model to the middle frame $t_m$, and the recovered box from frame $t_l$ is transformed with the inverse transformation matrices $W^{-1}_{l+1}, \ldots, W^{-1}_{m+1}$ to the middle frame $t_m$ with $m = \lceil (l - k)/2 \rceil$.

2. Linear interpolation of the transformed boxes is done.

3. The interpolated boxes are transformed back to their respective frames.

Note that a transformation matrix $W_n$ describes the motion from frame $n - 1$ to frame $n$. A visual illustration of our camera motion-aware interpolation technique and the standard linear interpolation is depicted in Figure 2.

As linear motion model for track propagation, we follow [48] using a Kalman filter with track state $T_{\text{state}} = (x, y, a, h, \dot{x}, y, \dot{a}, h)$, where $x, y, a, h$ are the track box vertical position, horizontal position, aspect ratio, and height.
respectively. While this model works well with precise detection boxes, we find it vulnerable w.r.t. low quality detections in crowded scenes, which is depicted in Figure 3. The last associated detection box, before a track turns inactive due to severe occlusion, might only contain body parts leading to an inaccurate bounding box which negatively affects the motion estimation in the Kalman filter. In our observations, a wrong height of the bounding box causes the most problems. Therefore, we propose to preserve the height of a track whenever it turns inactive by setting \( \dot{h} \) to zero.

The full pipeline of our tracker is listed in Algorithm 2. Contributions are highlighted with colors, which are in correspondence with Figures 1, 2, and 3. Note that the shown algorithm is the variant, where ambiguous tracks are re-

![Algorithm 2: Proposed tracking pipeline at time \( t \)](image)

- **Input:** Set of previous tracks \( T^{t−1} = \{T_1, \ldots, T_M\} \)
- **Input:** Set of current detections \( D^t = \{D_1, \ldots, D_N\} \)
- **Input:** Camera trafo matrices \( W = [W_1, \ldots, W_F]\)
- **Input:** Kalman filter (KF) with noise cov. \( Q \) and \( R \)
- **Input:** Similarity threshold \( \Delta \), maximum distance \( d_{max} \)
- **Output:** Updated set of tracks \( T^t \)

```
for T ∈ T^{t−1} do // apply motion models
    T ← camera_motion_compensation(T, W_t)
    T.mean, T.cov ← KF.predict(T.mean, T.cov, Q)
D ← \| − IoU(T^{t−1}, D^t) // distance matrix
// find and resolve amb. assignm.
ass_t_idx ← [ ]; ass_d_idx ← []
A^a ← find_amb_assignm(D, \Delta, d_{max}) // Alg. 1
for A ∈ A^a do // strategies from Sec. 3.2
    i, t ∈ T^{t−1} do // update tracks
        if i ∈ ass_t_idx then // assigned det
            j ← ass_d_idx[ass_t_idx.index(i)]
            T.mean, T.cov ← KF.update(T, D_j, R)
        if T.state = inactive then // Fig. 2
            cma_inter(T.mean, T.last_p, T.n_inac, W)
        if T.state = active then
            T.state ← inactive
        if T.state = inactive
            T.n_inac ← T.n_inac + 1
        if T.n_inac > i_{\max} then
            \( T^{t−1} ← T^{t−1} \setminus \{T\} \) // remove
        if T.state = active then
            T.state ← inactive
            T.mean[8] ← 0 // \( \dot{h} = 0 \), Fig. 3
for j ∈ D^t do // save updated tracks
    if j \notin ass_d_idx then
        T_{\text{new}} ← KF.initiate(D)
        T^t ← T^t \cup \{T_{\text{new}}\}
```

Figure 3: Motion prediction with (violet) and without (green) height preservation. Active track boxes are depicted in solid lines, propagated inactive track boxes in dashed lines. With height preservation \( \dot{h} = 0 \), the track can be continued after occlusion. Otherwise, the low quality box in the second frame makes the propagated track box shrink in each frame \( \dot{h} < 0 \), so that a re-activation fails.
4. Experiments

4.1. Datasets

**MOT17.** The MOT17 dataset [27] comprises 14 diverse sequences for multi-person tracking, 7 for training and testing each, including videos with both static as well as moving cameras. As the annotations for the test split are not publicly available, we follow [33, 34, 45, 49, 57] and divide the train split into two halves for ablative experiments.

**MOT20.** A more recent version of the MOTChallenge (https://motchallenge.net/) is MOT20 [9], which focuses on tracking in very crowded scenes. It consists of a train and a test split with 4 sequences each. For ablative experiments, the train split is also divided into two halves.

**CrowdHuman.** As one of the largest datasets for person detection, the CrowdHuman dataset [35] is frequently used for pre-training. It is divided into three splits – train (15000 images), validation (4370 images), and test (5000 images).

4.2. Evaluation Metrics

For evaluating tracking performance, we use MOTA [3], which incorporates numbers of false positives (FP), false negatives (FN), and identity switches (IDSW), as well as IDF1 [32], and the recently proposed HOTA [26]. Besides that, numbers of mostly tracked (MT) and mostly lost (ML) targets as well as number of fragmentations (FRAG) are reported. TrackEval [25] is used for calculating all metrics.

4.3. Implementation Details

**Tracker.** The parameters of our tracker are empirically set as follows: The threshold for the distances of two possible track-detection matches to be considered as similar is $\Delta = 0.1$. The minimum required IoU $\theta_{\text{min}}$ between a track and a detection box for matching is 0.2. Thus, the maximum allowed distance gets $d_{\text{max}} = 1 - \theta_{\text{min}} = 0.8$. The number of consecutive detections for a tentative track to become active, and the number of frames, a track is kept as inactive without assigned detection before termination, are set to $n_{\text{active}} = 4$ and $i_{\text{max}} = 40$, respectively. As motion model, a Kalman filter with the implementation of [48] is applied. For CMC on MOT17, the Enhanced Correlation Coefficient Maximization from [10] is leveraged. On MOT20, neither CMC nor the proposed camera motion-aware interpolation are performed, since there is hardly any camera motion in the sequences of MOT20.

**Detector.** Unless otherwise specified, a Faster R-CNN detector [31] with FPN [22] as neck and ResNet-50 [13] as backbone is used in ablative experiments. The model is pre-trained on CrowdHuman train split for 30 epochs with a batch size of 16 and a learning rate of 0.01, which is reduced by factor 10 after epochs 24 and 27. After that, fine-tuning on the first half of MOT17 train is conducted with an initial learning rate of 0.001 for another 30 epochs with the same schedule. When testing our tracker on the second half of MOT20 train, the first half of MOT20 train is taken for fine-tuning instead. In the comparison with the state-of-the-art on MOT17 / 20, the respective full train splits are taken for fine-tuning. The only used data augmentation is horizontal flipping. We also run experiments with RetinaNet [23] and the crowd-specific detector CrowdDet [8] applying the same neck, backbone, and training schedules. Faster RCNN and CrowdDet are trained with the implementation of [8] and MMDetection [7] is used for training RetinaNet. The non-maximum suppression threshold is set to 0.5, the minimum score threshold for detections kept in tracking is 0.9 for Faster RCNN and CrowdDet and 0.6 for RetinaNet.

4.4. Ablation Studies

**Modelling ambiguous assignments.** We first run experiments with the various strategies to resolve ambiguous detections and tracks proposed in Section 3.2. The tracking results are summarized in Table 1. The first line corresponds to the standard association, where no ambiguous assignments are modelled and all tracks and detections are assigned at once using the Hungarian method.

For resolving ambiguous detections, the strategy of deletion and waiting for clear matches yields the by far best results boosting MOTA, IDF1, and HOTA by 0.6, 1.3, and 0.8 points, respectively. The mht approach improves identity preservation, but MOTA is not enhanced. The strategy of allowing multiple associations does not work, as too many duplicate detections are introduced (FP increased by 70%). A qualitative example, in that the deletion of an ambiguous detection prevents an identity switch, is given in Figure 4.

As can be seen in Table 1, for resolving ambiguous tracks, the deletion strategy does only enhance identity preservation measured in IDF1, but MOTA is reduced due to an increased number of missing detections. In contrast,

<table>
<thead>
<tr>
<th>Amb. dets</th>
<th>Amb. tracks</th>
<th>MOTA</th>
<th>IDF1</th>
<th>HOTA</th>
</tr>
</thead>
<tbody>
<tr>
<td>standard</td>
<td>standard</td>
<td>74.0</td>
<td>76.9</td>
<td>62.9</td>
</tr>
<tr>
<td>delete</td>
<td>standard</td>
<td>74.6</td>
<td>78.2</td>
<td>63.7</td>
</tr>
<tr>
<td>mht</td>
<td>standard</td>
<td>73.2</td>
<td>77.3</td>
<td>63.2</td>
</tr>
<tr>
<td>multi</td>
<td>standard</td>
<td>66.3</td>
<td>74.6</td>
<td>60.8</td>
</tr>
<tr>
<td>standard</td>
<td>delete</td>
<td>73.1</td>
<td>77.2</td>
<td>62.8</td>
</tr>
<tr>
<td>standard</td>
<td>mht</td>
<td>74.4</td>
<td>76.6</td>
<td>62.8</td>
</tr>
<tr>
<td>standard</td>
<td>init</td>
<td>75.7</td>
<td>78.0</td>
<td>63.6</td>
</tr>
<tr>
<td>delete</td>
<td>init</td>
<td>76.5</td>
<td>79.4</td>
<td>64.5</td>
</tr>
</tbody>
</table>
the mht approach increases MOTA but lowers IDF1. The initialization technique, however, significantly improves all tracking measures: +1.7 MOTA, +1.1 IDF1, +0.7 HOTA. This is because duplicate detections often occur only in single frames, which do not initialize wrong tracks due to the tentative track state. At the same time, hardly any correct detections are removed with the initialization strategy.

The last line of Table 1 shows, that the deletion strategy for resolving ambiguous detections and the initialization strategy for resolving ambiguous tracks bring complementary gains, as their combination leads to further great improvements: The proposed modelling of ambiguous assignments and separate treatment of ambiguous detections and tracks raises both MOTA and IDF1 by 2.5 points and HOTA by 1.6 points w.r.t. the standard association.

**Improved interpolation and motion model.** We ablate the influence of the proposed camera motion-aware interpolation as well as the height preservation in the motion model in Table 2. The first two lines show the importance of interpolation for the final tracking performance, as even a linear interpolation significantly improves HOTA, IDF1, and MOTA by greatly reducing the number of FN. In comparison to the standard approach, our camera motion-aware interpolation achieves both lower values of FN and FP showing that the interpolated boxes are more accurate. This holds especially for sequences with severe camera motion, e.g., MOT17-13, in which the camera motion-aware interpolation improves MOTA by 2.1 points over the standard interpolation. Note that the used transformation matrices do not have to be calculated separately, but come from the CMC model and thus, the computational overhead w.r.t. the linear interpolation is negligible. The height preservation of inactive tracks enhances the accuracy of propagated track boxes which leads to a further improvement in identity preservation (+0.6 points), which is also beneficial for MOTA (+0.5 points) and HOTA (+0.2 points).

**Results with various detectors and datasets.** To demonstrate the generalization abilities of our tracker w.r.t. different detectors and datasets, we run several experiments applying detections from Faster RCNN, RetinaNet, and CrowdDet on MOT17 and MOT20. The results of these experiments are listed in Table 3. For each combination,

<table>
<thead>
<tr>
<th>Tracker</th>
<th>Dataset</th>
<th>Detector</th>
<th>IDF1</th>
<th>HOTA</th>
<th>MOTA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Base</td>
<td>MOT17</td>
<td>RetinaNet</td>
<td>66.3</td>
<td>54.5</td>
<td>59.3</td>
</tr>
<tr>
<td>Ours</td>
<td>MOT17</td>
<td>RetinaNet</td>
<td>69.6</td>
<td>56.8</td>
<td>62.9</td>
</tr>
<tr>
<td>Base</td>
<td>MOT17</td>
<td>FRCNN</td>
<td>76.1</td>
<td>62.6</td>
<td>73.2</td>
</tr>
<tr>
<td>Ours</td>
<td>MOT17</td>
<td>FRCNN</td>
<td>79.4</td>
<td>64.5</td>
<td>76.5</td>
</tr>
<tr>
<td>Base</td>
<td>MOT17</td>
<td>CrowdDet</td>
<td>78.2</td>
<td>64.3</td>
<td>73.5</td>
</tr>
<tr>
<td>Ours</td>
<td>MOT17</td>
<td>CrowdDet</td>
<td>80.6</td>
<td>65.6</td>
<td>76.7</td>
</tr>
<tr>
<td>Base</td>
<td>MOT20</td>
<td>RetinaNet</td>
<td>56.3</td>
<td>48.2</td>
<td>71.1</td>
</tr>
<tr>
<td>Ours</td>
<td>MOT20</td>
<td>RetinaNet</td>
<td>59.4</td>
<td>49.8</td>
<td>74.2</td>
</tr>
<tr>
<td>Base</td>
<td>MOT20</td>
<td>FRCNN</td>
<td>74.6</td>
<td>60.7</td>
<td>84.5</td>
</tr>
<tr>
<td>Ours</td>
<td>MOT20</td>
<td>FRCNN</td>
<td>75.2</td>
<td>61.0</td>
<td>84.4</td>
</tr>
<tr>
<td>Base</td>
<td>MOT20</td>
<td>CrowdDet</td>
<td>78.6</td>
<td>63.7</td>
<td>85.7</td>
</tr>
<tr>
<td>Ours</td>
<td>MOT20</td>
<td>CrowdDet</td>
<td>80.0</td>
<td>64.5</td>
<td>85.8</td>
</tr>
</tbody>
</table>

Table 2: Impact of standard linear interpolation (LI) and proposed camera motion-aware interpolation (CMAI) as well as height preservation (HP) in the motion model.
Table 4: State-of-the-art methods on MOT17 / 20 test set using private detections. Entries are sorted with ascending MOTA.

### MOT17

<table>
<thead>
<tr>
<th>Method</th>
<th>MOTA ↑</th>
<th>IDF1 ↑</th>
<th>HOTA ↑</th>
<th>MT ↑</th>
<th>ML ↓</th>
<th>FP ↓</th>
<th>FN ↓</th>
<th>IDSW ↓</th>
<th>FRAG ↓</th>
</tr>
</thead>
<tbody>
<tr>
<td>Semi-TCL [20]</td>
<td>73.3</td>
<td>73.2</td>
<td>59.8</td>
<td>972</td>
<td>441</td>
<td>22944</td>
<td>124980</td>
<td>2790</td>
<td>8010</td>
</tr>
<tr>
<td>FairMOT [56]</td>
<td>73.7</td>
<td>72.3</td>
<td>59.3</td>
<td>1017</td>
<td>408</td>
<td>27507</td>
<td>117477</td>
<td>3303</td>
<td>8073</td>
</tr>
<tr>
<td>PermaTrack [43]</td>
<td>73.8</td>
<td>68.9</td>
<td>55.5</td>
<td>1032</td>
<td>405</td>
<td>28998</td>
<td>115104</td>
<td>3699</td>
<td>6132</td>
</tr>
<tr>
<td>RelationTrack [54]</td>
<td>73.8</td>
<td>74.7</td>
<td>61.0</td>
<td>981</td>
<td>546</td>
<td>27999</td>
<td>118623</td>
<td>1374</td>
<td>2166</td>
</tr>
<tr>
<td>CSTrack [21]</td>
<td>74.9</td>
<td>72.6</td>
<td>59.3</td>
<td>978</td>
<td>411</td>
<td>23847</td>
<td>114303</td>
<td>3567</td>
<td>7668</td>
</tr>
<tr>
<td>CSTrack [21]</td>
<td>74.9</td>
<td>75.0</td>
<td>62.0</td>
<td>1170</td>
<td>444</td>
<td>32007</td>
<td>107616</td>
<td>1812</td>
<td>1824</td>
</tr>
<tr>
<td>TransTrack [41]</td>
<td>75.2</td>
<td>63.5</td>
<td>54.1</td>
<td>1302</td>
<td>240</td>
<td>50157</td>
<td>86442</td>
<td>3603</td>
<td>4872</td>
</tr>
<tr>
<td>TransTrack [41]</td>
<td>75.2</td>
<td>68.8</td>
<td>57.9</td>
<td>1203</td>
<td>321</td>
<td>32796</td>
<td>98475</td>
<td>3237</td>
<td>5686</td>
</tr>
<tr>
<td>MAATrack (ours)</td>
<td>79.4</td>
<td>75.9</td>
<td>62.0</td>
<td>1356</td>
<td>282</td>
<td>37320</td>
<td>77661</td>
<td>1452</td>
<td>2202</td>
</tr>
</tbody>
</table>

### MOT20

<table>
<thead>
<tr>
<th>Method</th>
<th>MOTA ↑</th>
<th>IDF1 ↑</th>
<th>HOTA ↑</th>
<th>MT ↑</th>
<th>ML ↓</th>
<th>FP ↓</th>
<th>FN ↓</th>
<th>IDSW ↓</th>
<th>FRAG ↓</th>
</tr>
</thead>
<tbody>
<tr>
<td>MLT [55]</td>
<td>48.9</td>
<td>54.6</td>
<td>43.2</td>
<td>384</td>
<td>274</td>
<td>45660</td>
<td>216803</td>
<td>2187</td>
<td>3067</td>
</tr>
<tr>
<td>TransCenter [52]</td>
<td>58.5</td>
<td>49.6</td>
<td>43.5</td>
<td>603</td>
<td>185</td>
<td>64217</td>
<td>146019</td>
<td>4695</td>
<td>9581</td>
</tr>
<tr>
<td>FairMOT [56]</td>
<td>61.8</td>
<td>67.3</td>
<td>54.6</td>
<td>855</td>
<td>94</td>
<td>103440</td>
<td>88901</td>
<td>5243</td>
<td>7874</td>
</tr>
<tr>
<td>TransTrack [41]</td>
<td>65.0</td>
<td>59.4</td>
<td>48.9</td>
<td>622</td>
<td>167</td>
<td>27191</td>
<td>150197</td>
<td>3608</td>
<td>11352</td>
</tr>
<tr>
<td>Semi-TCL [20]</td>
<td>65.2</td>
<td>70.1</td>
<td>55.3</td>
<td>761</td>
<td>131</td>
<td>61209</td>
<td>114709</td>
<td>4139</td>
<td>8508</td>
</tr>
<tr>
<td>LCC [58]</td>
<td>66.0</td>
<td>67.0</td>
<td>53.2</td>
<td>699</td>
<td>165</td>
<td>43938</td>
<td>129584</td>
<td>2237</td>
<td>4154</td>
</tr>
<tr>
<td>CSTrack [21]</td>
<td>66.6</td>
<td>68.6</td>
<td>54.0</td>
<td>626</td>
<td>192</td>
<td>25404</td>
<td>144358</td>
<td>3196</td>
<td>7632</td>
</tr>
<tr>
<td>GSDT [47]</td>
<td>67.1</td>
<td>67.5</td>
<td>53.6</td>
<td>660</td>
<td>164</td>
<td>31913</td>
<td>135409</td>
<td>3131</td>
<td>9875</td>
</tr>
<tr>
<td>RelationTrack [54]</td>
<td>67.2</td>
<td>70.5</td>
<td>56.5</td>
<td>773</td>
<td>111</td>
<td>61134</td>
<td>104597</td>
<td>4243</td>
<td>8236</td>
</tr>
<tr>
<td>MAATrack (ours)</td>
<td>73.9</td>
<td>71.2</td>
<td>57.3</td>
<td>741</td>
<td>153</td>
<td>24942</td>
<td>108744</td>
<td>1331</td>
<td>1450</td>
</tr>
</tbody>
</table>

we also apply a baseline tracker, that performs a standard association with the Hungarian method and uses a linear interpolation. One can see, that there are significant improvements w.r.t. the baseline among all detectors and on both datasets. The largest improvements are observed using detections from RetinaNet with gains of 3.3 (3.1), 2.3 (1.6), and 3.6 (3.1) points in terms of IDF1, HOTA, and MOTA, respectively, on MOT17 (MOT20). As expected, the tracking performance improves consistently by applying better detection models, whereby the overall best tracking results are achieved with the crowd-specific model CrowdDet. Note that, for example on MOT17, the detection results of the three models measured in average precision at an IoU threshold of 0.5 (AP$_{50}$) are 78.3 for RetinaNet, 86.5 for Faster RCNN, and 87.8 for CrowdDet.

4.5. Comparison with the State-of-the-Art

As the best results have been achieved in combination with CrowdDet, we keep it as detection model when applying our method on the test sets of MOT17 and MOT20. The results of our tracker termed MAATrack (Modelling Ambiguous Assignments), generated by the official evaluation server, are compared against the state-of-the-art in Table 4. MAATrack achieves the overall best tracking performance among all methods with large gains compared to the second best entry CorrTracker [45] / RelationTrack [54] in terms of MOTA (+2.9 / +6.7), IDF1 (+2.3 / +0.7), and HOTA (+1.3 / +0.8) on MOT17 / 20. Furthermore, the best values of MT and FN are obtained on MOT17. On MOT20, MAATrack has the least numbers of FP, IDS, and FRAG. The superior results show that a separate treatment of ambiguous assignments is beneficial for multi-target tracking, especially in crowds, as most tracking errors occur in such situations.

5. Conclusion

We develop a new association method for multi-target tracking in crowded scenes, which explicitly models ambiguous assignments of detections and tracks and treats those separately from clear track-detection matches. To resolve these ambiguous situations, different strategies are investigated. Moreover, we introduce two additional modules, a camera motion-aware interpolation technique and an adapted motion model, to further improve tracking performance. The effectiveness of our approach is shown with ablative experiments and state-of-the-art results are obtained on two popular benchmarks for multi-person tracking.
References


