Quo Vadis: Is Trajectory Forecasting the Key Towards Long-Term Multi-Object Tracking?

Anonymous Author(s)
Affiliation
Address
email

Abstract

Recent developments in monocular multi-object tracking have been very successful in tracking visible objects and bridging short occlusion gaps, mainly relying on data-driven appearance models. While we have significantly advanced short-term tracking performance, bridging longer occlusion gaps remains elusive: state-of-the-art object trackers bridge less than 10% of occlusions longer than three seconds. We suggest that the missing key is reasoning about future trajectories over a longer time horizon. Intuitively, the longer the occlusion gap, the larger the search space for possible associations. In this paper, we show that even a small yet diverse set of trajectory predictions for moving agents will significantly reduce this search space and thus improve *long-term tracking* robustness. Furthermore, we show that crucial components of our approach are reasoning in bird-eye space and generating a small yet diverse set of forecasts while accounting for their localization uncertainty. This way, we can advance state-of-the-art trackers on the MOTChallenge dataset and show we can significantly improve their long-term tracking performance.

1 Introduction

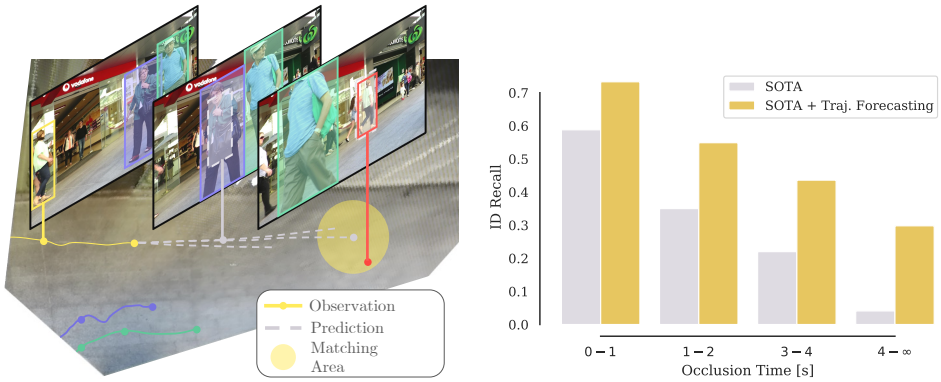
Multi-object tracking (MOT) is a long-standing research problem with applications ranging from real-time dynamic situational awareness for robot navigation [17, 13, 65, 35, 50, 9], traffic monitoring [58], studying animal behavior [41] and monitoring biological phenomena [3].

State-of-the-art MOT methods [64, 8, 4, 70, 63, 53] combine regression [64, 4] and combinatorial optimization [8] in conjunction with identity re-identification (ReID) models [25, 47, 8, 64, 54, 4] to track objects in the image space. Such approaches have been very successful for tracking visible objects and bridging *short-term* occlusions. However, as can be seen in Figure 1, *long-term tracking* remains an open challenge: state-of-the-art methods successfully bridge 50% of occlusions within one second, falling below 10% when the occlusion extends for more than 3 seconds. This is often not reflected in standard benchmarks [13, 17, 58, 65], as long-term occlusions are statistically rare.

ReID models, often in conjunction with simple motion models, have been immensely helpful [13] for short-term tracking. Nonetheless, as the occlusion becomes longer, one has to expand the search space since the object could have moved in any direction. This becomes even more challenging when tracking pedestrians since their motion is driven by non-observable factors such as goals, intent, or simply preferences. As the set of possible associations grows exponentially with the increasing gap length [44], the ability of ReID models to disambiguate between objects decreases. Continued efforts in the development of stronger appearance models will remain an important research direction in vision-based MOT. In this work, we propose an alternative: using long-term *trajectory forecasting* in order to prune down the combinatorial search space of feasible trajectory continuations.

As the main contribution of this paper, we carefully study *what is needed* to bridge the gap between multi-object tracking and trajectory forecasting. We have recently witnessed rapid progress in learning-based trajectory forecasting [45, 30, 19, 1]. However, these methods operate in fully-observed, metric bird-eye space, effectively disentangling the effect of the perspective projection on reasoning about motion. By contrast, monocular MOT methods only observe a projection of

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.



(a) Our method. (b) Association recall.

Figure 1: State-of-the-art methods for vision-based MOT can successfully track visible objects and bridge short occlusion gaps, however, fail at long-term tracking. (a) To bridge occlusion gaps, we lift monocular 2D detections to 3D world space, in which we reason about their possible future locations. This, in turn, allows us to re-connect detections that undergo long occlusions: as **yellow track** becomes occluded, our method predicts a small set of plausible future locations in 3D. In turn, we correctly associate **red detection** to the **yellow track** by accounting for the forecast uncertainty area. (b) As can be seen from the ratio of correct track association after different occlusion time lengths for the **prior work** and **our method**, this approach allows us to significantly improve long-term tracking capabilities and gap longer occlusion gaps. *Best seen in color.*

the visible portion of our 3D space. Our analysis reveals that we can bridge this gap by localizing trajectories in BEV-space, but crucially, the localization of 2D bounding boxes in BEV must be *temporally coherent*. We achieve this by estimating a single homography per sequence in a data-driven manner.

Forecasting methods can reason beyond simple linear extrapolations, predict multiple possible future outcomes, and account for social interactions. But are these all necessary ingredients for bridging complex and long-term occlusions? Our study suggests that the key ingredient is to estimate a set of forecasts that can cover possibly several diverging future paths with only a handful of samples and account for prediction uncertainty.

Our results suggest that our *tracking-by-forecasting* approach can be applied to improve the long-term tracking capabilities of existing object tracking methods. In particular, by applying our framework on top of the state-of-the-art method [68], we improve the performance on HOTA on MOT17 by 0.09pp and MOT20 by 0.10pp and further decrease the number of IDSW by 93 and 36, respectively. We hope that our conclusions will encourage the community to continue investigating how can the interplay between 3D reconstruction and trajectory forecasting can further improve long-term tracking.

We summarize our **main contributions** as follows: we (i) present a study on how we can reconcile two related fields of research on vision-based trajectory forecasting and monocular multi-object tracking. Our study reveals suggests that the core component of this interplay is temporally coherent reasoning about motion in 3D space. We (ii) utilize a synthetic MOT dataset to study how to localize objects in 3D BEV space in a manner that facilitates robust reasoning about plausible future motion and which are the core forecasting components as needed to bridge longer occlusion gaps; Finally, (iii) we demonstrate that we can generalize our conclusions from synthetic sandbox to a real-world monocular *MOTChallenge* sequences and demonstrate that our recipe can be used to improve long-term tracking performance for several object trackers.

2 Preliminaries

This section discusses the fundamentals of vision-based multi-object tracking and trajectory forecasting, the current state-of-the-art, and analyzes failure cases.

2.1 Multi-object Tracking

Monocular multi-object tracking (MOT) is the task of localizing objects as bounding boxes in image sequences and assigning them an identity-preserving unique ID. State-of-the-art methods decompose the problem into (i) object detection, and (ii) detection association.

Quantifying tracking errors. The *detection* aspect of the task is commonly quantified by counting per-frame detection errors over the sequence. To quantify *association* errors, we generally count two types of association errors we can encounter, namely *identity switches* (IDSW) (*i.e.*, wrong ID swaps, or re-initializing track with a different ID) and *identity transfers* (IDTR) (*i.e.*, incorrectly linking two different objects). While a successful association over occlusion gaps decreases the number of IDSW, a wrong association leads to an IDTR *and* IDSW when the ground truth object is then initialized with a new track. Recently introduced HOTA [32] separately evaluates object detection and temporal association aspects of this task. The *association accuracy* (AssA) is composed of the *association recall* (AssA) and *association precision* AssPr term. The AssRe accounts for IDSW errors, while the AssPr for IDTR errors.

Are all identity errors created equal? Correctly associating objects that undergo longer occlusion gaps is especially challenging, because the appearance and position of an object may drastically change. In MOT datasets [13, 17] the majority of occlusions are short occlusions ($\leq 2s$). Hence, solving rare long occlusions ($> 2s$) does not have a significant impact on the model performance, with the result that long-term tracking is commonly overlooked in the literature. This can be seen in Figure 1: state-of-the-art methods bridge less than 10% gaps beyond three-second-long occlusions.

Prior art. Early tracking methods focus on combinatorial optimization [67, 22, 29, 60, 38] and hand-crafting visual and motion-based descriptors [37, 26, 10], especially beneficial in the era of unreliable object detectors. State-of-the-art methods for monocular visual MOT are data-driven and largely rely on appearance. Regression-based methods [4, 64, 70] can localize objects even when object detections are missing, often used in conjunction with ReID models to bridge short occlusions. However, regression models fail, when an occluded person appears at a distant image position. For solving long-term occlusion, discrete optimization methods, combined with end-to-end learning based on graph neural networks [8, 59, 66] construct large graphs stretching over multiple seconds leading to high computational costs and complexity. Motion has always played an important role in visual tracking [5, 16, 4], especially beneficial in 3D where it is dis-entangled from projective distortion [27, 39, 21]. The interplay between reasoning in 3D for monocular pedestrian tracking based on linear motion was investigated in [23]. We hypothesize that bridging long-term gaps requires an understanding of the projection geometry and more sophisticated motion models that reason about plausible diverging future paths and non-linear motion.

2.2 Trajectory Forecasting

Problem formulation. Pedestrian trajectory forecasting has been studied independently to the closely-related task of object tracking. Forecasting is challenging, because (i) human behavior and, therefore, future motion underlies the effect of complex social and scene interactions, as well as latent navigating intent. Moreover, (ii) full scene geometry is usually not directly visible to the observer, and in general, it is difficult to precisely localize past trajectories. To this end, prior art ([42, 28, 30, 45]) studies forecasting in idealized conditions: given full, bird-eye view of the scene and perfectly-localized past trajectories, predict trajectory continuations in metric space.

Quantifying forecasting accuracy. Forecasting performance is measured in metric space as L_2 distance between the prediction and ground-truth trajectory (as final displacement error, FDE, or average displacement error, ADE) *wrt.* top-k forecasts (commonly, $k = 20$). We note that this approach mainly incentivizes high forecasting recall and neglects forecasting precision [11].

Prior Art. Early methods were deterministic, firstly based on physical models [20], and later on data-driven LSTM-based encoder-decoder networks [1] methods, focusing on modeling social [19, 1, 2] and scene [46, 24] interactions. As forecasting is inherently uncertain, and from several plausible forecasts we always only observe a single continuation, recent methods that learn a distribution of possible future trajectories using generative models [19, 46, 24, 2, 12, 11] have emerged as state-of-the-art prediction methods. Recent efforts have been explicitly focusing on conditioning forecasting on estimated pedestrian goal/intent [12, 34, 33] and estimating multi-modal posterior distributions [30, 11] that yield diverse trajectories that cover different plausible directions.

Can we bring the *two worlds* together, and if so, *how*? Furthermore, which of the aforementioned aspects (*i.e.*, stochasticity, non-linearity, multi-modality, diversity, accounting for interactions) are *important* in the context of multi-object tracking? These are the questions we aim to answer in the following sections.

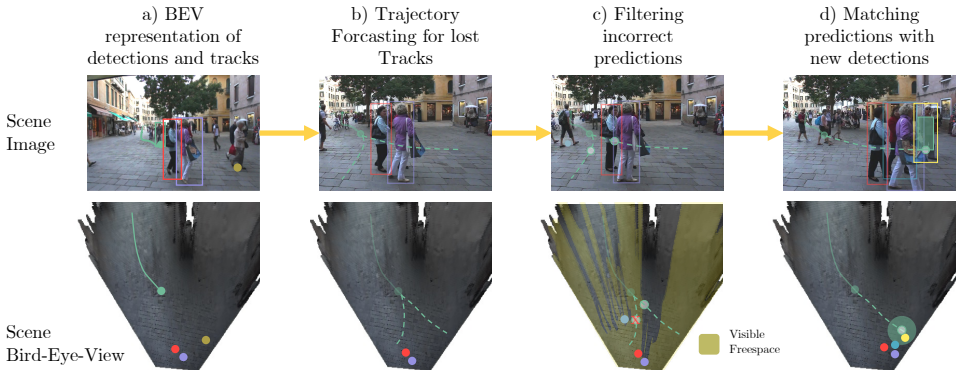


Figure 2: **Our method:** we bridge long-term occlusions by (a) localizing object tracks in BEV via estimated homography and (b) forecasting future trajectories for tracks that we cannot associate with new object detections. We (d) continually aim to match these *inactive* tracks with new object detections under a visibility constraint (c).

3 Methodology

In this section, we present our method for long-term multi-object tracking based on trajectory forecasting in bird-eye view (BEV) scene representation. Simply applying trajectory prediction to multi-object tracking is not trivially possible, as object trajectories observed in the image space break multiple assumptions of real-world trajectory prediction. While trajectory prediction works in top-down coordinates, motion and size of objects in image space depend on the intrinsic parameters, orientation, and position of the camera. In addition, we face temporal (limited length of observation), association (association errors along observation), and measurement (imprecise localization of objects) uncertainties of the trajectories. In object tracking objects are represented as bounding boxes in the image instead of single 2D positions. To bridge the gap between prediction and tracking, we assume objects move on a planar ground. Thus, the bottom-center points of detection bounding boxes $\mathbf{p}$ can be mapped to 2D BEV coordinate $\mathbf{x}$ via an (unknown!) homography transformation H that relates the homogeneous coordinates as $\mathbf{x} \propto H \cdot \mathbf{p}$.

Overview. Given a monocular video sequence, captured from a stationary camera from *arbitrary* viewpoint, we first estimate a homography for the whole sequence (Section 3.1). Then our model is incorporated into an online tracker where we consume the tracker output and (i) localize 2D detections in BEV representation (Figure 2*a*) Next, we (ii) run a trajectory forecasting network (described in Section 3.2) for lost tracks (Figure 2*b*). Finally, we integrate forecasts into the tracking (Section 3.3) pipeline accounting for the uncertainty in estimated forecasts, and matching of new detections to bridge short- and long-term occlusions (Figure 2*d*).

3.1 Data-driven Homography Estimation

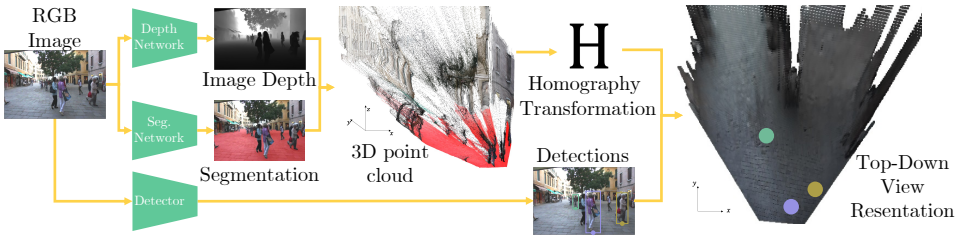


Figure 3: We estimate a sequence homography by reconstructing 3D point clouds, representing the scene, from a data-driven monocular depth estimator. We obtain ground image-to-point-cloud correspondences using a semantic segmentation model, that masks ground pixels, as needed to estimate the homography matrix. With the estimated homography matrix we transform the bottom points of bounding boxes to 2D BEV coordinates.

To bridge the gap between monocular object tracking and forecasting we first need to transform object detections and tracks from the image sequence into points and trajectories in bird-eye view (BEV) representation. Given a set of 2D object detections represented as bounding boxes localized

in the image plane, we aim to find a homography that maps their bottom-center positions to their corresponding 2D BEV coordinates.

In Figure 3 we outline our homography estimation method. We first train a monocular depth estimator [6] on synthetic dataset [15] to reconstruct a 3D point cloud (with estimated or known intrinsics) representing the scene in the first frame of the sequence. Then, we leverage the semantic segmentation network [62] to mask ground pixels.

We estimate the normal vector of the ground plane in 3D rotate the plane facing z direction. Then, we project ground points along the z-axis, leaving us with a pair-wise correspondence between ground pixels in the image and a 2D position in BEV, as needed to estimate the homography between the two planes. We stabilize the homography transformation for pixel positions close to the edge of the 2D image plane by linearizing the homography transformation.

We only compute the homography for the first frame of the sequence, and use it throughout the sequence, making our pedestrian localization robust to temporal fluctuations of the depth estimator.

For a detailed explanation, we refer to the supplementary material. In Section 4.1, we experimentally confirm that reasoning about future trajectories in 3D BEV rather than projective image space is crucial, and confirm that the proposed temporally-stable homography estimation is necessary to obtain reliable trajectory forecasts (as opposed to simply relying on monocular depth estimators).

3.2 Forecasting

Localization of object tracks in BEV enables us to leverage data-driven forecasting models beyond simple linear motion to reason for future trajectories. However, these models expect ground-truth fixed-size past trajectory observations, while our trajectories are noisy and of varying length. As discussed in Section 2.2, prediction models are optimized for metrics that incentivize a large number of predictions and minimize L_2 distance to ground-truth trajectories. It is thus unclear how different proposed concepts translate into real-world tracking scenarios. We, therefore, identify the main design patterns proposed in the forecasting community and verify their impact *directly* in the context of *forecasting to track*.

Preprocessing. Forecasting models encode trajectories using LSTM encoder-decoder [1] architecture, which takes a fixed-size observed trajectory as input and predicts a future trajectory. We construct input trajectories out of temporally consecutive detections of the same track ID, localized in BEV. To account for localization noise, we smooth the noisy observations using the Kalman filter and linearly extrapolate trajectories into the past to get trajectories of the required fixed-size input length.

Forecasting design patterns. A dominant trend in trajectory prediction is (i) stochastic, (ii) socially aware, (iii) and multi-modal trajectory prediction. In our experimental setup, we build on the LSTM encoder-decoder architecture [1] and include the following key design patterns that recently emerged in the forecasting community.

Stochasticity. Stochastic trajectory predictors enable us to sample multiple plausible future trajectories to account for the uncertainty in the future positions. We follow the approach by [19] and learn a generative model using GAN [18] and best-of-many [7] loss. As a result, the network internally learns an observation conditioned distribution of future trajectories, from which we can sample.

Social Interactions. Social interactions impact future motion: pedestrians adapt their trajectories on-the-fly to avoid collisions. Several methods [19, 1, 2, 24] account for interactions in the forecasting models. These methods leverage pooling [19], attention [46], or graph neural networks [24] to provide social context (*i.e.*, trajectories of surrounding agents) with the trajectory decoder. To answer the question of whether modeling social interactions matters for tracking, we implement Social GAN (S-GAN) [19] which uses pair-wise interaction features between neighboring pedestrians which are max-pooled and passed to the decoder.

Multimodality and Diversity. While these generative models learn a distribution over trajectories, they need to sample a very large number of trajectories to cover all modes present in the scene, as learning a multi-modal posterior with a single GAN is difficult [52]. To cover the main modes of the scene and cover them with as few samples as possible, we implement a multi-generator GAN network [11], that extends the presented GAN architecture by training multiple decoder heads, where each decoder learns to focus on a particular model. We get a set of plausible but maximally separated predictions by sampling from these different generators.

3.3 Forecasting for Tracking

We assume we have an online object tracker, capable of tracking visible objects (*e.g.*, bounding box regression-based tracker [4]). As long as tracks are constantly updated with new detections, we consider them *active* $\mathcal{S}_A$. Once we cannot associate a detection, a track becomes *inactive*. We keep such tracks in a memory-bank $\mathcal{S}_I$. For each frame t the tracker outputs a set of tracks $\mathcal{O} = (o_1, \ldots, o_M)$ with $o_i = (\texttt{ID}_i, b_i,)$ where $\texttt{ID}$ represents the track identity and b the pixel space bounding box as shown in Figure 2(a). We use the detections and project them into a BEV cooridnates x using our estimated homography H.

Quo Vadis? If an object track becomes inactive (*i.e.*, temporally lost), we move the active track into the memory bank and predict k trajectories of length τ_{max} in BEV space using the trajectory forecasting model as described in Section 3.2. As long as the track is inactive and not yet matched, we move along the predicted trajectory do not predict a new trajectory in each frame.

Filtering and removing predictions. *No prediction can live forever.* When we use stochastic trajectory predictors with multiple predictions, we need to limit the number of inaccurate or obsolete predictions to decrease the chance of false re-association. In practice, we limit a lifetime of a prediction to a maximal lifetime of τ_{max} and try to filter out *unlikely* forecasts. We use spatial and social context to determine the *freespace* [23] in which objects should be visible to the camera. We assume that visible objects eventually are detected and, therefore, remove prediction branches that should be visible for more than τ_{vis} frames.

We consider an object as *visible* if the object is not occluded by the scene or other pedestrians. In particular, this means that the predicted BEV position lies in an area of the project ground-mask (shown in Figure 2 c) and has no bounding box overlap ≥ 0.25 with any other object detection, closer to the camera. Relative order can be determined based on bottom bounding box coordinates foramodal detections. If all predictions from an inactive track are removed even before τ_{max} we also remove the entire track.

Matching predictions with new detection. Given trajectory forecasts, we aim to match them with newly-detected objects via bi-partite matching, employing the Hungarian algorithm, following the standard practice in tracking [69, 57, 31, 5]. This boils down to computing association costs c_{ij} between a prediction of an inactive track i and new un-associated detections j as follows:

$$c_{ij} = (\Delta_{\text{IoU}} + \max\left(\tau_{L_2} - \Delta_{L_2}, 0\right)) \cdot (\Delta_{\text{App}} \geq \tau_{App} \text{ and } \Delta_{\text{IoU}} \geq \tau_{iou}), \tag{1}$$

where Δ_{IoU} is the IoU score between the two bounding boxes, Δ_{L_2} is the euclidean distance of the prediction and detection in BEV, and Δ_{App} the cosine distance of visual features f. We obtain the visual features from a pre-trained CNN by passing the image appearances of the last occurrence of the inactive track and the new detection. τ_{L_2}, τ_{iou}, and τ_{App} denote thresholds for the matching respectively. Therefore we determine association in BEV metric space, as well as image domain based on IoU bounding box overlap between the forecast and detected box.

Only taking spatial terms into the account significantly reduces the number of ID switches (IDSW), this approach may lead to ID transfer errors (IDTR). While forecasting does significantly narrow down combinatorial search space for associations, in practice, it is still beneficial to verify potential associations with an appearance model. To this end, Δ_{App} evaluated as similarity of the visual features measured by cosine distances and, for objects close to the camera, a required minimal IoU. score τ_{iou} accounting for correct object sizes. Note, that these appearance features are not used to construct the cost function but rather to filter-out matches, incompatible in terms of appearance.

Purely relying on spatial distance in real space and IoU leads to high recall of detection but also to IDTRs, especially in crowded scenes with many new detections close to each other. To prevent an increase of IDTR errors, we threshold the matching by a required maximal similarity of the visual features measured by cosine distances and minimum IoU for close objects. Note, that these appearance features are not directly used to construct the cost function but rather to filter out wrong matches. In essence, we obtain a pre-selection of potential matching candidates by using the trajectory forecast and filter those if the appearance drastically deviates between the last observation and new detection.

4 Experimental Evaluation

In this section, we first discuss our evaluation test-bed, followed by an experimental study on bird-eye-view (BEV) trajectory reconstruction (Section 4.1). Then, we analyze different design patterns in the field of forecasting, applied to the domain of object tracking in BEV space, and discuss the relevance

of different model modules for tracking (Section 4.2). Afterwards, we discuss how our approach can be used to improve several vision-based MOT methods on static sequences and justify our design decisions. Finally, we show that our forecasting model can be used to establish new state-of-the-art on real-world MOT17 and MOT20 datasets (Section 4.4). For visualization of our tracking method we refer the reader the suupplementary material.

Datasets. We evaluate our trajectory prediction framework on different pedestrian tracking datasets, namely synthetic MOTSynth [15] and two real-world MOT17 and MOT20 datasets [13]. *MOTSynth* is a large synthetic dataset for multi-object tracking. It provides 764 diverse sequences with various viewpoints and lighting and weather conditions. Importantly, it provides ground-truth depth information and 3D keypoints for pedestrians, which allows us to carefully study the impact of scene representation on trajectory localization. *MOTChallenge* MOT17 [36] and MOT20 [14] are real-world tracking datasets, commonly used to evaluate pedestrian tracking models. We use these datasets to evaluate our method on real-world recordings. For our experiments, we utilize the commonly used split of the MOT17 training set, where the first half of each sequence is used for training, and the second half for the evaluation [71, 68, 61].

Metrics. For trajectory prediction, we report the L_2 final displacement error (FDE) for top-k predictions for $2s$ and $4s$ prediction horizons (see Section 2.2).

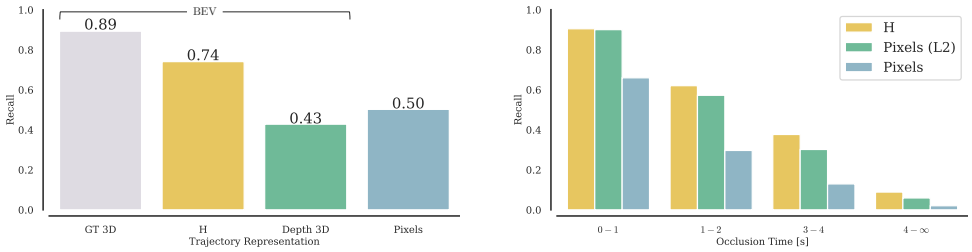
For multi-object tracking evaluation, we report higher-order tracking accuracy (HOTA) [32], with focus on the association aspect of the task. To this end, we also report AssA, AssPr,as well as number of ID switches, IDSWs. Additionally, we report IDSW when the tracker loses an object and re-initiates a new track for the same object when it re-appears. We call these errors as ID^{lost}. For metric discussion, we refer to Section 2.1.

Metrics labeled with either S or L only consider prediction or occlusion lengths shorter or longer $2s$, respectively

For all experiments, we use the following parameters for the mathching of detections with tracks: $\tau_{L_2} = 2.5m, \tau_{app} = 0.8, \tau_{iou} = 0.2$ The maximal lifetime of prediction is $\tau_{max} = 6s$ and maximal visibility $\tau_{vis} = 1s$ before it is removed. For further information on implementation details, we refer the reader to the Supplementary Material.

Object Trackers We study and ablate our method on 7 high ranked state-of-the art trackers of MOTChallenge and refer to them as *baseline*. We use BYTE [68], JDE [57], CSTrack [71], FairMOT [69], TraDes [55], QDTrack [40], CenterTrack [70], Chained-Tracker [43] and TransTrack [51] for the evaluation on the MOT17 validation set and BYTE and CenterTrack on the MOT20 training dataset. These trackers either use ReID similarity and/or simple motion cues to bridge short-term occlusions. In the Supplementary, we show the detailed performance of our method applied to each of these trackers.

4.1 Bird Eye View Estimation



(a) Overall recall (BEV and pixel-space). (b) Recall *wrt.* different occlusion lengths.

Figure 4: Endpoint matching recall of predictions and GT trajectories using a linear motion predictor. For predictions in bird-eye-view also project the prediction back to the image. Prediction is matched when $\Delta_{\text{IoU}} > 0.5$ or Δ_{L_2} distance < $2m$. The model *Pixel (L2)* predicts motion in pixel space and projects end point into BEV for matching.

In this section, we discuss the suitability of different scene representations for forecasting. We work with static sequences of the *MOTSynth* dataset (that provides depth maps, used for evaluating and training a monocular depth estimator) and evaluate the ability of a linear motion model to gap occlusions of different lengths for tracks, obtained by using [70]. We evaluate the ratio of successful

Table 1: Which forecasting modules matter for tracking? Evaluated on MOT17 validation set.

Model	Nr. Samples	Deter - ministic	Stochastic	Social	Multi-modal	Prediction		Tracking					
						$\text{FDE}_S \downarrow$	$\text{FDE}_L \downarrow$	HOTA ↑	AssA ↑	AssRe ↑	AssPr ↑	$\text{ID}_S^{\text{lost}} \downarrow$	$\text{ID}_L^{\text{lost}} \downarrow$
Baseline						–	–	50.71	46.87	51.80	**78.11**	0 %	0 %
Static	1	✓				1.59	2.09	53.84	53.51	60.04	72.95	-14.77 %	-8.40 %
Kalman Filter (pixel)	1	✓				–	–	54.08	54.02	60.45	72.81	**-22.37** %	-8.99 %
Kalman Filter	1	✓				0.69	1.23	54.11	54.04	60.75	71.73	-19.50 %	-16.07 %
GAN	3		✓			0.85	1.26	54.43	54.61	61.11	73.21	-17.99 %	-8.64 %
GAN	20		✓			**0.65**	**0.99**	53.81	53.40	60.45	71.31	-18.03 %	-15.63 %
S-GAN	3		✓	✓		0.87	1.21	**54.52**	54.78	61.22	73.28	-16.92 %	-8.57 %
MG-GAN	3		✓		✓	0.67	1.03	**54.52**	**54.80**	**61.35**	73.13	-21.19 %	**-17.43** %

matches between the target and the linear prediction and count a match to be successful if the IoU of the predicted bounding box is larger than 0.5 or the L_2 distance in metric space is lower than $2m$. We get the predicted bounding by translating the last observed bounding box by the predicted displacement in the image.

Baselines. We compare motion in (i) BEV and (ii) pixel space. We evaluate different approaches to localize trajectories: (a) using ground truth 3D coordinates orthographically projected to BEV (oracle), (b) the proposed homography estimation as described in Section 3.1, and (c) directly using learned monocular depth estimates and resulting point clouds, followed by orthogonal projection of points representing an object (those are determined to form the segmentation mask).

Conclusions. As can be seen in Figure 4, GT 3D (oracle) based motion estimates solve 89.3% of the gaps, pointing that the motion in the synthetic dataset is dominantly linear. Our proposed data-driven homography estimation approach only drops by 15% compared to using ground-truth 3D keypoints. By contrast, estimating linear motion in pixels space only solves 50.2% gaps. Using per-frame monocular depth estimates resolves only 0.43% of gaps, due to the fact that depth estimates are not temporally stable and are thus not useful for forecasting methods. As can be seen in Figure 4b, this trend is especially apparent for longer occlusion gaps. We conclude that our proposed homography transformation is suitable for forecasting.

4.2 Trajectory Prediction Models

In this section, we evaluate different forecasting components (as discussed in section 3.2), comparing them to the constant-velocity model (Kalman filter) in BEV and pixel space, and identity (static) model, a stochastic GAN predictor generating $k = 3$ and $k = 20$ samples, as well as Social GAN and the multi-modal MG-GAN in BEV space. We report the average final displacement error (FDE), and predict over $2s$ (S) and $4s$ (L) horizons.

Forecasting. We observe that the linear model performs well for short-term window (0.69 FDE_S), suggesting that linear motion is suitable for short occlusions. We also do not see a significant difference between GAN w/o social module (S-GAN). While FDE error suggests vanilla GAN ($k = 20$) yields the best results (0.65 FDE_S and 0.99 FDE_L), tracking results suggest otherwise.

Tracking. To investigate the effect on long-term occlusion focus on the change of ID_L^{lost} for short ($t_{occl} \leq 2s$) and long ($t_{occl} > 2s$) occlusion gaps. As can be seen, even the static motion model solves 8.4% (ID_L^{lost}), as many occluded objects do not move. By modeling linear motion (Kalman filter in pixel space) we can improve on short-term re-association, improving for 0.59 (long-term IDSW) over the static model.

We focus the discussion on long occlusion gaps. In terms of the generative model, we observe that interaction-aware S-GAN ($\downarrow 8.64\%$ ID_L^{lost}) is on-par with vanilla GAN ($\downarrow 8.57\%$ ID_L^{lost}) for $k = 3$; interestingly, both are below linear Kalman filter (BEV) performance, suggesting that these models suffer from low precision. Only MG-GAN, explicitly trained to generate multimodal trajectories, outperforms the linear model ($\downarrow 17.43\%$ ID_L^{lost}), and significantly outperforms vanilla GAN with only three samples. These conclusions generalize to tracking metrics.

4.3 Tracking Evaluation

In this section, we study the impact of forecasting models on the tracking performance on the validation set. First, we discuss the impact of different design decisions on matching strategy, as explained in section 3.3. We provide detailed result tables and discussion in the supplementary, and here summarize key conclusions.

Trajectory matching. We find that a combination of L_2 and IoU leads to the highest decrease in ID^{lost}, and yields overall highest $AssRe$, however, at the cost of decreasing $AssPr$ (69.27). We conclude, that purely via geometric matching, we can significantly reduce IDSW, at the cost of

Table 2: We improve tracking results of *all* top-7 state-of-the-art models (MOT17 validation set and MOT20 training set). Difference to the baseline performance are shown in $(\cdot)$.

	MOT17 (val, static scenes)							MOT20 (train)	
	BYTE [68]	CenterTrack [70]	CSTrack [71]	FairMOT [69]	JDE [57]	TraDeS [55]	TransTrack [51]	BYTE [68]	CenterTrack [70]
HOTA	71.36 (+0.21)	61.78 (+3.56)	61.60 (+0.43)	58.42 (+0.09)	51.06 (+0.20)	62.45 (+0.67)	60.68 (-0.23)	56.85 (+0.06)	32.71 (+0.62)
AssA	73.96 (+0.49)	66.18 (+7.54)	63.84 (+0.8)	59.21 (+0.37)	54.36 (+0.45)	67.41 (+1.6)	63.47 (-0.49)	53.97 (+0.20)	28.94 (+1.34)
IDSW	84 (-3)	137 (-146)	269 (-28)	198 (-12)	316 (-19)	106 (-32)	112 (-3)	1815 (-78)	5240 (-2700)

Table 3: Comparison under the "private detector" protocol on MOT17 test set.

Tracker	HOTA	IDF1	MOTA	IDSW	AssA
ReMOT [49]	59.73	71.99	77.01	2853	57.08
CrowdTrack [56]	60.26	73.62	75.61	2544	59.26
TLR [69]	60.72	73.58	76.48	3369	58.88
MAA [48]	61.98	75.88	79.36	1452	60.16
ByteTrack [68]	63.05	77.30	80.25	2196	61.97
QuoVadis (Ours)	**63.14**	**77.71**	**80.27**	**2103**	**62.07**

Table 4: Comparison under the "private detector" protocol on MOT20 test set.

Tracker	HOTA	IDF1	MOTA	IDSW	AssA
FairMOT [69]	54.42	68.44	59.57	1881	56.6
CrowdTrack [56]	54.95	68.24	70.68	3198	52.57
MAA [48]	57.28	71.15	73.90	1331	55.14
ReMOT [49]	61.15	73.14	77.42	1789	58.68
ByteTrack [68]	61.34	75.20	77.76	1223	59.55
QuoVadis (Ours)	**61.48**	**75.70**	**77.77**	**1187**	**59.87**

increasing IDTR errors. Adding appearance-based threshold (Section 3.3) provides the best trade-off, and overall highest $AssA$ (54.29) and overall HOTA score (54.27), while still recovering 21% of lost trajectories. A more detailed table different matching metrics can be found in the Suppl.

Validation results. In Table 2 we present the performance of different state-of-the-art trackers on the static sequences of MOT17-val and MOT20-train (trained on MOT17), equipped with our trajectory forecasting model. We find that our model brings stable improvements over all the key metrics: HOTA, $AssA$, and IDSW. We find that our trajectory prediction model reduces IDSW for all models.

4.4 Benchmark Evaluation

In this section, we demonstrate that our method can be applied to state-of-the-art tracker ByteTrack [68] and improve its tracking capabilities, establishing new state-of-the-art on *MOTChallenge* MOT17&MOT20 benchmarks. We evaluate our method in the *private detection* regime, as these trackers use private detectors.

We compare our *QuoVadis* to base tracker ByteTrack[68] and compare both to MOT17 benchmark published top-perfomers. As shown in table 3, we improve performance on key metrics overall top methods. Particularly, we reduce the number of identity switches by 93 compared to [68] and establish a new state-of-the-art in terms of HOTA (63.14). We observe similar trends on MOT20, where we improve over the base tracker ByteTrack [68] by $+0.5$ in IDF1 and reduce the number of identity switches by 36, establishing a new state-of-the-art $(61.48$ HOTA).

5 Limitations, Broader Impact, and Conclusion

We presented the first attempt to bridge the gap between real-world trajectory prediction and single-camera tracking. Throughout our study, we identify necessary hurdles and present the first solutions. Our main conclusion is that long-term tracking can be improved via trajectory forecasting in BEV space. We suggest that the key component is a forecasting approach can reason about multiple feasible future directions with a small set of forecasts.

As for now, we encounter the following limitations of our work. The homography transformation depends on the quality of depth estimates. Furthermore, the presented models do not yet implicitly account for the uncertainty of the BEV position and correct association of observation. This shows the need for developing trajectory forecasting models explicitly dealing with these uncertainties. Our motion models presented in this work do not yet directly address localization uncertainty positions and association error of observations. Instead, we partially account for these uncertainties via preprocessing of input trajectories via smoothing but argue that forecasting models should implicitly deal with uncertainties. Finally, matching trajectory predictions with new detection is essential and requires a more sophisticated solution than thresholding different matching scores. Ideally, the trajectory forecasting and matching is not done on top of an online tracking output but rather directly end-to-end incorporated into the tracking model.

We hope that our work helps to improve tracking methods that enable autonomous platforms to safely navigate and interacts with humans.

References

[1] Alexandre Alahi, Kratarth Goel, Vignesh Ramanathan, Alexandre Robicquet, Li Fei-Fei, and Silvio Savarese. Social lstm: Human trajectory prediction in crowded spaces. In *CVPR*, June 2016.

[2] Javad Amirian, Jean-Bernard Hayet, and Julien Pettré. Social Ways: Learning multi-modal distributions of pedestrian trajectories with GANs. In *CVPR Workshops*, 2019.

[3] Samreen Anjum and Danna Gurari. Ctmc: Cell tracking with mitosis detection dataset challenge. In *CVPR Workshops*, 2020.

[4] Philipp Bergmann, Tim Meinhardt, and Laura Leal-Taixé. Tracking without bells and whistles. In *ICCV*, 2019.

[5] Alex Bewley, Zongyuan Ge, Lionel Ott, Fabio Ramos, and Ben Upcroft. Simple online and realtime tracking. In *ICIP*, 2016.

[6] Shariq Farooq Bhat, Ibraheem Alhashim, and Peter Wonka. AdaBins: Depth estimation using adaptive bins. In *CVPR*, 2021.

[7] Apratim Bhattacharyya, Bernt Schiele, and Mario Fritz. Accurate and diverse sampling of sequences based on a "Best of Many" sample objective. In *CVPR*, 2018.

[8] Guillem Braso and Laura Leal-Taixe. Learning a neural solver for multiple object tracking. In *CVPR*, 2020.

[9] Holger Caesar, Varun Bankiti, Alex H. Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuScenes: A multimodal dataset for autonomous driving. In *CVPR*, 2020.

[10] Wongun Choi. Near-online multi-target tracking with aggregated local flow descriptor. In *ICCV*, 2015.

[11] Patrick Dendorfer, Sven Elflein, and Laura Leal-Taixé. MG-GAN: A multi-generator model preventing out-of-distribution samples in pedestrian trajectory prediction. In *ICCV*, 2021.

[12] Patrick Dendorfer, Aljoša Ošep, and Laura Leal-Taixé. Goal-GAN: Multimodal Trajectory Prediction Based on Goal Position Estimation. In *ACCV*, 2020.

[13] Patrick Dendorfer, Aljoša Ošep, Anton Milan, Konrad Schindler, Daniel Cremers, Ian Reid, and Stefan Roth Laura Leal-Taixé. Motchallenge: A benchmark for single-camera multiple target tracking. *IJCV*, 2020.

[14] Patrick Dendorfer, Hamid Rezatofighi, Anton Milan, Javen Shi, Daniel Cremers, Ian Reid, Stefan Roth, Konrad Schindler, and Laura Leal-Taixé. MOT20: A benchmark for multi object tracking in crowded scenes, 2020.

[15] Matteo Fabbri, Guillem Brasó, Gianluca Maugeri, Orcun Cetintas, Riccardo Gasparini, Aljoša Ošep, Simone Calderara, Laura Leal-Taixé, and Rita Cucchiara. Motsynth: How can synthetic data help pedestrian detection and tracking? In *CVPR*, 2021.

[16] Andreas Geiger, Martin Lauer, Christian Wojek, Christoph Stiller, and Raquel Urtasun. 3d traffic scene understanding from movable platforms. *PAMI*, 2014.

[17] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the KITTI vision benchmark suite. In *CVPR*, 2012.

[18] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NeurIPS*, 2014.

[19] Agrim Gupta, Justin Johnson, Li Fei-Fei, Silvio Savarese, and Alexandre Alahi. Social gan: Socially acceptable trajectories with generative adversarial networks. In *CVPR*, 2018.

[20] Dirk Helbing and Péter Molnár. Social force model for pedestrian dynamics. *Physical Review E*, 1995.

[21] Hou-Ning Hu, Qi-Zhi Cai, Dequan Wang, Ji Lin, Min Sun, Philipp Krahenbuhl, Trevor Darrell, and Fisher Yu. Joint monocular 3d vehicle detection and tracking. *arXiv preprint arXiv:1811.10742*, 2018.

[22] Chang Huang, Bo Wu, and Ramakant Nevatia. Robust object tracking by hierarchical association of detection responses. In *ECCV*, 2008.

[23] Tarasha Khurana, Achal Dave, and Deva Ramanan. Detecting invisible people. In *ICCV*, 2021.

[24] Vineet Kosaraju, Amir Sadeghian, Roberto Martín-Martín, Ian Reid, Hamid Rezatofighi, and Silvio Savarese. Social-BiGAT: Multimodal trajectory forecasting using Bicycle-GAN and graph attention networks. In *Neural Information Processing Systems*, 2019.

[25] Laura Leal-Taixé, Cristian Canton-Ferrer, and Konrad Schindler. Learning by tracking: Siamese cnn for robust target association. *CVPR Workshops*, 2016.

[26] Laura Leal-Taixé, Michele Fenzi, Alina Kuznetsova, Bodo Rosenhahn, and Silvio Savarese. Learning an image-based motion context for multiple people tracking. In *CVPR*, 2014.

[27] Bastian Leibe, Konrad Schindler, Nico Cornelis, and Luc Van Gool. Coupled object detection and tracking from static cameras and moving vehicles. *PAMI*, 2008.

[28] Alon Lerner, Yiorgos Chrysanthou, and Dani Lischinski. Crowds by Example. *Comput. Graph. Forum*, 2007.

[29] Yuan Li, Chang Huang, and Ramkat Nevatia. Learning to associate: Hybridboosted multi-target tracker for crowded scene. In *CVPR*, 2009.

[30] Junwei Liang, Lu Jiang, Kevin Murphy, Ting Yu, and Alexander Hauptmann. The Garden of Forking Paths: Towards Multi-Future Trajectory Prediction. In *CVPR*, 2020.

[31] Qiankun Liu, Qi Chu, Bin Liu, and Nenghai Yu. Gsm: Graph similarity model for multi-object tracking. In *IJCAI*, 2020.

[32] Jonathon Luiten, Aljoša Ošep, Patrick Dendorfer, Philip Torr, Andreas Geiger, Laura Leal-Taixé, and Bastian Leibe. HOTA: A higher order metric for evaluating multi-object tracking. *IJCV*, 2020.

[33] Karttikeya Mangalam, Yang An, Harshayu Girase, and Jitendra Malik. From goals, waypoints & paths to long term human trajectory forecasting. In *ICCV*, 2021.

[34] Karttikeya Mangalam, Harshayu Girase, Shreyas Agarwal, Kuan-Hui Lee, Ehsan Adeli, Jitendra Malik, and Adrien Gaidon. It is not the journey but the destination: Endpoint conditioned trajectory prediction. In *ECCV*, 2020.

[35] Roberto Martin-Martin*, Mihir Patel*, Hamid Rezatofighi*, Abhijeet Shenoi, JunYoung Gwak, Eric Frankel, Amir Sadeghian, and Silvio Savarese. JRDB: A dataset and benchmark of egocentric robot visual perception of humans in built environments. *PAMI*, 2021.

[36] Anton Milan, Laura Leal-Taixe, Ian Reid, Stefan Roth, and Konrad Schindler. MOT16: A benchmark for multi-object tracking, 2016.

[37] Anton Milan, Stefan Roth, and Konrad Schindler. Continuous energy minimization for multitarget tracking. *PAMI*, 2014.

[38] Anton Milan, Konrad Schindler, and Stefan Roth. Detection- and trajectory-level exclusion in multiple object tracking. In *CVPR*, 2013.

[39] Aljoša Ošep, Wolfgang Mehner, Markus Mathias, and Bastian Leibe. Combined image- and world-space tracking in traffic scenes. In *ICRA*, 2017.

[40] Jiangmiao Pang, Linlu Qiu, Xia Li, Haofeng Chen, Qi Li, Trevor Darrell, and Fisher Yu. Quasi-dense similarity learning for multiple object tracking. In *CVPR*, 2020.

[41] Malte Pedersen, Joakim Bruslund Haurum, Stefan Hein Bengtson, and Thomas B Moeslund. 3d-zef: A 3d zebrafish tracking benchmark dataset. In *CVPR*, 2020.

[42] S. Pellegrini, Andreas Ess, and L. Gool. Improving data association by joint modeling of pedestrian trajectories and groupings. In *ECCV*, 2010.

[43] Jinlong Peng, Changan Wang, Fangbin Wan, Yang Wu, Yabiao Wang, Ying Tai, Chengjie Wang, Jilin Li, Feiyue Huang, and Yanwei Fu. Chained-tracker: Chaining paired attentive regression results for end-to-end joint multiple-object detection and tracking. In *ECCV*, 2020.

[44] Donald B Reid. An algorithm for tracking multiple targets. *TAC*, 1979.

[45] Alexandre Robicquet, Amir Sadeghian, Alexandre Alahi, and Silvio Savarese. Learning social etiquette: Human trajectory understanding in crowded scenes. In *ECCV*, 2016.

[46] Amir Sadeghian, Vineet Kosaraju, Ali Sadeghian, Noriaki Hirose, Hamid Rezatofighi, and Silvio Savarese. Sophie: An attentive GAN for predicting paths compliant to social and physical constraints. In *CVPR*, 2019.

[47] Jeany Son, Mooyeol Baek, Minsu Cho, and Bohyung Han. Multi-object tracking with quadruplet convolutional neural networks. In *CVPR*, 2017.

[48] Daniel Stadler and Jürgen Beyerer. Modelling ambiguous assignments for multi-person tracking in crowds. In *WACV*, 2022.

[49] Daniel Stadler and Jürgen Beyerer. On the performance of crowd-specific detectors in multi-pedestrian tracking. In *AVSS*, 2021.

[50] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, et al. Scalability in perception for autonomous driving: Waymo open dataset. In *CVPR*, 2020.

[51] Peize Sun, Jinkun Cao, Yi Jiang, Rufeng Zhang, Enze Xie, Zehuan Yuan, Changhu Wang, and Ping Luo. Transtrack: Multiple-object tracking with transformer. *arXiv preprint arXiv: 2012.15460*, 2020.

[52] Ugo Tanielian, Thibaut Issenhuth, Elvis Dohmatob, and Jeremie Mary. Learning disconnected manifolds: a no gans land. In *Proceedings of Machine Learning and Systems*, 2020.

[53] Pavel Tokmakov, Jie Li, Wolfram Burgard, and Adrien Gaidon. Learning to track with object permanence. In *CVPR*, 2021.

[54] Paul Voigtlaender, Michael Krause, Aljosa Osep, Jonathon Luiten, B.B.G Sekar, Andreas Geiger, and Bastian Leibe. MOTS: Multi-object tracking and segmentation. In *CVPR*, 2019.

[55] Weiyao Wang, Matt Feiszli, Heng Wang, and Du Tran. Unidentified video objects: A benchmark for dense, open-world segmentation. *arXiv preprint arXiv:2104.04691*, 2021.

[56] Yongxin Wang, Kris Kitani, and Xinshuo Weng. Joint object detection and multi-object tracking with graph neural networks, 2020.

[57] Zhongdao Wang, Liang Zheng, Yixuan Liu, Yali Li, and Shengjin Wang. Towards real-time multi-object tracking. In *ECCV*, 2020.

[58] Longyin Wen, Dawei Du, Zhaowei Cai, Zhen Lei, Ming-Ching Chang, Honggang Qi, Jongwoo Lim, Ming-Hsuan Yang, and Siwei Lyu. Ua-detrac: A new benchmark and protocol for multi-object detection and tracking. *arXiv preprint arXiv:1511.04136*, 2015.

[59] Xinshuo Weng, Yongxin Wang, Yunze Man, and Kris Kitani. Gnn3dmot: Graph neural network for 3d multi-object tracking with multi-feature learning. In *CVPR*, 2020.

[60] Bo Wu and Ramkat Nevatia. Detection and segmentation of multiple, partially occluded objects by grouping, merging, assigning part detection responses. *IJCV*, 2009.

[61] Jialian Wu, Jiale Cao, Liangchen Song, Yu Wang, Ming Yang, and Junsong Yuan. Track to detect and segment: An online multi-object tracker. In *CVPR*, 2021.

[62] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. `https://github.com/facebookresearch/detectron2`, 2019.

[63] Yihong Xu, Yutong Ban, Guillaume Delorme, Chuang Gan, Daniela Rus, and Xavier Alameda-Pineda. Transcenter: Transformers with dense queries for multiple-object tracking. *arXiv preprint arXiv:2103.15145*, 2021.

[64] Yihong Xu, Aljoša Ošep, Yutong Ban, Radu Horaud, Laura Leal-Taixé, and Xavier Alameda-Pineda. How to train your deep multi-object tracker. In *CVPR*, 2020.

[65] Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In *CVPR*, 2020.

[66] Jan-Nico Zaech, Alexander Liniger, Dengxin Dai, Martin Danelljan, and Luc Van Gool. Learnable online graph representations for 3d multi-object tracking. 2022.

[67] Li Zhang, Li Yuan, and Ramakant Nevatia. Global data association for multi-object tracking using network flows. In *CVPR*, 2008.

[68] Yifu Zhang, Peize Sun, Yi Jiang, Dongdong Yu, Fucheng Weng, Zehuan Yuan, Ping Luo, Wenyu Liu, and Xinggang Wang. ByteTrack: Multi-object tracking by associating every detection box, 2021.

[69] Yifu Zhang, Chunyu Wang, Xinggang Wang, Wenjun Zeng, and Wenyu Liu. FairMOT: On the fairness of detection and re-identification in multiple object tracking. *International Journal of Computer Vision*, 2021.

[70] Xingyi Zhou, Vladlen Koltun, and Philipp Krähenbühl. Tracking objects as points. In *ECCV*, 2020.

[71] Zijie Zhuang, Longhui Wei, Lingxi Xie, Tianyu Zhang, Hengheng Zhang, Haozhe Wu, Haizhou Ai, and Qi Tian. Rethinking the distribution gap of person re-identification with camera-based batch normalization. In *ECCV*, 2020.

Checklist

The checklist follows the references. Please read the checklist guidelines carefully for information on how to answer these questions. For each question, change the default **[TODO]** to [Yes] , [No] , or [N/A] . You are strongly encouraged to include a **justification to your answer**, either by referencing the appropriate section of your paper or providing a brief inline description. For example:

- Did you include the license to the code and datasets? [Yes] See Section **??**.
- Did you include the license to the code and datasets? [No] The code and the data are proprietary.
- Did you include the license to the code and datasets? [N/A]

Please do not modify the questions and only use the provided macros for your answers. Note that the Checklist section does not count towards the page limit. In your paper, please delete this instructions block and only keep the Checklist section heading above along with the questions/answers below.

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] See 5
   (c) Did you discuss any potential negative societal impacts of your work? [N/A]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] See Supplementary.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Supplementary.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] See 4
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See Supplementary.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] See 3.1
   (b) Did you mention the license of the assets? [No] The dataset is freely available for research purposes
   (c) Did you include any new assets either in the supplemental material or as a URL? [No] No new assets were introduced
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [Yes]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [Yes]

5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]