Monocular 3D detection and reID-enhanced tracking of multiple traffic participants

anonymous submission

Keywords: Monocular 3d detection, Multi-target tracking, Target Re-ID, KITTI tracking benchmark

Abstract: Autonomous driving is becoming a major scientific challenge and applied domain of significant impact, also triggering a demand for the enhanced safety of vulnerable road users, such as cyclists and pedestrians. The recent developments in Deep Learning have demonstrated that monocular 3D pose estimation is a potential detection modality in safety related task domains such as perception for autonomous driving and automated traffic monitoring. Deep Learning offers enhanced ways to represent targets in terms of their location, shape, appearance and motion. Learning can capture the significant variations seen in the training data while retaining class- or target-specific cues. Learning even allows for discovering specific correlations within an image of a 3D scene, as a perspective image contains many hints about an object's 3D location, orientation, size and identity. In this paper we propose an attention-based representational enhancement to enhance the spatial accuracy of 3d pose and the temporal stability of multi-target tracking. The presented methodology is evaluated on the KITTI multi-target tracking benchmark. It demonstrates competitive results against other recent techniques, and when compared to a baseline relying solely on a Kalman-Filter-based kinematic association step.

1 INTRODUCTION

Spatial awareness and reasoning are fundamental traits of modern vision-based robotic systems. However, monocular (single view) vision-based perception is associated with ambiguities such as depth-scale ambiguity or viewpoint invariance. These ambiguities arise from projecting the 3D world onto a 2D imaging plane, where multiple 3D scene configurations can result in the same projected image if scale information is *a priori* not known. Many ambiguities associated with a single view can be mitigated if the views correspond to a street-level observer observing common object types with more-or-less known dimensions. In such cases, the learning task can be formulated such that a 2D image content can be regressed to a set of 3d object locations on a ground plane with an estimated heading orientation.

Image-based 3D object detection and pose parameter regression are typical multi-task learning problems as they require classifying image content into classes while also regressing their 3D bounding boxes. Association of detected objects to consistent motion trajectories can be facilitated by including a re-identification task (*reID*), which distils each object's appearance into a compact and discriminative feature set (Wang et al., 2020).

This work proposes a representation-enhanced end-to-end Deep Learning approach for 3D pose-aware multiple-object detection and tracking, using only monocular RGB images as input. Its representational concept is based on an encoder-decoder type multi-task learning scheme while also integrating recent representational breakthroughs devised explicitly for coping with spatial ambiguities and association uncertainties. The tracking integrates a *reID* approach that utilizes a Transformer Encoder (Vaswani et al., 2017) with deformable attention (Zhu et al., 2020) to obtain target-specific appearance features using a spatially delocalized exploration and correlation scheme. The proposed methodology is evaluated for the multi-target tracking task employing the KITTI benchmarking scheme and compared to several recent competing algorithmic concepts.

The paper is structured as follows: in Section 2 we describe related works. Section 3 presents the proposed methodology, which is evaluated and discussed in Section 4. Finally, Section 5 concludes the paper.

2 RELATED WORK

This section provides a brief overview on relevant state-of-the-art approaches of monocular 3D object detection and tracking. First, basic concepts of 2D object detection schemes are presented. Next, based on the described core concepts, prevailing representational extensions towards monocular 3D detection and parameter regression are characterized.

Object Detection: Object detection frameworks can be categorized by their architectural aspects. Accordingly, two-stage, single-stage and anchorless detectors can be distinguished.

Two-Stage Detectors: As the earliest detection concepts, two-stage detectors first generate region proposals and then classify them as either one of the predefined object categories or background. Classified regions of interest (ROIs) can be further refined, and additional attributes can be predicted. The earliest model using this approach is the Region-based convolutional neural network (R-CNN) (Girshick et al., 2014), followed by the Fast R-CNN (Girshick, 2015) and Faster R-CNN (Ren et al., 2017) models.

Single-Stage Detectors: Considerations of computational simplification have led to single-stage detectors, which combine the two steps into a single step and view object detection as a regression problem. This concept was introduced by YOLO (Redmon et al., 2016) which divides the image into an $S \times S$ grid and predicts for each cell class probabilities. Combining both stages into one yielded an improved run-time performance while retaining competitive accuracy. Its successor versions (Redmon and Farhadi, 2017; Redmon and Farhadi, 2018) still achieve state-of-the-art results.

Anchorless Detectors: Both, R-CNN and YOLO are anchor-based detectors, as they operate with a fixed number of region candidates for predicting objects and refining their delineations. Anchorless detectors on the other hand do not use such pre-defined boxes. Instead, they formulate box parameters (such as the center or corners) as key-points and regress corresponding bounding box parameters directly. Among the first models adopting this approach were CornerNet (Law and Deng, 2018) and CenterNet (Zhou et al., 2019). In these works, network output yields a down-sampled dense grid and predictions of relevant object points at each cell. This simple inference scheme is complemented by additional object attribute estimators. CenterNet achieved state-of-the-art results while significantly reducing run-time.

Monocular 3D Object Detection: Common 3D object detection schemes predict 3D object attributes on a common ground plane, hence extending the 2D bounding box representation. The correlation between the image space and 3D world space is learned end-to-end from annotated training data. In the next section, a concise overview on different representations used for monocular 3D detection is given.

Concepts: Image formation can be commonly approximated by a linear pinhole camera model, projecting points from a 3D space onto a 2D image plane (Figure 1). Thereby, camera-centric world coordinates are commonly used (Szeliski, 2022) to define the spatial relation with respect to an object. Given a set of image-space coordinates for an object, the reconstruction of the corresponding 3D coordinates requires the depth (distance) of the object location. This information, however, is not contained in a monocular image and therefore needs to be directly estimated via a learned model. Additionally, there is an ambiguity between object size and depth, as there is no way of telling only from the image whether an object is small or just far away and vice versa.

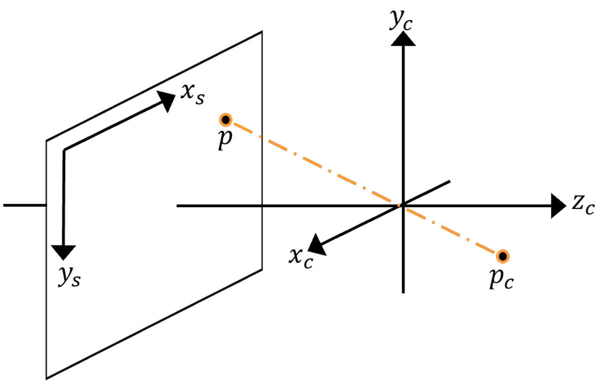

Figure 1: Projection of a 3D camera-centered point p_c onto the sensor plane at p. $[x_c, y_c, z_c]$ are the camera coordinate system, $[x_s, y_s]$ the image coordinate system.

Representations: Early neural approaches for monocular 3D object detection used a two-stage approach containing a template matching step (Chabot et al., 2017). Image content within detected objects were matched with templates, leading to inherent limitations regarding the number of template models. Other approaches transform the street-level front-view image into another data spaces, such as a birds-eye-view (BEV) (Kim and Kum, 2019; Roddick et al., 2018; Srivastava et al., 2019) or a pseudo point cloud (Weng and Kitani, 2019); the latter computed via a monocular depth estimation network (Godard et al., 2017). In the BEV space, the task is reduced to an oriented 2D bounding box detection, while the pseudo point cloud representation enables the use of off-the-shelf LiDAR detection schemes.

Our proposed method also follows the strategy of directly regressing spatial and class-specific object properties. Typical schemes adopting the regression concept are CenterNet, regressing 2D bounding box parameters and SS3D (Jörgensen et al., 2019), which additionally regresses distance, dimensions, observation angle and projected 3D bounding box corners.

Multi-Target Tracking: Multi-target tracking aims to associate and partition time-consecutive detection responses such that each partition belongs to the same target identity. Its computational scheme can be online or off-line, depending on whether an access only to the most recent, or also to all previous observations is given. In the followings typical location-, motion- and appearance-based cues are described which support the underlying association task:

Location- and Motion-based Methods: Physical constraints governing object motion typically define strong criteria which aid association. A common approach to treat tracking as a filtering process, where association (Kuhn, 1955) and prediction steps (e.g., by a Kalman Filter (Kalman et al., 1960)) alternate during the tracking process. Modern variants, such as SORT (Bewley et al., 2016) and AB3DMOT (Weng et al., 2020) employ a similar concept. Motion-based approaches tend to have issues in crowded scenes and in presence of low framerates, where location/motion-correlation of targets degrades.

Appearance-Based Methods: The abstraction capability of neural representations offers powerful means to capture the appearance of targets and use it in an association step. Architectural concepts range from a sequential multi-stage (detect, encode and associate) to parallel, multi-branch (Voigtlaender et al., 2019) approaches. Such feature-based similarities or embedding can again be used in a conventional association step. Contrary to motion, appearance-based methods tend to be robust to detection gaps and larger inter-frame displacements.

Since object detection and tracking are mutually supporting intermediate processing steps, it is intuitive to formulate them as a jointly trained multi-task problem. In our work we adopt a tight coupling between these tasks, by using a common backbone for feature computation, followed by task-specific subnetworks. Our proposed scheme devises mutually supporting representations, which attempt to simultaneously meet criteria of 2d projective, 3D BEV and target specificity constraints.

3 METHODS

In this section, we describe the proposed monocular 3D multi-target object detection and tracking framework, including the proposed enhancements regarding the computational backbone network and the reID branch. Additionally, the loss functions used for the optimization process are detailed.

Monocular 3D Multi-Target Detection & Tracking: The aim of the proposed detection and tracking network is to simultaneously predict the 3D location, dimension, orientation, and ID of objects given a sequence of monocular RGB images as input. The overall network architecture is extended from CenterNet, which utilizes object centers to identify objects and detection responses are obtained via predicting a confidence heatmap. Several regression heads are used to obtain the desired 3D bounding box of each object. A Transformer Encoder network calculates an embedding for each object that aids in identifying each unique object instance across a sequence. The overall architecture is illustrated in Figure 2.

Attention Enhanced Backbone: The proposed framework uses the hierarchical layer fusion network DLA-34 (Yu et al., 2018) as its computational backbone network. The hierarchical aggregation connections are replaced by deformable convolution (Zhu et al., 2019) layers as in CenterNet. To further improve the ability to consider long-range dependencies within the image, the deformable convolution layers have been enhanced with an additional attention layer that only utilizes the information contained in the key as the query content and the relative position are already covered by the deformable convolution (Zhu et al., 2019).

Object Detection and Representation: The 3D information of an object is encoded via seven parameters: $[x, y, z, h, w, l, \alpha]$. Here, x, y and z are the 3D location of the object center, h, w and l represent the object dimensions and α denotes the apparent yaw angle of the object on the ground plane. Object centers are predicted using the position on the heatmap combined with a regressed offset. To obtain more accurate predictions in cases where objects are located near the image border, the decoupled representation in combination with the edge fusion module proposed by (Zhang et al., 2021a) was also incorporated into the model. In addition to the location prediction from the heatmap, the offset due to discretization is predicted as well. The network predicts the apparent yaw angle directly. The depth is predicted using two approaches, mutually supporting an accurate depth estimation: On one hand, the depth is directly regressed. On the other hand, the corner points of the 3D bounding box are utilized to validate a second depth prediction. This is achieved by using the relative proportion between pixel height and estimated object height. Given the cameras focal length, the depth of the vertical line from a top corner to the corresponding bottom corner of the bounding box can be calculated as:

$$z_l = \frac{f \times H}{h_l} \tag{1}$$

where f is the focal length, H the predicted object height, h_l the pixel height of the vertical edge of the bounding box and z_l the resulting depth. The weighted sum of the calculated depths is the final prediction. Here, the weights are given by the inverse of an uncertainty prediction, that the model regresses for

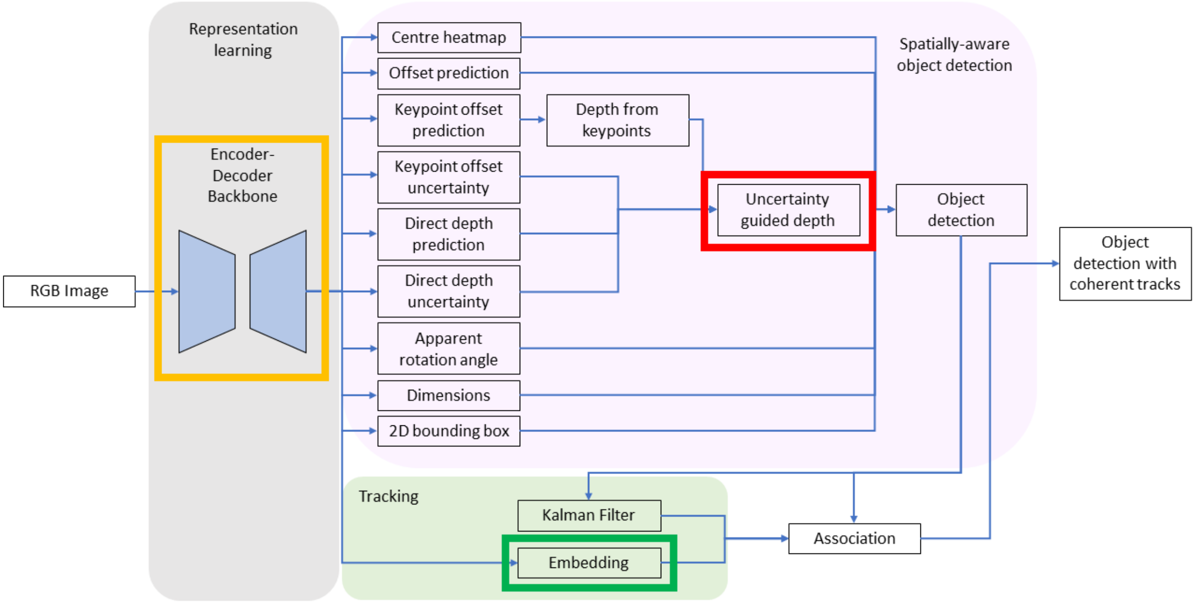

Figure 2: Illustration of the overall network architecture. Highlighted boxes emphasize the proposed novelties: yellow represents the attention-enhanced backbone, red illustrates the Robust Kullback-Leibler loss and green the Transformer Encoder reID subnetwork.

each depth.

ReID and Tracking: The proposed method follows a joint detection and embedding approach and incorporates a reID branch atop of the backbone feature extractor, like FairMOT (Zhang et al., 2021c). However, instead of using a Convolutional Neural Network (CNN) as a subnetwork to extract the embeddings, a deformable Transformer Encoder (Zhu et al., 2020) is used to capture long-range relationships between the extracted features. The extracted features are enriched with a positional embedding to preserve spatial relations. Three separate linear layers generate sampling offsets, attention weights and features values. The sampling offsets are used to obtain key sample values, which are then multiplied by the attention weights and their aggregation is performed. A final linear layer generates the outputs. The distances between the generated embeddings are used to calculate a cost matrix between existing tracks and new detection responses. Assignment is then accomplished using the Hungarian algorithm (Kuhn, 1955).

Loss Functions A weighted sum of multiple loss functions is used to train the proposed framework. In this section, the individual loss components are described.

Like in (Zhou et al., 2019), the penalty-reduced focal loss is used for object center estimation. It is defined as:

$$f = \frac{-1}{N}\sum(1-\hat{Y}_{xyc})^{\alpha}\log(\hat{Y}_{xyc}) \tag{2}$$

$$g = \frac{-1}{N}\sum(1-Y_{xyc})^{\beta}(\hat{Y}_{xyc})^{\alpha}\log(1-\hat{Y}_{xyc}) \tag{3}$$

$$L_{\text{centre}} = \begin{cases} f \text{ if } Y_{xyc} = 1 \\ g \text{ otherwise} \end{cases} \tag{4}$$

, where α and β are hyperparameters which define the degree of penalty reduction and down-weighting of easy examples. For all experiments, $\alpha = 2$ and $\beta = 4$ are used.

To compensate for the discretization, spatial offsets are learned and optimized as in (Zhang et al., 2021a):

$$L_{\text{offset}} = \begin{cases} |\hat{\delta}_{inside} - \delta_{inside}| & \text{if inside} \\ \log(1 + |\hat{\delta}_{outside} - \delta_{outside}|) & \text{otherwise} \end{cases} \tag{5}$$

To obtain the orientation, the apparent angle $\hat{\alpha}$ is directly regressed and clamped to $[-\pi, \pi]$. It is optimized using an L2 loss as follows:

$$L_{\text{angle}} = (\sin(\hat{\alpha}) - \sin(\alpha))^2 + (\cos(\hat{\alpha}) - \cos(\alpha))^2 \tag{6}$$

with α being the ground truth angle.

The dimensions are regressed as offsets $\hat{\delta}_k$ deviating from the class average, using the L1 loss:

$$L_{\text{dimension}} = \sum_{k \in \{h,w,l\}} |\bar{k}_c e^{\hat{\delta_k}} - k| \tag{7}$$

The corner points of the projected 3D bounding box are regressed as offsets from the discretized object center and optimized using L1 loss. However, only those corner points that are visible in the image are penalized in the loss function, which is indicated by $I_{\text{inside}}(k_i)$. The loss function becomes:

$$L_{\text{keypoints}} = \frac{\sum_{i=0}^{N_k} I_{\text{inside}}(k_i) |\hat{\delta_{ki}} - \delta_{ki}|}{\sum_{i=0}^{N_k} I_{inside}(k_i)} \tag{8}$$

Additionally including a loss function for a 2D bounding box regression task has been shown to also improve the 3D detection performance (Zhang et al., 2021a). Therefore, the 2D bounding box estimation task is also added, which predicts spatial offsets from the object center. The generalized Intersection-over-Union loss ($GIoU$) is used, yielding a 2D bounding box loss L_{2D}:

$$GIoU = IoU - \frac{A^c - U}{A^c} \tag{9}$$

$$L_{\text{2D}} = 1 - GIoU \tag{10}$$

For the depth estimation, instead of the Laplacian Kullback-Leibler loss (such as in MonoFLEX), we use the Robust Kullback-Leibler loss, proposed by (Chen et al., 2021), to overcome the issue of an increasing gradient during training due to the reduced uncertainty. It is defined as:

$$L_{\text{robust KL}} = \frac{1}{\hat{w}} \begin{cases} \frac{1}{2} e^2 + \log\sigma & |e| <= \sqrt{2} \\ \sqrt{2}|e| - 1 + \log\sigma & |e| > \sqrt{2} \end{cases} \tag{11}$$

$$\hat{w} \leftarrow \alpha\hat{w} + (1-\alpha)\frac{1}{N}\sum_{i=0}^{N}\frac{1}{\sigma_i} \tag{12}$$

where $e = |\hat{y} - y|/\sigma$ is the L1 error of the prediction, σ denotes the predicted uncertainty in the estimation, and N is the number of predictions made. α is a hyperparameter that determines the impact of new observations on the exponential moving average of the inverse of the uncertainties $\hat{w}$.

Finally, the eight corner points $\hat{v}_i$ of the 3D bounding box which are compared in terms of a 3D spatial deviation to the corresponding ground truth 3D corner points. The resulting spatial discrepancy gives rise to an L1 loss, whose optimization enforces a strong depth-based criterion, thus supporting different subtasks:

$$L_{\text{bounding box}} = \sum_{i=0}^{7} |\hat{v}_i - v_i| \tag{13}$$

The reID branch is treated as a classification during training, where the generated embeddings are the input to a linear layer that outputs a probability for every unique object instance K in the training dataset. To optimize this task, the cross-entropy loss is used:

$$L_{\text{reID}} = -\sum_{i=0}^{N}\sum_{k=0}^{K} L^i(k)\log(p(k)) \tag{14}$$

with $p(k)$ being the predicted probability that the detection is object k, $L^i(k)$ the one-hot encoded representation of the ground truth label and N the total number of detection responses in the image.

4 RESULTS & DISCUSSION

This section presents training details and obtained experimental results of the proposed framework. Based on quantitative and qualitative results for the detection and tracking quality, we discuss the observed algorithmic qualities and encountered failure modes. Finally, the impact of the proposed additions is demonstrated in an ablation study.

4.1 Training & Evaluation Metrics

For all experiments, the model was trained on the KITTI dataset (Geiger et al., 2012), using the training (5,027 images) / validation (2,981 images) split proposed by (Voigtlaender et al., 2019). The input images were padded to 384×1280 px and AdamW was used as the optimizer with an initial learning rate of 3×10^{-4}, decaying by a factor of 10 at epochs 80 and 90. Overall, the network was trained for 100 epochs. The input data was augmented using random horizontal flips. Each of the prediction heads consists of two convolutional layers with a batch norm and a ReLU activation in between. Training involves three object categories: passenger cars, cyclists, and pedestrians.

Training of a multi-task regression network naturally faces the complexity of a specific data need. Joint optimization for detection and tracking in a 3D space requires datasets, which provide at the same time category labels, 3D spatial annotations and tracking information. The employed dataset provides this information, however, more data and/or increased diversity would probably lead to an even better accuracy. For fair comparison, all experiments and comparisons use the same data and data split.

For object detection, the average precision on the car class for the three different difficulty levels (easy, medium, hard as defined by the KITTI benchmark) is reported (see Table 1). The IoU threshold used is 0.7.

Method	AP_{3D}		
	Easy	**Moderate**	**Hard**
SMOKE (Liu et al., 2020)	14.76	12.85	11.50
MonoGeo (Zhang et al., 2021b)	18.45	14.48	12.87
Ground-aware Monocular 3D Obj. Det. (Liu et al., 2021)	23.63	16.16	12.06
MonoFlex (Zhang et al., 2021a)	23.64	17.51	14.83
Proposed Method	**20.56**	**15.00**	**11.79**

Table 1: The monocular 3D object detection results of the proposed method compared to state of the art methods.

The main evaluation metric for tracking used is the higher order tracking accuracy (*HOTA*) (Luiten et al., 2021). Additionally, the detection accuracy (*DetA*) and the association accuracy (*AssA*) are reported to better grasp the contributions of the two components. The three metrics are defined as follows:

$$\text{HOTA}_\alpha = \sqrt{\frac{\sum_{c\in\{\text{TP}\}} A(c)}{|\text{TP}|+|\text{FN}|+|\text{FP}|}} \tag{15}$$

$$A(c) = \frac{|\text{TPA}(c)|}{|\text{TPA}(c)|+|\text{FNA}(c)|+|\text{FPA(c)}|} \tag{16}$$

$$\text{DetA}_\alpha = \frac{|\text{TP}|}{|\text{TP}|+|\text{FN}|+|\text{FP}|} \tag{17}$$

$$\text{AssA}_\alpha = \frac{1}{|\text{TP}|} \sum_{c\in\{TP\}} A(c) \tag{18}$$

α is the confidence threshold at which detections are counted as positive. TP, FN and FP are true positive, false negative and false positive detections respectively, while TPA(c), FNA(c) and FPA(c) denote the true positive, false negative and false positive associations of a true positive detection. c are the elements of the set of true positive detections. The final metric is approximated by averaging over 19 different thresholds instead of using the integral:

$$\text{HOTA} = \approx \frac{1}{19} \sum_{\alpha\in\{0.05,\ldots,0.95\}} \text{HOTA}_\alpha \tag{19}$$

$$\text{DetA} = \approx \frac{1}{19} \sum_{\alpha\in\{0.05,\ldots,0.95\}} \text{DetA}_\alpha \tag{20}$$

$$\text{AssA} = \approx \frac{1}{19} \sum_{\alpha\in\{0.05,\ldots,0.95\}} \text{AssA}_\alpha \tag{21}$$

4.2 Quantitative Results

Table 1 presents a detection accuracy comparison in terms of *AP* scores to some recent competing approaches. Scores in this table originate from an *IoU* evaluation within a metric 3D space, therefore slight regression errors in depth, location or orientation quickly lead to a strong decay in the computed score. As it can be seen from the table, the obtained detection accuracy is comparable to the current state-of-the art. For hard examples represent the proposed scheme exhibits some weakness, mainly due to objects which are small-sized and near the image border, where enforcing depth constraints is less effective.

Table 2 compares the tracking results on the car class to a baseline method, which solely relies on a track association based on *IoU* with predictions from a Kalman filter. The results show that the proposed method is an improvement over the baseline, especially regarding the *AssA* criterion, where the difference was the most pronounced. We attribute this improvement to the introduced ReID representation which generates an improvement especially for vehicles, which are in the view periphery, far from the image center. In this region, upon translatory motion or in case of a turning camera (observer vehicle), motion-based association quickly degrades and ReID feature mitigate this problem to a certain extent.

Method	HOTA	DetA	AssA
Baseline (Kalman Filter)	30.86	22.79	42.68
Proposed Method	**30.96**	**22.89**	**42.81**

Table 2: The monocular 3D multi-target tracking results of the proposed method compared to the Kalman Baseline.

4.3 Qualitative Results

In Figure 3, qualitative results on the KITTI validation dataset are shown. Additionally, in Figure 4 the proposed method is compared qualitatively to the Kalman Filter baseline. As it can be seen from these figures, a situation of a rapid change in the underlying motion characteristics of the observed vehicle leads to association problems. The Kalman Filter, as it is based on purely kinematic attributes, struggles with such sudden changes in motion. In contrast, the reID network manages to correctly match the detection responses and establish consistent trajectories.

4.4 Ablation Study

To demonstrate the impact of the various proposed improvements, Table 3 shows the results obtained

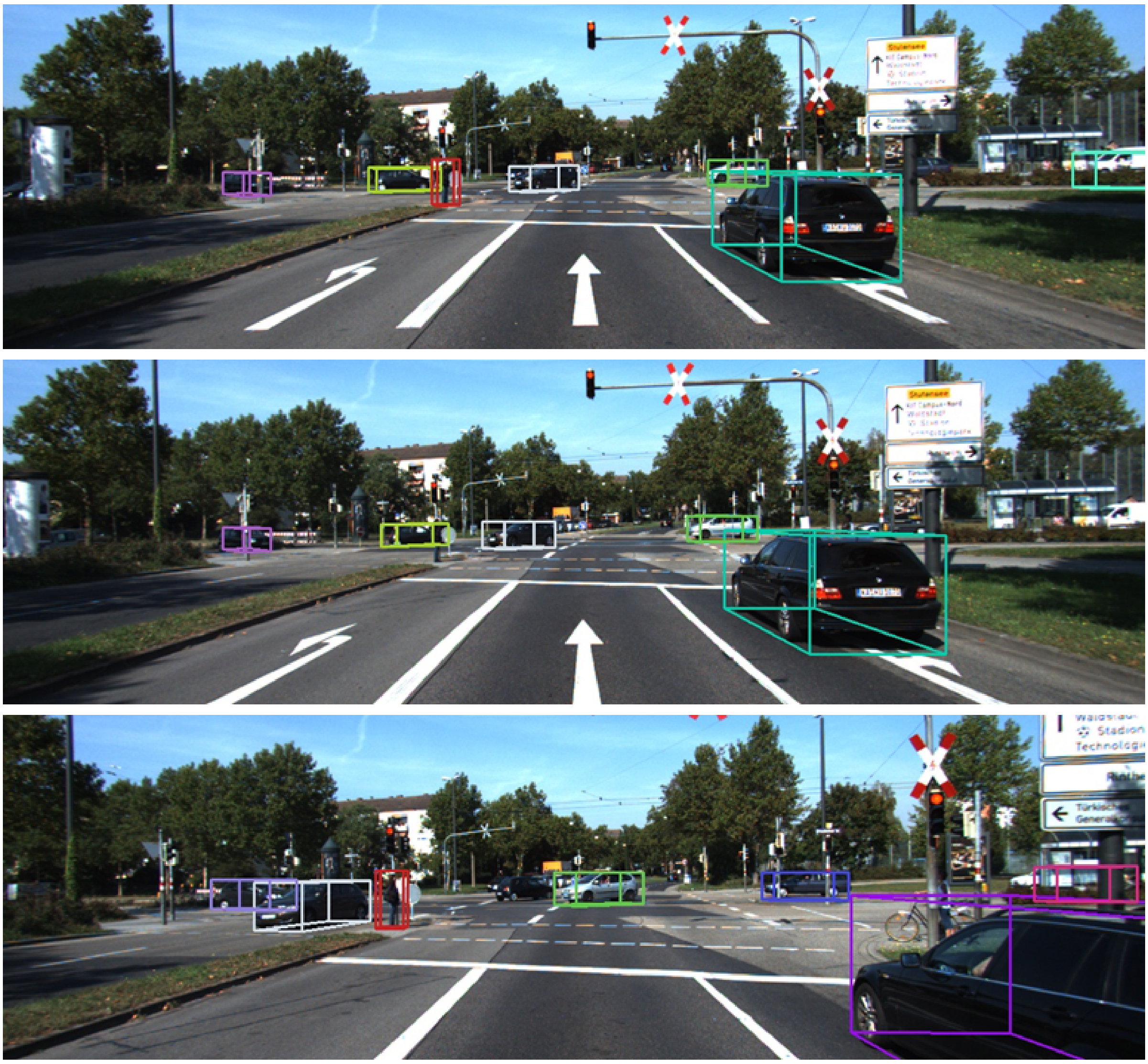

Figure 3: Qualitative tracking results of the proposed method on the KITTI validation set. Same coloured bounding boxes denote the same track identity assigned.

Method	**HOTA**	**DetA**	**AssA**
Without attention in backbone	30.04	21.88	42.34
Without Robust KL loss	29.57	21.23	42.57
With vanilla CNNs instead of Transformer Encoder	30.25	21.77	43.5
Full model	**30.96**	**22.89**	**42.81**

Table 3: The impact of the proposed changes to the 3D multi-target tracking results.

without the individual changes. All experiments were conducted on the KITTI validation set and to combat run-to-run variance, the results have been averaged over three runs. As one can see, the additional attention in the backbone feature extractor showed the most impact in the *DetA* metric. The use of the robust Kullback-Leibler loss led to the greatest overall improvement and affected the *DetA* the most. Finally, the standard CNNs had a better *AssA* than the proposed method but due to their negative impact on the DetA, the overall *HOTA* was still lower compared to the Transformer Encoder reID network.

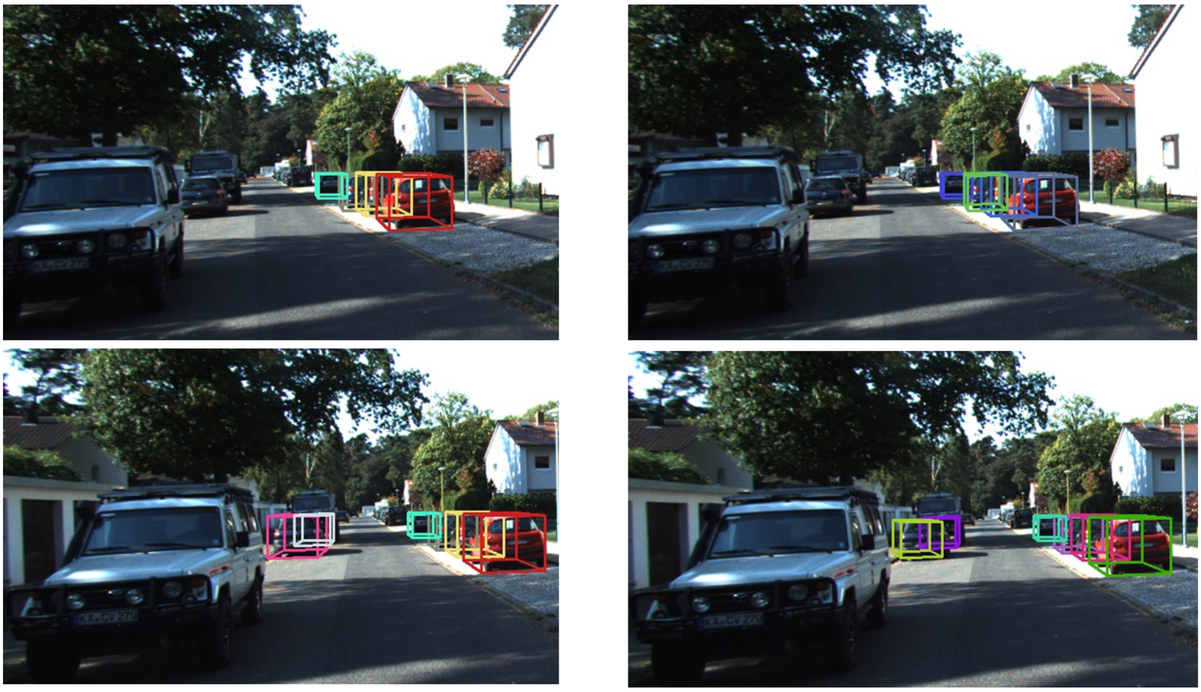

Figure 4: Comparison between proposed reID approach (left) and simple Kalman Filter (right) on the KITTI validation set.

5 Conclusion

In this work, we proposed a representation-enhanced end-to-end Deep Learning approach for 3D multi-target detection and tracking, that utilizes a Transformer Encoder sub-network to extract representative reID features. This approach facilitates an improved tracking performance compared to a motion-based baseline on the KITTI benchmark dataset. Regarding monocular 3D object detection, the proposed method is competitive with current SOTA models.

REFERENCES

Bewley, A., Ge, Z., Ott, L., Ramos, F., and Upcroft, B. (2016). Simple online and realtime tracking. In *IEEE International Conference in Image Processing (ICIP)*, pages 3464–3468.

Chabot, F., Chaouch, M., Rabarisoa, J., Teuliere, C., and Chateau, T. (2017). Deep manta: A coarse-to-fine many-task network for joint 2d and 3d vehicle analysis from monocular image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2040–2049.

Chen, H., Huang, Y., Tian, W., Gao, Z., and Xiong, L. (2021). Monorun: Monocular 3d object detection by reconstruction and uncertainty propagation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10379–10388.

Geiger, A., Lenz, P., and Urtasun, R. (2012). Are we ready for autonomous driving? the kitti vision benchmark suite. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3354–3361.

Girshick, R. (2015). Fast r-cnn. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1440–1448.

Girshick, R., Donahue, J., Darrell, T., and Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 580–587.

Godard, C., Mac Aodha, O., and Brostow, G. J. (2017). Unsupervised monocular depth estimation with left-right consistency. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 270–279.

Jörgensen, E., Zach, C., and Kahl, F. (2019). Monocular 3d object detection and box fitting trained end-to-end using intersection-over-union loss. *arXiv preprint arXiv:1906.08070*.

Kalman, R. E. et al. (1960). A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 82(1):35–45.

Kim, Y. and Kum, D. (2019). Deep learning based vehicle position and orientation estimation via inverse perspective mapping image. In *IEEE Intelligent Vehicles Symposium (IV)*, pages 317–323.

Kuhn, H. W. (1955). The hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2(1-2):83–97.

Law, H. and Deng, J. (2018). Cornernet: Detecting objects as paired keypoints. In *European Conference on Computer Vision (ECCV)*, pages 734–750.

Liu, Y., Yixuan, Y., and Liu, M. (2021). Ground-aware monocular 3d object detection for autonomous driving. *IEEE Robotics and Automation Letters*, 6(2):919–926.

Liu, Z., Wu, Z., and Tóth, R. (2020). Smoke: Single-stage monocular 3d object detection via keypoint estimation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 4289–4298.

Luiten, J., Os Ep, A. A., Dendorfer, P., Torr, P., Geiger, A., Leal-Taixé, L., and Leibe, B. (2021). Hota: A higher order metric for evaluating multi-object tracking. *International Journal of Computer Vision*, 129(2):548–578.

Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 779–788.

Redmon, J. and Farhadi, A. (2017). Yolo9000: better, faster, stronger. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7263–7271.

Redmon, J. and Farhadi, A. (2018). Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*.

Ren, S., He, K., Girshick, R., and Sun, J. (2017). Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6):1137–1149.

Roddick, T., Kendall, A., and Cipolla, R. (2018). Orthographic feature transform for monocular 3d object detection. *arXiv preprint arXiv:1811.08188*.

Srivastava, S., Jurie, F., and Sharma, G. (2019). Learning 2d to 3d lifting for object detection in 3d for autonomous vehicles. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4504–4511.

Szeliski, R. (2022). *Computer Vision*. Springer International Publishing, Cham.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, 30.

Voigtlaender, P., Krause, M., Osep, A., Luiten, J., Sekar, B. B. G., Geiger, A., and Leibe, B. (2019). Mots: Multi-object tracking and segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7942–7951.

Wang, Z., Zheng, L., Liu, Y., Li, Y., and Wang, S. (2020). Towards real-time multi-object tracking. In *European Conference on Computer Vision (ECCV)*, pages 107–122.

Weng, X. and Kitani, K. (2019). Monocular 3d object detection with pseudo-lidar point cloud. In *IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, pages 857–866.

Weng, X., Wang, J., Held, D., and Kitani, K. (2020). Ab3dmot: A baseline for 3d multi-object tracking and new evaluation metrics. *arXiv preprint arXiv:2008.08063*.

Yu, F., Wang, D., Shelhamer, E., and Darrell, T. (2018). Deep layer aggregation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2403–2412.

Zhang, Y., Lu, J., and Zhou, J. (2021a). Objects are different: Flexible monocular 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3289–3298.

Zhang, Y., Ma, X., Yi, S., Hou, J., Wang, Z., Ouyang, W., and Xu, D. (2021b). Learning geometry-guided depth via projective modeling for monocular 3d object detection. *arXiv preprint arXiv:2107.13931*.

Zhang, Y., Wang, C., Wang, X., Zeng, W., and Liu, W. (2021c). Fairmot: On the fairness of detection and re-identification in multiple object tracking. *International Journal of Computer Vision*, 129(11):3069–3087.

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

Zhu, X., Hu, H., Lin, S., and Dai, J. (2019). Deformable convnets v2: More deformable, better results. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9308–9316.

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