VETRA: A Dataset for Vehicle Tracking in Aerial Imagery – New Challenges for Multi-Object Tracking

Jens Hellekes[1], Manuel Mühlhaus[1], Reza Bahmanyar[1], Seyed Majid Azimi[1], and Franz Kurz[1]

Remote Sensing Technology Institute, German Aerospace Center (DLR),
Wessling, Germany
jens.hellekes@dlr.de
https://www.dlr.de/eoc/en

Abstract. The informative power of traffic analysis can be enhanced by considering changes in both time and space. Vehicle tracking algorithms applied to drone videos provide a better overview than street-level surveillance cameras. However, existing aerial MOT datasets only address stationary settings, leaving the performance in moving-camera scenarios covering a considerably larger area unknown. To fill this gap, we present VETRA, a dataset for vehicle tracking in aerial imagery introducing heterogeneity in terms of camera movement, frame rate, as well as type, size and number of objects. When dealing with these challenges, state-of-the-art online MOT algorithms experience a decrease in performance compared to other benchmark datasets. The integration of camera motion compensation and an adaptive search radius enables our baseline algorithm to effectively handle the moving field of view and other challenges inherent to VETRA, although potential for further improvement remains. Making the dataset available to the community adds a missing building block for both testing and developing vehicle tracking algorithms for versatile real-world applications.
VETRA can be downloaded here: https://www.dlr.de/en/eoc/vetra.

Keywords: Multi-object tracking · Vehicle tracking dataset · Aerial image sequences

1 Introduction

Running Multi-Object Tracking (MOT) algorithms specifically designed for vehicles can provide valuable insights for transport research, such as analyzing the current traffic state, calculating travel times along corridors, incident analysis and generating naturalistic driver data. Aerial perspectives offer superior overviews compared to CCTV cameras, enabling behavioural analysis of the same traffic participants over time and under various conditions. However, the use of common Unmanned Aerial Vehicle (UAV) for data acquisition presents technical and regulatory challenges, such as control range, velocity, flight altitude, and restrictions over built-up areas. These limitations often confine data

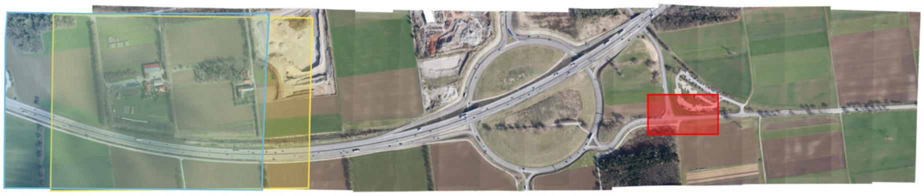
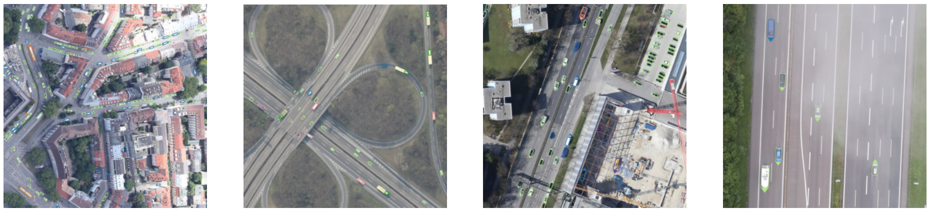


Fig. 1: Schematic comparison of the area covered by a VETRA sequence (underlying mosaic) and a nadir sequence from the VisDrone dataset [57] (red overlay). The camera movement between two consecutive frames is visualized by blue and yellow overlay.

collection to a single intersection or a city block. While increasing coverage by flying multiple UAVs simultaneously is an option, it introduces its own set of complications [4]. This requires other platforms such as airplanes, helicopters, and special UAVs that fly at higher altitudes and cover longer ranges. To maintain information richness, they are typically equipped with photogrammetric camera systems that acquire images at 1 to 2 Hz, providing the necessary Field Of View (FOV), spatial resolution, and forward motion compensation for rapid surveillance. Altogether, this makes MOT more challenging, but successful application of algorithms opens up new and relevant use cases.

In aerial and UAV imagery, MOT presents unique challenges compared to in-situ scenarios. Aerial imagery typically contains smaller objects from multiple categories, often with a highly imbalanced category count, which increases the complexity of detection and degrades the reliability of appearance-based methods. These images often cover large areas with numerous objects with varying motion patterns. In such scenarios, both the target objects and the camera exhibit fast and irregular motion. This requires a more accurate motion modeling approach than traditional methods such as the Kalman Filter (KF). Therefore, several recent works have introduced specialized MOT algorithms tailored for aerial and UAV imagery [19, 27, 46, 47]. These methods have been evaluated on the existing datasets like UAVDT and VisDrone and show remarkable performance. So far, scenarios with moving cameras and low frequencies are reflected only to a very limited extent by the available datasets so that the performance of MOT algorithms for such applications remain unclear. Our contributions:

- We release the VETRA dataset for MOT in aerial images, unique in terms of spatial coverage per sequence, the extent of camera movement and high-quality polygon, Oriented Bounding Box (OBB) and Horizontal Bounding Box (HBB) annotations. Furthermore, we propose a vehicle classification scheme which is application-driven and can make the link between computer vision and transport research more seamless.
- We benchmark various online MOT algorithms to highlight challenges introduced by VETRA and identify areas for future MOT development.
- We propose Deep SR-SORT, an enhancement of Deep-SORT with the BRISK algorithm for motion compensation and an adaptive search radius. It outperforms all algorithms tested.



(a) High number of objects in an urban scenario with very low resolution.

(b) Occluded and invisible objects on an interchange at low resolution.

(c) Non-classifiable vehicles in a suburban area at medium resolution.

(d) Adverse weather conditions on a motorway at high resolution.

Fig. 2: Zoomed-in polygon annotations from VETRA dataset illustrate the variety of spatial structures depicted under various viewing settings. The class colors are shown in Tab. 1. Not all attributes are displayed.

2 Dataset

The VETRA dataset consists of 52 image sequences acquired by both airplane and helicopter with DLR's 3k and 4k camera systems [20, 21] between 2012 and 2022 at different times of day and year. The imagery reflects heterogeneous spatial structures and infrastructure types throughout Germany under varying illumination and viewing conditions (see Fig. 2). Due to different flight parameters and camera system configurations, objects are displayed at various Ground Sampling Distance (GSD), frame rate and viewing angle. Furthermore, the platforms performed movements along all three axis leading to overlaps between subsequent frames of 55 % to almost 100 %. This range makes VETRA unique among the aerial multiple-vehicle tracking datasets and leads to a substantially increased spatial coverage ranging from 0.06 to 9.22 km^2 (see also supplementary material, Sec. 6.2 and 6.5). Fig. 1 shows the area covered by an exemplary VETRA sequence and the size comparison with a VisDrone sequence taken from the nadir perspective [57]. In addition to the camera motion, vehicles move from low (traffic jams, city traffic) to high speeds (free flow on motorways). Detailed sequence and scene statistics are shown in Fig. 3 and Sec. 6.3. Currently, most MOT methods operate without geospatial data input. However, integrating this information can enhance robustness, particularly in scenarios involving object and camera movement [16]. Hence, we provide metadata for each sequence.

2.1 Vehicle classification scheme

The scheme for classifying vehicles in the VETRA dataset can built upon extensive previous work. Several aerial detection datasets feature road-based vehicles as one of the object classes among others, *e.g.* AI-TOD-v2 [45], DIOR [24], COWC [33], UCAS-AOD [56], and NWPU VHR-10 [11]. Some datasets focus on vehicles or cars only, like CARPK dataset [15]. For multi-class detection, most datasets distinguish in two classes (small and large vehicles), *e.g.* SODA-A [12], EAGLE [2], DOTA [44], and DLR-MVDA [26]. DroneVehicle [39], VAID [25],

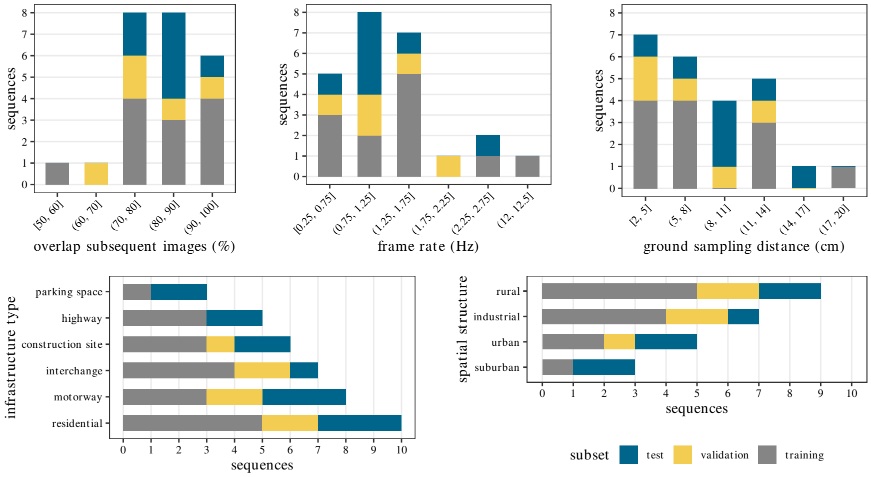


Fig. 3: Image sequence and scene characteristics of VETRA train, val, and test sets. As the LAM test set is application-oriented with less variety, it is not shown here.

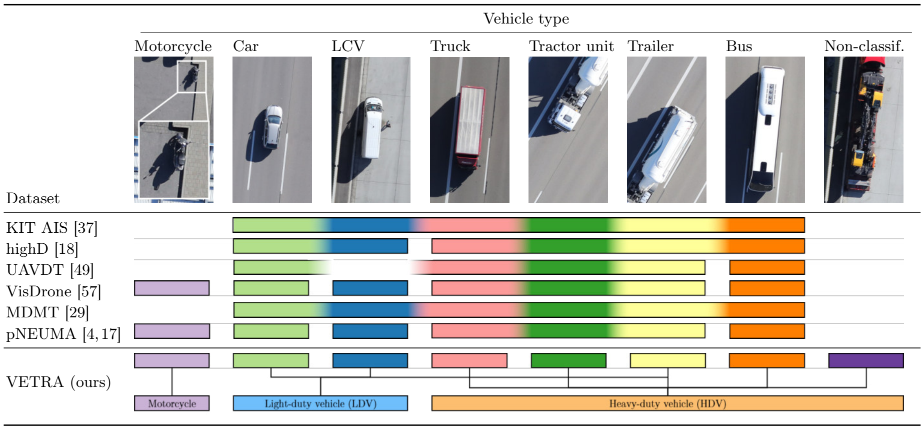
xView [22], and VEDAI dataset [35] feature more than two road-based vehicle classes, but each with different category definitions. Datasets like xView are focused on images with a GSD of 30 cm so that motorcycles are not annotated.

In the field of MOT on canonical images, GMOT-40 [3], Waymo [38] and KITTI MOTS [40] feature one vehicle class. Both the BDD100K [48] and UA-DETRAC dataset [42] distinguish 4 classes, while the latter consider trucks in the class ‘others’. TAO [13] features 15 vehicle classes but the definitions are mainly appearance-based, *e.g.* ‘car’, ‘cab_(taxi)’, ‘convertible_(automobile)’.

To the best of our knowledge, six datasets exist currently for vehicle tracking in aerial imagery. The generically formulated vehicle types in Tab. 1 are mostly covered by these datasets but in different levels of granularity. Both MDMT [29] and KIT AIS [37] consider vehicles as one class for annotation, while highD [18] differentiates cars and trucks. UAVDT [49] adds busses to this list, but from the information available it remains unclear how Light Commercial Vehicle (LCV) are categorized. pNEUMA Vision [17] and VisDrone [57] are the first datasets introducing motorcycles as additional class.

Altogether, the incompleteness of vehicle types covered paired with a partially soft delimitation, restricts the application potential in areas like transport research. With VETRA, we want to make the MOT results comparable and compatible with the requirements of other domains, *e.g.* by allowing them to be merged with count station data of high temporal resolution. But instead of annotating the same classes as defined by a single count station operator, we implement a framework that flexibly adapts to schemes of multiple authorities and considers the computer vision perspective on the problem. The classification scheme used by VETRA and the conversion to other schemes is shown exemplarily for Germany [7] in Tab. 1 and in the supplementary material, Sec. 6.4.

Table 1: Comparison of the classification schemes used by MOT annotation datasets. A combination of multiple vehicle types within one dataset class is represented by color transition. Fading color bars represent a fuzzy class delimitation. Sample images are displayed at the same spatial resolution and zoom level to allow for comparisons. The VETRA dataset offers both fine-grained and aggregated classification schemes; the latter is used for multi-class detections in this paper.


Vehicle type
Motorcycle
Car
LCV
Truck
Tractor unit
Trailer
Bus
Non-classif.
Dataset
KIT AIS [37]
highD [18]
UAVDT [49]
VisDrone [57]
MDMT [29]
pNEUMA [4, 17]
VETRA (ours)
Motorcycle
Light-duty vehicle (LDV)
Heavy-duty vehicle (HDV)

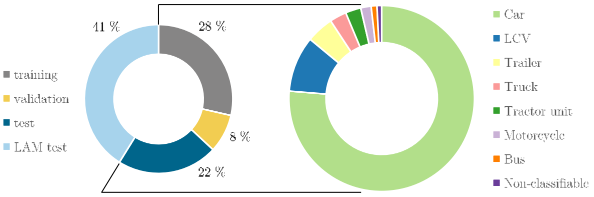
2.2 Vehicle annotation

VETRA is the first MOT dataset using polygons in addition to HBB and OBB despite the high cost of annotation. HBB as state of practice for MOT algorithms show deficits in aerial imagery, as they contain a significant part of the background and therefore distract the networks from important objects [34]. This problem is amplified when the GSD becomes higher and objects are displayed by fewer pixels. OBB can only partially address this: in high-resolution images they still contain a significant amount of background, esp. in oblique views, for articulated busses, and vehicles with trailers. Polygon annotation with a limited number of points (for VETRA, up to 8 points for articulated busses which are truncated two times by an image corner) is the trade-off between OBB and fine-grained segmentation masks we chose. To ease first experiments, we provide both OBB and HBB annotations in addition to polygon annotations. The vehicle classes for both types of Bounding Box (BBox) are aggregated according to Tab. 1.

The class-wise annotations are enriched by attributes indicating if a vehicle is temporarily *stopped* (*e.g.*, due to traffic light), or *carried* by another vehicle. For trailers, the attribute *attached* describes whether they are coupled to another vehicle. At the same time, tractor units receive in such cases the attribute *connected*. The attributes *occluded* and *invisible* reflect if a vehicle is partially/fully covered by other objects, *truncated* shows that it's partially outside of the image borders, and *difficult* when class or attribute assignment is ambiguous.

		Instances per GSD (cm)		
Class	Objects	[2, 5]	(5, 14]	(14, 20]
LDV	16539	6874	78359	2509
HDV	3367	523	15884	813
Motor.	110	664	519	0

(a) Annotated objects and instances (aggregated scheme).



(b) Instance share per subset and class (detailed scheme).

Fig. 4: Statistics for annotated objects and instances. The vehicle classification schemes are explained in Tab. 1.

The sequences are carefully assigned to training, validation and test set not only to ensure a balanced distribution of instances (49 %, 14 %, and 37 % respectively, see Fig. 4) but also to account for the sequence and scene characteristics (see Fig. 3). We also consider an application-centric test set where we focus on the fast moving camera and large area monitoring, called LAM. It includes seven scenarios in each of which the airplane flies over a highway or motorway four times (28 sequences in total).

2.3 Comparison with existing datasets

In Tab. 2, we compare VETRA with other MOT datasets that feature road-based vehicle classes. Due to the domain gap between canonical and aerial views, the comparison focuses solely on overhead imagery. VETRA covers all application cases outlined in Sec. 1, including hovering at low altitudes, expanded FOV from higher altitudes, and scenarios requiring a moving camera. UAVs, the most commonly used platform, allow for long sequences captured at high frame rates with comparatively little effort. However, their limited observable area results in a small number of vehicles captured at low altitudes (11 to 45 instances for UAVDT and KIT AIS, respectively). The pNEUMA dataset stands out as the only one with extensive spatial coverage due to its relatively high altitude. Although primarily designed for traffic analysis, its annotation type and classification are suitable for computer vision tasks to some extent. The majority of VETRA sequences exhibit a significantly reduced overlap of subsequent frames (55 to 89 %). Due to the image sensor sizes (up to 80 times larger than VisDrone images) and high flight altitudes, the instance density can reach up to 623 vehicles (average: 109 instances/frame), highlighting the added value of more criteria for dataset comparison than solely the total number of frames. VETRA sequences support long-term tracking by (i) showing vehicles over a larger area, (ii) sequences with a long duration of 21 s on average, and (iii) sequences with a camera movement of similar speed as the vehicles. Lastly, the majority of the sequences in the existing datasets were captured in China. VETRA adds diversity by presenting sequences of urban, rural, and highway scenes throughout Germany (see supplementary material, Sec. 6.1), which can be adapted to other countries, especially in Central Europe.

Table 2: Comparison of existing datasets with VETRA dataset per application case: 1) hovering, small FOV, 2) hovering, large FOV, and 3) moving camera. Aerial imagery is captured by airplane (A), UAV or helicopter (H) from nadir (N) or oblique (O) view. All information are extracted from the publicly available annotations and/or the corresponding publications. '–' indicates that data are unavailable. The number of sequences and instances accounts for classes of motorized road traffic.

			Application			Frames		Sequences			Annotations				
Dataset	Plat.	View	1)	2)	3)	Nr.	Avg. size (px)	FPS	Nr.	Length (s)	Type	Class.	Obj.	Inst.	Obj./S.
KIT AIS [37]	H	N	✓			239	1085 × 618	2	9	13	OBB	1	464	10817	52
highD [18]	UAV	N	✓			1485000	4096 × 2160	25	60	1020	HBB	2	110000	–	1833
UAVDT [49]	UAV	N, O	✓			80000	1080 × 540	30	50	27	HBB	3	2700	841500	54
VisDrone [57]	UAV	N, O	✓			32131	1902 × 1069	–	78	–	HBB	4	5703	900426	73
MDMT [29]	UAV	N, O	✓			38977	1920 × 1080	–	87	–	HBB	1	8117	1495458	93
pNEUMA [17]	UAV	N		✓		35000	3840 × 2100	2.5	18	780	OBB	6	10622	–	590
VETRA (ours)	A, H	N, O	✓	✓	✓	3870	5592 × 3728	1.3	52	72	OBB	8	20370	109195	392

3 Algorithms

The common approach in most MOT methods is tracking-by-detection. First, a Deep Neural Network (DNN) is used to detect objects within each frame of the image sequence. In the association phase, the detected objects are then connected across frames based on their motion information or visual appearance, or a combination of both.

The KF is a widely used motion modeling technique in MOT methods. SORT [5] predicts tracklet positions in the new frame using KF and computes the similarity as the Intersection over Union (IoU) between the detected and predicted boxes. ByteTrack [52] represents a significant advancement over SORT with an improved detector and accounting for low confidence detections in the matching process. OC-SORT [8] fixes the error accumulation of KF parameters to improve performance in the presence of occlusion and nonlinear motion. BoT-SORT [1] modifies the KF's state vector to estimate the BBox dimensions directly and compute affine transformations for Camera Motion Compensation (CMC). To handle occlusions and minimize identity switches over long tracking periods, Deep SORT [43] merges object motion and appearance information. It uses a combination of Mahalanobis and cosine distances, with motion derived from a KF and appearance from a DNN trained on a re-identification (Re-ID) dataset. Deep OC-SORT [31] enhances OC-SORT through the adaptive integration of appearance information, leading to superior performance. SMILEtrack [41] employs a transformer-based Siamese network for appearance feature matching and Re-ID, alongside a modified KF for motion estimation. It effectively addresses MOT challenges including occlusions, objects of similar appearance, and complex scenes. SUSHI [10] unifies short- and long-term associations for occluded and non-occluded objects, making the model general across temporal scales. It uses spatial and motion-based proximity, temporal distance, and appearance features to create the associations. MOTR [50] is an end-to-end MOT method that builds on DETR [9], a transformer-based detection method that incorporates joint motion and appearance modeling. MOTR frames MOT as a set of sequence prediction tasks, where each sequence represents an object trajectory. MOTRv2 [53] enhances MOTR by introducing an additional object detection step before MOTR is applied, resulting in significant performance improvements.

Several recent works have introduced MOT algorithms specifically designed for challenges in aerial and UAV imagery [19,27,46,47]. AerialMPTNet [19] uses a Siamese network to match appearance features and detect the new position of objects in each frame. It integrates motion information with a Long Short-Term Memory (LSTM) and considers object interactions with a Graph Convolutional Neural Network (GCN). UAVMOT [27] introduces an ID feature update module to improve object association and uses an adaptive motion filter to handle complex motion. It also employs focal loss for category imbalance and small object detection. STN-Track [46] uses a detection method based on the Swin Transformer [28] and improves tracking accuracy by using NSA-KF [14] and generalized IoU [36]. FOLT [47] balances speed and accuracy with an advanced detector and lightweight optical flow extraction. It enhances detection of small objects and improves tracking of objects with large displacements.

4 Experiments

In this section, we benchmark several MOT algorithms including ByteTrack, BoT-SORT, Deep SORT, and Deep OC-SORT. To ensure comparability, tracking experiments are performed based on HBB. For our experiments, we choose methods with publicly available source code that can address the challenges of aerial MOT. SUSHI was excluded because to its offline tracking and MOTRv2 because of its high computational requirements (particularly relevant given the large images in the VETRA dataset). Note that the detections are generated in OBB format and converted to HBB afterwards.

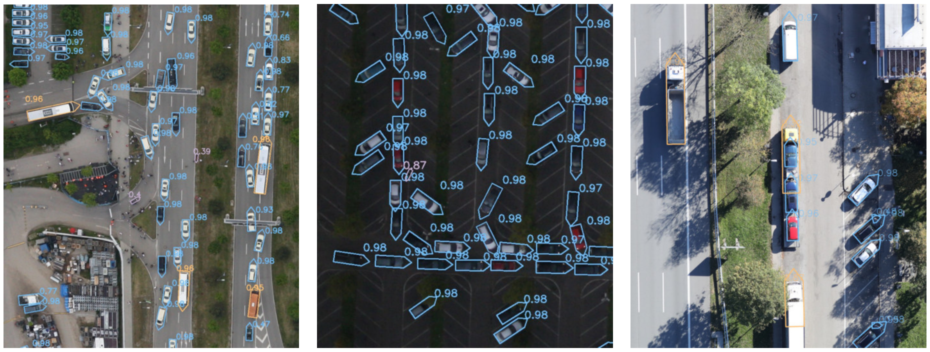
4.1 Detection

For detection, we use the DINO algorithm [51] with an adaptation for OBB detection following [32] and pre-training on the EAGLE dataset [2]. The large aerial images are divided into patches of size 1024 × 1024 pixels with 20 % overlap. Multi-scale training is performed by resampling the image patches at rates of {0.5, 1, 1.5}. In order to avoid multiple detections for each object, intraclass and interclass Non-maximum Suppression (NMS) with IoU thresholds of 0.1 and 0.3 respectively are used.

Tab. 3 shows the detection results on the validation and test sets, including AP for each class and mAP. Orientation prediction is evaluated using the OTP, which is the correctly predicted orientation for True Positive (TP) divided by the total number of TP, with the IoU threshold set to 0.5 and the orientation threshold set to 10 °. Both Tab. 3 and Fig. 5 show a satisfactory overall detection performance for the HDV and LDV classes, but a poor performance for the motorcycle class. This can be attributed to the limited representation

Table 3: Detection results on the validation and test sets. All values are in percent. The symbol ‘ * ’ indicates joint training on the training and validation sets.

Set	AP_{HDV}	AP_{LDV}	AP_{M}	mAP	OTP_{HDV}	OTP_{LDV}	OTP_{M}
Val	78.01	97.19	29.81	68.34	83.44	98.19	55.33
Test	83.49	96.89	31.31	70.56	85.90	98.09	62.34
Test*	86.67	96.71	34.47	72.62	87.57	97.93	60.46



(a) Groups of people are misinterpreted as motorcycles in a medium-resolution scenario.

(b) Accuracy of BBox orientation is high even in scenarios with low illumination.

(c) High-resolution scenes challenge the assignment of the correct BBox size and orientation.

Fig. 5: Zoomed-in detections based on aggregated class scheme as displayed in Tab. 1.

of motorcycles in the dataset (1.1 % of all instances). In addition, the detection algorithm performs better on the test set than on the validation set for the HDV and motorcycle classes, highlighting greater challenges in the validation set for these classes. Additionally, using both the training and validation sets improve the results on the test set, especially for the HDV and motorcycle classes. Upon closer inspection, we found that the results for high-resolution sequences are generally below average, likely due to the low number of overall high-resolution vehicle instances present (see Fig. 5c).

4.2 Re-identification

For Re-ID, we utilize ResNet50 from the Torchreid library, designed primarily for person Re-ID [55]. The model is trained on our dataset’s training set and evaluated on the test set. During evaluation, a random sample from each object’s tracklet is placed in the query set, and the remainder in the gallery. Re-ID is carried out with (i) HBB, and (ii) HBB, whereby the background of the object is masked by the corresponding OBB. The comparison shows that incorporating background information using HBB yields superior performance with an mAP of 73.2 % and a Rank-1 of 92.9 % than the approach without background information, which achieves an mAP of 68.3 % and a Rank-1 of 91.5 %. This highlights the importance of including background information for effective Re-ID.

Table 4: Tracking on the VETRA test set. All metrics except IDSW are in percent.

Method	HOTA↑	MOTA↑	MOTP↑	IDSW↓	IDR↑	IDP↑	IDF1↑
Byte-Track	36.37	13.56	87.75	17,328	17.68	17.95	17.81
BOT-SORT	50.84	18.50	87.04	6,886	56.32	42.08	48.17
Deep SORT	58.29	76.42	88.35	3,077	58.52	57.58	58.05
Deep SORT$_{\text{OBB}}$	54.92	71.94	88.35	4,091	54.02	53.16	53.59
Deep OC-SORT	46.84	44.69	88.28	10,334	31.43	30.92	31.17
Deep OC-SORT$_{\text{DIoU}}$	39.47	32.59	88.28	13,068	32.40	31.89	32.14
Deep SR-SORT (our)	**82.18**	**88.45**	**88.51**	**792**	**89.27**	**91.39**	**90.32**

4.3 Tracking

For our tracking experiments, we use the MOT methods available in the Box-MOT framework [6]. We customize the detection and Re-ID modules as described in Sec. 4.1 and 4.2. The default parameters are kept except for 'maximum age', which defines the number of frames an object remains in memory after disappearing and is set to 2 to account for the low frame rates, and 'min hits', which is set to 0 to allow tracks with vehicles appearing only once. Tab. 4 shows the tracking results on VETRA test set based on the HOTA, MOTA, MOTP, IDSW, IDR, IDP, and IDF1 metrics from the TrackEval codebase [30].

ByteTrack, which relies solely on motion information, shows low tracking performance across all metrics. The model encounters difficulties in initializing the motion model because it is unable to find the initial matching BBox based on IoU. The qualitative evaluation shows that ByteTrack is only effective in scenarios with stationary vehicles and minimal camera movements, such as in hovering scenarios. BoT-SORT achieves a significantly smaller ID Switch (IDSW) through the combined use of CMC and Re-ID modules. However, the difficulties in initializing the motion model and the use of IoU, which is not designed for sequences with low frame rates and large object motion, limit the improvement in tracking accuracy compared to ByteTrack. Deep SORT achieves significantly better tracking results with lower IDSW by relying solely on the Re-ID module. We removed the influence of the motion model by setting the KF influence to 0, following the recommendation in [43]. While this reduces the problems caused by inappropriate motion models, it introduces a new challenge: even if a similar object exists at a considerable distance from the expected position of the target object, there is a high probability that the two objects will be incorrectly matched. To assess the impact of background information in Re-ID on tracking results, we use HBB with masked backgrounds for the Re-ID task in Deep SORT$_{\text{OBB}}$, as described in Sec. 4.2. The results show a slight decrease in tracking accuracy with an increased number of IDSW.

Deep OC-SORT performs worse than Deep SORT despite the integration of CMC, motion models, and Re-ID. Similar to BoT-SORT, current motion modeling is not well adapted to the scenarios reflected by the VETRA dataset, resulting in more mismatches. However, due to its Re-ID module, the tracking accuracy is improved. Qualitative evaluation shows that CMC performs well, resulting in high tracking accuracy for stationary vehicles. Moreover, prolonged visibility

Table 5: Deep SR-SORT's tracking results on the validation, test, and LAM test sets of VETRA. The LAM test results represent the average over four revisits. For 'LAM_Stuttgart', the results of each individual revisit are shown in detail. All metrics except IDSW and IDSWR are in percent.

Set	ID	Sequence	GSD	FPS	Overl. (%)	Inst./Fr.	HOTA↑	MOTA↑	MOTP↑	IDSWR↓	IDSW↓	IDF1↑
Val	13	Landsberg_roundabout	13	0.56	80	70	84.47	92.98	86.94	0.083	17	95.44
	14	Munich_VHR_motorway_service	2	1.45	64	14	76.98	63.98	91.42	0.090	6	79.53
	15	Hamburg_river_bridge	11	1.00	78	109	72.75	71.27	87.81	0.430	136	76.53
	16	Munich_test_vehicle	8	0.98	91	28	74.92	82.74	90.09	0.671	55	77.78
	17	Munich_multimodal_crossing	4	1.87	83	86	75.30	76.13	92.26	1.010	175	76.36
		Average of validation set	7	1.34	83	61	76.18	78.24	90.68	0.460	389	78.94
Test	18	Munich_stadium	10	0.42	70	189	80.52	83.94	90.00	0.215	91	86.77
	19	Munich_railroad	14	1.11	88	623	78.74	86.22	84.53	0.130	131	91.00
	20	Moencheng._parking	8	1.67	89	498	86.78	94.59	89.20	0.100	111	95.92
	21	Munich_Y_interchange	10	0.76	90	51	75.07	85.96	86.53	0.643	81	80.09
	22	Kufstein_river	16	1.00	86	120	83.82	87.87	88.15	0.056	16	93.28
	23	Holzk._VHR_motorway	3	2.39	80	6	75.58	61.09	92.03	0.109	7	79.28
	24	Munich_tunnel	9	0.98	88	322	81.82	88.05	89.56	0.346	355	88.55
		Average of test set	8	1.49	84	163	82.18	88.45	88.51	0.200	792	90.32
LAM test	25-28	LAM_Augsburg	9	1.00	79	15	78.40	75.71	89.24	0.107	438	84.93
	29-32	LAM_Nesselwang	9	1.00	79	3	78.51	76.51	89.56	0.136	33	84.33
	33-36	LAM_Kempten	9	1.00	79	4	81.45	76.55	90.51	0.039	12	87.09
	37-40	LAM_Grunbach	10	1.00	79	11	82.18	80.86	90.89	0.058	40	88.21
	41-44	LAM_Stuttgart	9	1.00	79	30	87.48	90.27	91.23	0.070	151	94.13
	45-48	LAM_Ammelshain	10	1.00	79	13	83.33	86.24	89.12	0.102	112	91.38
	49-52	LAM_Rothschoenberg	10	1.00	79	16	83.24	84.73	89.32	0.081	118	91.10
		Average of LAM test set	9	1.00	79	14	82.17	82.01	89.86	0.094	904	89.07
	41	LAM_Stuttgart_revisit_1	9	1.00	79	23	86.78	88.77	91.27	0.063	27	93.23
	42	LAM_Stuttgart_revisit_2	9	1.00	79	19	85.20	84.86	91.33	0.148	47	90.81
	43	LAM_Stuttgart_revisit_3	9	1.00	79	37	87.89	90.17	91.31	0.037	25	94.43
	44	LAM_Stuttgart_revisit_4	9	1.00	79	43	88.57	93.65	91.10	0.072	52	95.87

of an object in the sequence is associated with improved tracking performance, highlighting the critical challenge of obtaining the initial match. Therefore, incorporating orientation information using OBB can improve the motion model through faster and more accurate initialization, even for shorter tracks. To restrict the search area and prevent matches over large distances, we replaced IoU with Distance IoU (DIoU) [54] in Deep OC-SORT$_{\mathrm{DIoU}}$ with a threshold of 0.05. While this effectively prevents mismatches across the entire image, it degrades performance for nearby stationary vehicles, especially in parking lots.

Leveraging insights gained from the challenges faced by the tested methods, we propose Deep SR-SORT, an improvement of Deep SORT by adding BRISK [23] for CMC and an adaptive search area, where infinite cost is assigned to the objects outside this radius during matching. The search area is an oriented rectangle with dimensions determined by the maximum possible vehicle displacement. This displacement is obtained by estimating the GSD from the average detection size of the LDV class, the frame rate of the sequence, and assuming a maximum vehicle speed of 60 m/s. The rectangle is then oriented and positioned in front of the vehicles using OBB detections.

Tab. 5 shows the tracking results obtained by Deep SR-SORT for the validation, test, and LAM test sequences, together with their specifications. For 'LAM_Stuttgart', both the average and individual revisit results are provided to show the temporal dynamics of traffic on this highway segment.

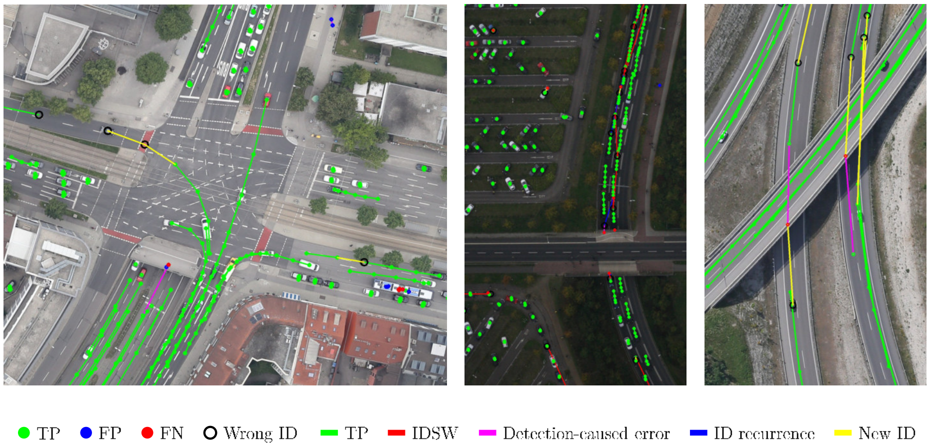


Fig. 6: Exemplary tracking results of the proposed baseline method Deep SR-SORT. The confusion plots show the sequences 17, 20 and 21 (left to right).

In both validation and test sets, the results are worse for the sequences with low GSD (seq. 14 and 23). Despite better object visibility, the narrower FOV and fewer instances (see Tab. 4) limit effective training. In addition, factors such as camera motion, frame rate, and vehicle speed affect tracking more significantly, especially for methods that rely on object motion information (*e.g.*, a vehicle in seq. 23 appears for about 7 frames in the direction of flight and only for 3 frames in the opposite direction). In addition, for sequences with high GSD (*e.g.*, seq. 21), relying solely on appearance features for vehicle association leads to poor tracking accuracy and a large number of mismatches. Furthermore, a complex scenario and varying vehicle motion behavior can cause tracking errors and mismatches, such as in a multi-level intersection (seq. 24).

Fig. 6 shows the confusion plot for selected areas from seq. 17, 20, and 21 to provide a visual representation of the tracking performance. In the first example, at the entrance of the tunnel, the mostly occluded car is not detected correctly. The car turning left is incorrectly assigned a new track ID and the articulated bus on the right is recognized as two objects. Despite the low illumination and poor visual features in the center image, the vehicles are tracked well. In the last example, Deep SR-SORT fails to track vehicles passing under the bridge, highlighting a limitation in capturing the historical information of the tracks.

Applying Deep SR-SORT to the LAM test sequences with similar acquisition characteristics in terms of GSD, frame rate, and overlap demonstrates that both the number of vehicles and the complexity of the traffic influence the tracking performance. As displayed in Fig. 7, the traffic dynamics on the same highway segment can differ greatly between revisits. Especially in a congested situation, Deep SR-SORT shows deficits in matching the correct HDV, as they have similar appearance and are located in the same search area.

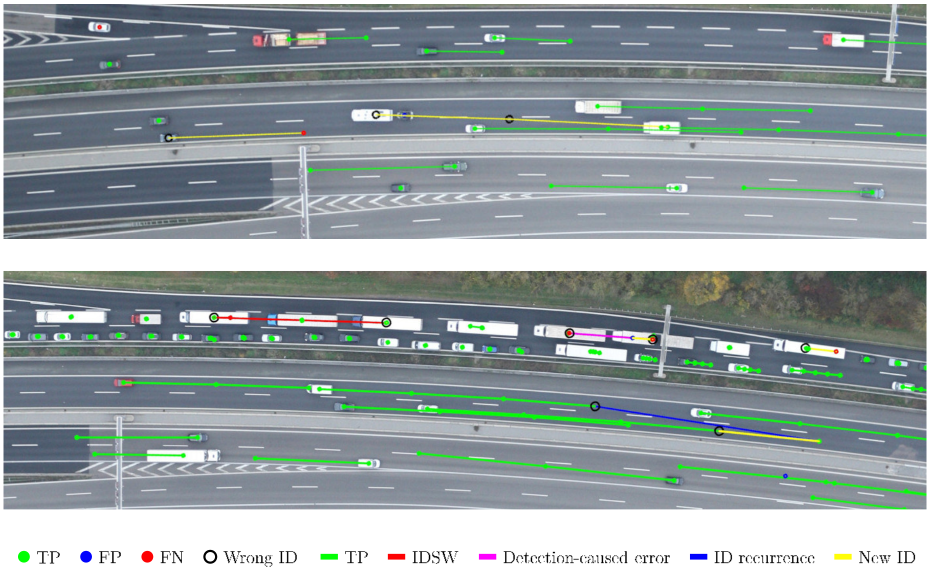


Fig. 7: Confusion plots for the tracking results generated by Deep SR-SORT. The road sections are part of 'LAM_Stuttgart' and were captured at different points in time (sequences 42 and 44).

4.4 Cross-dataset evaluation

To evaluate the generalization capabilities of the trained models across datasets, we performed a cross-dataset evaluation between VETRA and VisDrone. Deep SR-SORT was trained on each dataset and then tested on the other. To remove the influence of detection performance, we use the ground truth labels. As MOTA and HOTA are metrics for detection and tracking performance, these figures are inflated and need to be interpreted accordingly. Tab. 6 displays the tracking and Re-ID results, since it is the only trained module in Deep SR-SORT. The results consistently show the superior performance of Deep SR-SORT on VisDrone, even when trained on the VETRA dataset. This can be attributed to the relatively simple scenarios in VisDrone, such as an almost stationary camera position and high frame rates. The good performance on VisDrone achieved by training on VETRA underlines the robustness and generalizability of the trained models. However, the lower performance on VETRA (*e.g.*, more IDSW), regardless of whether training is performed on VETRA or VisDrone, indicates the presence of specific challenges that Deep SR-SORT can hardly overcome. To effectively master these challenges, MOT algorithms need to be further developed.

Table 6: Cross-dataset evaluation of Deep SR-SORT's tracking results based on ground truth annotations. All metrics except IDSW are in percent.

Train	Test	HOTA ↑	MOTA ↑	IDSW ↓	IDF1 ↑	Re-ID mAP ↑	Re-ID Rank-1 ↑
VETRA	VETRA	94.74	95.17	791	93.76	**73.2**	92.9
VETRA	VisDrone	96.26	99.76	374	95.19	59.7	99.8
VisDrone	VisDrone	**97.90**	**99.88**	**183**	**97.02**	71.6	**99.9**
VisDrone	VETRA	94.24	94.54	932	93.03	69.4	91.9

5 Conclusion

In this paper, we introduce the VETRA dataset, a novel resource for multi-class vehicle tracking in aerial imagery. By releasing it to the public, we aim to encourage the development of innovative MOT algorithms capable of addressing real-world challenges. In particular, the careful data selection and annotation of our dataset enables its applicability not only to tracking, but also to instance segmentation, detection, and classification tasks. As valuable addition to the current MOT datasets, VETRA presents unique challenges such as low frame rates, small fast-moving objects, and high camera motion. These characteristics allow for extended tracking of numerous vehicles with varying motion behaviors over large areas.

Previous multi-class vehicle detection datasets have used appearance-based classification schemes. However, differentiating between small and large vehicles, a common approach, faces challenges as highlighted in [33]. Other datasets, such as VAID, differentiate up to seven classes, but exhibit imbalances in the level of detail, making comparability difficult. VETRA addresses this by providing vehicle annotations on an atomic level that target the computer vision perspective. At the same time, VETRA proposes a fusion scheme that allows for the compilation of classes to make the results compatible with transport research applications. In future work, we plan to expand the dataset to include sequences with higher GSD, particularly relevant for high altitude platforms. In addition, we will include sequences with larger fields of view and a greater number of objects.

Benchmarking state-of-the-art MOT methods on VETRA highlights the challenges they face due to the unique characteristics of the dataset. Most MOT methods are optimized for scenarios with low camera motion, high frame rates, and slow moving objects. The motion modeling in these methods needs to be adapted to effectively handle the complex motion in VETRA. Leveraging the insights gained from the challenges faced by the tested methods, we propose Deep SR-SORT, an improvement of Deep SORT by adding BRISK for CMC and an adaptive search area. It significantly outperforms the tested methods. We also perform a cross-dataset validation that demonstrates the robustness and generalizability of the models trained on the VETRA dataset. Our future work involves developing a tracking algorithm that addresses the unique challenges presented by the VETRA dataset.

References

1. Aharon, N., Orfaig, R., Bobrovsky, B.Z.: BoT-SORT: Robust Associations Multi-Pedestrian Tracking (2022), `https://arxiv.org/abs/2206.14651`
2. Azimi, S.M., Bahmanyar, R., Henry, C., Kurz, F.: EAGLE: Large-Scale Vehicle Detection Dataset in Real-World Scenarios using Aerial Imagery. In: Proceedings of ICPR 2021. pp. 6920–6927. IEEE (2021). `https://doi.org/10.1109/ICPR48806.2021.9412353`
3. Bai, H., Cheng, W., Chu, P., Liu, J., Zhang, K., Ling, H.: GMOT-40: A Benchmark for Generic Multiple Object Tracking. In: Proceedings of CVPR 2021. pp. 6715–6724. IEEE (2021). `https://doi.org/10.1109/CVPR46437.2021.00665`
4. Barmpounakis, E., Geroliminis, N.: On the new era of urban traffic monitoring with massive drone data: The pNEUMA large-scale field experiment. Transportation Research Part C: Emerging Technologies **111**, 50–71 (2020). `https://doi.org/10.1016/j.trc.2019.11.023`
5. Bewley, A., Ge, Z., Ott, L., Ramos, F., Upcroft, B.: Simple online and realtime tracking. In: Proceedings of ICIP 2016. pp. 3464–3468. IEEE (2016). `https://doi.org/10.1109/ICIP.2016.7533003`
6. Broström, M.: BoxMOT: pluggable SOTA tracking modules for object detection, segmentation and pose estimation models (2023), `https://zenodo.org/record/7629840`
7. Bundesministerium für Verkehr und digitale Infrastruktur: Richtlinien für die Straßenverkehrszählung 2020 im Jahre 2021 auf den Bundesfernstraßen (2020), `https://www.bast.de/DE/Statistik/Verkehrsdaten/2020/Richtlinien-2020.pdf?__blob=publicationFile&v=6`
8. Cao, J., Pang, J., Weng, X., Khirodkar, R., Kitani, K.: Observation-Centric SORT: Rethinking SORT for Robust Multi-Object Tracking. In: Proceedings of CVPR 2023. pp. 9686–9696. IEEE (2023)
9. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-End Object Detection with Transformers. In: Proceedings of ECCV 2020. Springer (2020). `https://doi.org/10.1007/978-3-030-58452-8_13`
10. Cetintas, O., Brasó, G., Leal-Taixé, L.: Unifying Short and Long-Term Tracking With Graph Hierarchies: SUSHI. In: Proceedings of CVPR 2023. pp. 22877–22887. IEEE (2023). `https://doi.org/10.1109/CVPR52729.2023.02191`
11. Cheng, G., Han, J., Zhou, P., Guo, L.: Multi-class geospatial object detection and geographic image classification based on collection of part detectors. ISPRS Journal of Photogrammetry and Remote Sensing **98**, 119–132 (2014). `https://doi.org/10.1016/j.isprsjprs.2014.10.002`
12. Cheng, G., Yuan, X., Yao, X., Yan, K., Zeng, Q., Xie, X., Han, J.: Towards Large-Scale Small Object Detection: Survey and Benchmarks. IEEE TPAMI **45**(11), 13467–13488 (2023). `https://doi.org/10.1109/TPAMI.2023.3290594`
13. Dave, A., Khurana, T., Tokmakov, P., Schmid, C., Ramanan, D.: TAO: A Large-Scale Benchmark for Tracking Any Object. In: Proceedings of ECCV 2020, pp. 436–454. Springer (2020). `https://doi.org/10.1007/978-3-030-58558-7_26`
14. Du, Y., Wan, J., Zhao, Y., Zhang, B., Tong, Z., Dong, J.: GIAOTracker: A comprehensive framework for MCMOT with global information and optimizing strategies in VisDrone. In: ICCV Workshops 2021. pp. 2809–2819. IEEE (2021). `https://doi.org/10.1109/ICCVW54120.2021.00315`
15. Hsieh, M.R., Lin, Y.L., Hsu, W.H.: Drone-Based Object Counting by Spatially Regularized Regional Proposal Network. In: Proceedings of ICCV 2017. pp. 4165–4173. IEEE (2017). `https://doi.org/10.1109/ICCV.2017.446`

16. Kiefer, B., Quan, Y., Zell, A.: Memory Maps for Video Object Detection and Tracking on UAVs. In: Proceedings of the International Conference on Intelligent Robots and Systems. pp. 3040–3047. IEEE (2023). https://doi.org/10.1109/IROS55552.2023.10342453
17. Kim, S., Anagnostopoulos, G., Barmpounakis, E., Geroliminis, N.: Visual extensions and anomaly detection in the pNEUMA experiment with a swarm of drones. Transportation Research Part C: Emerging Technologies **147** (2023). https://doi.org/10.1016/j.trc.2022.103966
18. Krajewski, R., Bock, J., Kloeker, L., Eckstein, L.: The highD Dataset: A Drone Dataset of Naturalistic Vehicle Trajectories on German Highways for Validation of Highly Automated Driving Systems. In: Intelligent Transportation Systems Conference. pp. 2118–2125. IEEE (2018). https://doi.org/10.1109/ITSC.2018.8569552
19. Kraus, M., Azimi, S.M., Erçelik, E., Bahmanyar, R., Reinartz, P., Knoll, A.: AerialMPTNet: Multi-Pedestrian Tracking in Aerial Imagery Using Temporal and Graphical Features. In: Proceedings of ICPR 2020. pp. 2454–2461. IEEE (2020). https://doi.org/10.1109/ICPR48806.2021.9413031
20. Kurz, F., Rosenbaum, D., Meynberg, O., Mattyus, G., Reinartz, P.: Performance of a real-time sensor and processing system on a helicopter. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences **XL-1**, 189–193 (2014). https://doi.org/10.5194/isprsarchives-XL-1-189-2014
21. Kurz, F., Türmer, S., Meynberg, O., Rosenbaum, D., Runge, H., Reinartz, P., Leitloff, J.: Low-cost optical Camera Systems for real-time Mapping Applications. Photogrammetrie - Fernerkundung - Geoinformation **2012**(2), 159–176 (2012). https://doi.org/10.1127/1432-8364/2012/0109
22. Lam, D., Kuzma, R., McGee, K., Dooley, S., Laielli, M., Klaric, M., Bulatov, Y., McCord, B.: xView: Objects in Context in Overhead Imagery (2018), https://arxiv.org/abs/1802.07856
23. Leutenegger, S., Chli, M., Siegwart, R.Y.: BRISK: Binary Robust Invariant Scalable Keypoints. In: Proceedings of ICCV 2011. pp. 2548–2555. IEEE (2011). https://doi.org/10.1109/ICCV.2011.6126542
24. Li, K., Wan, G., Cheng, G., Meng, L., Han, J.: Object detection in optical remote sensing images: A survey and a new benchmark: DIOR dataset. ISPRS Journal of Photogrammetry and Remote Sensing **159**, 296–307 (2020). https://doi.org/10.1016/j.isprsjprs.2019.11.023
25. Lin, H.Y., Tu, K.C., Li, C.Y.: VAID: An Aerial Image Dataset for Vehicle Detection and Classification. IEEE Access **8**, 212209–212219 (2020). https://doi.org/10.1109/ACCESS.2020.3040290
26. Liu, K., Máttyus, G.: Fast Multiclass Vehicle Detection on Aerial Images. IEEE Geoscience and Remote Sensing Letters **12**(9), 1938–1942 (2015). https://doi.org/10.1109/LGRS.2015.2439517
27. Liu, S., Li, X., Lu, H., He, Y.: Multi-Object Tracking Meets Moving UAV. In: Proceedings of CVPR 2022. pp. 8866–8875. IEEE (2022). https://doi.org/10.1109/CVPR52688.2022.00867
28. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin Transformer: Hierarchical Vision Transformer using Shifted Windows. In: Proceedings of ICCV 2021. IEEE (2021). https://doi.org/10.1109/ICCV48922.2021.00986
29. Liu, Z., Shang, Y., Li, T., Chen, G., Wang, Y., Hu, Q., Zhu, P.: Robust Multi-Drone Multi-Target Tracking to Resolve Target Occlusion: A Benchmark. IEEE

Transactions on Multimedia **25**, 1462–1476 (2023). https://doi.org/10.1109/TMM.2023.3234822

30. Luiten, J., Hoffhues, A.: TrackEval. https://github.com/JonathonLuiten/TrackEval (2020)
31. Maggiolino, G., Ahmad, A., Cao, J., Kitani, K.: Deep OC-SORT: Multi-Pedestrian Tracking by Adaptive Re-Identification. In: Proceedings of ICIP 2023. IEEE (2023). https://doi.org/10.1109/ICIP49359.2023.10222576
32. Mühlhaus, M., Kurz, F., Guridi Tartas, A.R., Bahmanyar, R., Azimi, S., Hellekes, J.: Vehicle classification in urban regions of the Global South from aerial imagery. ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences pp. 1–8 (2023). https://doi.org/10.5194/isprs-annals-X-1-W1-2023-371-2023
33. Mundhenk, T.N., Konjevod, G., Sakla, W.A., Boakye, K.: A Large Contextual Dataset for Classification, Detection and Counting of Cars with Deep Learning. In: Proceedings of ECCV 2016, vol. 9907, pp. 785–800. Springer (2016). https://doi.org/10.1007/978-3-319-46487-9_48
34. Rao, S., Böhle, M., Parchami-Araghi, A., Schiele, B.: Using Explanations to Guide Models (2023), https://arxiv.org/abs/2303.11932
35. Razakarivony, S., Jurie, F.: Vehicle detection in aerial imagery: A small target detection benchmark. Journal of Visual Communication and Image Representation **34**, 187–203 (2016). https://doi.org/10.1016/j.jvcir.2015.11.002
36. Rezatofighi, H., Tsoi, N., Gwak, J., Sadeghian, A., Reid, I., Savarese, S.: Generalized Intersection Over Union: A Metric and a Loss for Bounding Box Regression. In: Proceedings of CVPR 2019. pp. 658–666. IEEE (2019). https://doi.org/10.1109/CVPR.2019.00075
37. Schmidt, F.: Data Set for Tracking Vehicles in Aerial Image Sequences (2012), https://www.ipf.kit.edu/downloads_data_set_AIS_vehicle_tracking.php
38. Sun, P., Kretzschmar, H., Dotiwalla, X., Chouard, A., Patnaik, V., Tsui, P., Guo, J., Zhou, Y., Chai, Y., Caine, B., Vasudevan, V., Han, W., Ngiam, J., Zhao, H., Timofeev, A., Ettinger, S., Krivokon, M., Gao, A., Joshi, A., Zhang, Y., Shlens, J., Chen, Z., Anguelov, D.: Scalability in Perception for Autonomous Driving: Waymo Open Dataset. In: Proceedings of CVPR 2020. pp. 2443–2451. IEEE (2020). https://doi.org/10.1109/CVPR42600.2020.00252
39. Sun, Y., Cao, B., Zhu, P., Hu, Q.: Drone-Based RGB-Infrared Cross-Modality Vehicle Detection Via Uncertainty-Aware Learning. IEEE Transactions on Circuits and Systems for Video Technology **32**(10), 6700–6713 (2022). https://doi.org/10.1109/TCSVT.2022.3168279
40. Voigtlaender, P., Krause, M., Osep, A., Luiten, J., Sekar, B.B.G., Geiger, A., Leibe, B.: MOTS: Multi-Object Tracking and Segmentation. In: Proceedings of CVPR 2019. pp. 7934–7943. IEEE (2019). https://doi.org/10.1109/CVPR.2019.00813
41. Wang, Y.H., Hsieh, J.W., Chen, P.Y., Chang, M.C.: SMILEtrack: SiMIlarity LEarning for Multiple Object Tracking (2022), http://arxiv.org/pdf/2211.08824v2
42. Wen, L., Du, D., Cai, Z., Lei, Z., Chang, M.C., Qi, H., Lim, J., Yang, M.H., Lyu, S.: UA-DETRAC: A new benchmark and protocol for multi-object detection and tracking. Computer Vision and Image Understanding **193** (2020). https://doi.org/10.1016/j.cviu.2020.102907
43. Wojke, N., Bewley, A., Paulus, D.: Simple online and realtime tracking with a deep association metric: DeepSORT. In: Proceedings of ICIP 2017. pp. 3645–3649. IEEE (2017). https://doi.org/10.1109/ICIP.2017.8296962

44. Xia, G.S., Bai, X., Ding, J., Zhu, Z., Belongie, S., Luo, J., Datcu, M., Pelillo, M., Zhang, L.: DOTA: A Large-Scale Dataset for Object Detection in Aerial Images. In: Proceedings of CVPR 2018. pp. 3974–3983. IEEE (2018). https://doi.org/10.1109/CVPR.2018.00418
45. Xu, C., Wang, J., Yang, W., Yu, H., Yu, L., Xia, G.S.: Detecting tiny objects in aerial images: A normalized Wasserstein distance and a new benchmark. ISPRS Journal of Photogrammetry and Remote Sensing **190**, 79–93 (2022). https://doi.org/10.1016/j.isprsjprs.2022.06.002
46. Xu, X., Feng, Z., Cao, C., Yu, C., Li, M., Wu, Z., Ye, S., Shang, Y.: STN-Track: Multiobject Tracking of Unmanned Aerial Vehicles by Swin Transformer Neck and New Data Association Method. Journal of Selected Topics in Applied Earth Observations and Remote Sensing **15**, 8734–8743 (2022). https://doi.org/10.1109/JSTARS.2022.3213438
47. Yao, M., Wang, J., Peng, J., Chi, M., Liu, C.: FOLT: Fast Multiple Object Tracking from UAV-Captured Videos Based on Optical Flow. In: ACM International Conference on Multimedia. p. 3375–3383 (2023). https://doi.org/10.1145/3581783.3611868
48. Yu, F., Chen, H., Wang, X., Xian, W., Chen, Y., Liu, F., Madhavan, V., Darrell, T.: BDD100K: A Diverse Driving Dataset for Heterogeneous Multitask Learning. In: Proceedings of CVPR 2020. pp. 2633–2642. IEEE (2020). https://doi.org/10.1109/CVPR42600.2020.00271
49. Yu, H., Li, G., Zhang, W., Huang, Q., Du, D., Tian, Q., Sebe, N.: The Unmanned Aerial Vehicle Benchmark: Object Detection, Tracking and Baseline. International Journal of Computer Vision **128**(5), 1141–1159 (2020). https://doi.org/10.1007/s11263-019-01266-1
50. Zeng, F., Dong, B., Zhang, Y., Wang, T., Zhang, X., Wei, Y.: MOTR: End-to-End Multiple-Object Tracking with Transformer. In: Proceedings of ECCV 2022 (2022). https://doi.org/10.1007/978-3-031-19812-0_38
51. Zhang, H., Li, F., Liu, S., Zhang, L., Su, H., Zhu, J., Ni, L.M., Shum, H.Y.: DINO: DETR with Improved DeNoising Anchor Boxes for End-to-End Object Detection (2022), https://arxiv.org/abs/2203.03605
52. Zhang, Y., Sun, P., Jiang, Y., Yu, D., Weng, F., Yuan, Z., Luo, P., Liu, W., Wang, X.: ByteTrack: Multi-object Tracking by Associating Every Detection Box. In: Proceedings of ECCV 2022. Springer (2022). https://doi.org/10.1007/978-3-031-20047-2_1
53. Zhang, Y., Wang, T., Zhang, X.: MOTRv2: Bootstrapping End-to-End Multi-Object Tracking by Pretrained Object Detectors. In: Proceedings of CVPR 2023. pp. 22056–22065. Springer (2023). https://doi.org/10.1109/CVPR52729.2023.02112
54. Zheng, Z., Wang, P., Liu, W., Li, J., Ye, R., Ren, D.: Distance-IoU Loss: Faster and Better Learning for Bounding Box Regression. In: Proceedings of the Conference on Artificial Intelligence (2020). https://doi.org/10.1609/aaai.v34i07.6999
55. Zhou, K., Yang, Y., Cavallaro, A., Xiang, T.: Learning Generalisable Omni-Scale Representations for Person Re-Identification. IEEE TPAMI **44**(09), 5056–5069 (2022). https://doi.org/10.1109/TPAMI.2021.3069237
56. Zhu, H., Chen, X., Dai, W., Fu, K., Ye, Q., Jiao, J.: Orientation robust object detection in aerial images using deep convolutional neural network. In: Proceedings of ICIP 2015. pp. 3735–3739. IEEE (2015). https://doi.org/10.1109/ICIP.2015.7351502

57. Zhu, P., Wen, L., Du, D., Bian, X., Fan, H., Hu, Q., Ling, H.: Detection and Tracking Meet Drones Challenge: VisDrone dataset. IEEE TPAMI pp. 7380–7399 (2022). https://doi.org/10.1109/TPAMI.2021.3119563

6 Supplementary material

6.1 Acquisition sites of VETRA dataset

The acquisition sites are located in Germany and Austria, the spatial distribution for each subset is displayed in Fig. 8.

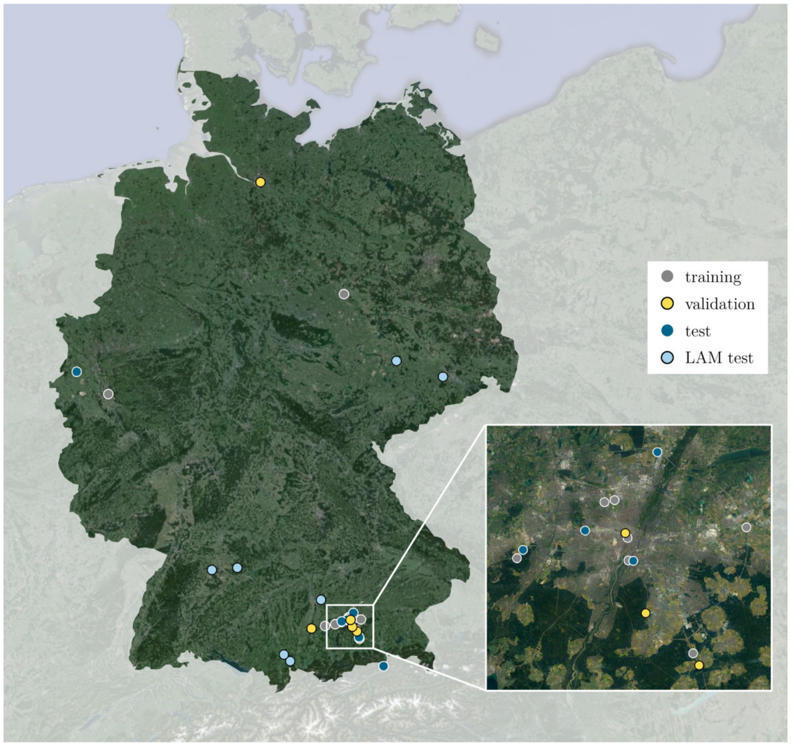


Fig. 8: Locations where VETRA sequences were captured. For LAM test set, each data point represents four sequences.

6.2 Mosaics and annotations of VETRA dataset

Precise geo-information is available for each image of the VETRA dataset. Mosaics are generated by overlaying frames per sequence in chronological order. For the sequences that are discussed in the main paper, the corresponding mosaics are shown in Fig. 10 to 22. For the sequences used for training, validation and test, each vehicle trajectory is plotted in a randomly assigned color; classes and attributes are not shown. The 'LAM_Stuttgart' sequences (Fig. 22) illustrate exemplarily that the traffic situation can differ greatly between revisits despite an unchanged environment. To increase the visibility of vehicle densities, the annotations are not overlaid.

6.3 Sequence-wise statistics of VETRA dataset

The statistics presented in Fig. 3 and 4 and as well as Tab. 2 and 5 characterize the VETRA dataset as a whole and give more detailed information for the validation, test and LAM test sets. Tab. 7 provides for each sequence statistics like the spatial resolution, frame rate, camera movement, and duration.

6.4 Vehicle classification schemes of VETRA dataset

For the training, validation and test subsets of VETRA, the polygon annotations are provided on the atomic level (as introduced in Tab. 1). This scheme is flexible and allows for further aggregation (see Fig. 9): the atomic units (top) can be translated into detailed (middle) and more aggregated (bottom) official classification schemes as used in Germany and other countries. The LAM test set is application driven; therefore, the vehicles in these sequences are directly labeled according to the middle scheme.

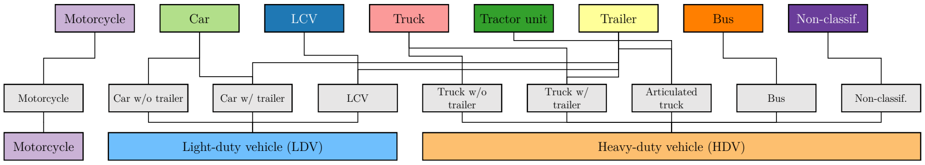


Fig. 9: The proposed vehicle classification scheme of VETRA captures all motorized road-based vehicles and allows for aggregation on various levels.

6.5 Samples of MOT datasets for vehicle tracking in aerial imagery

As reflected by Tab. 2, most of the existing MOT datasets for vehicle tracking in aerial imagery are acquired by UAV with a limited camera movement and a medium number of vehicle instances per frame due to the flight altitude. In addition, the vehicle motion is captured at high frame rates which facilitates the association in subsequent images. To help understand the features present in the current datasets and the diversity that VETRA adds to this, three consecutive frames from one sequence (focus on road traffic, randomly selected) per dataset are shown in Fig. 23 to 28. Only publicly accessible images are shown. All images are resized to optimize file size.

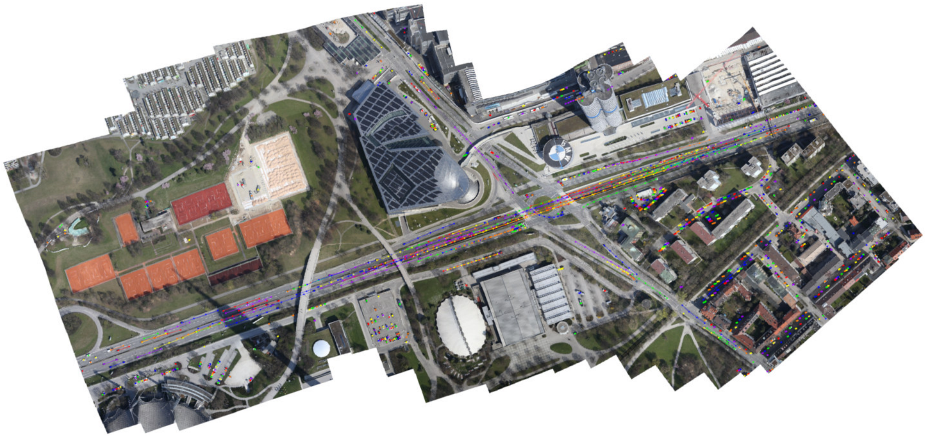


Fig. 10: Mosaic and annotations of sequence 04.

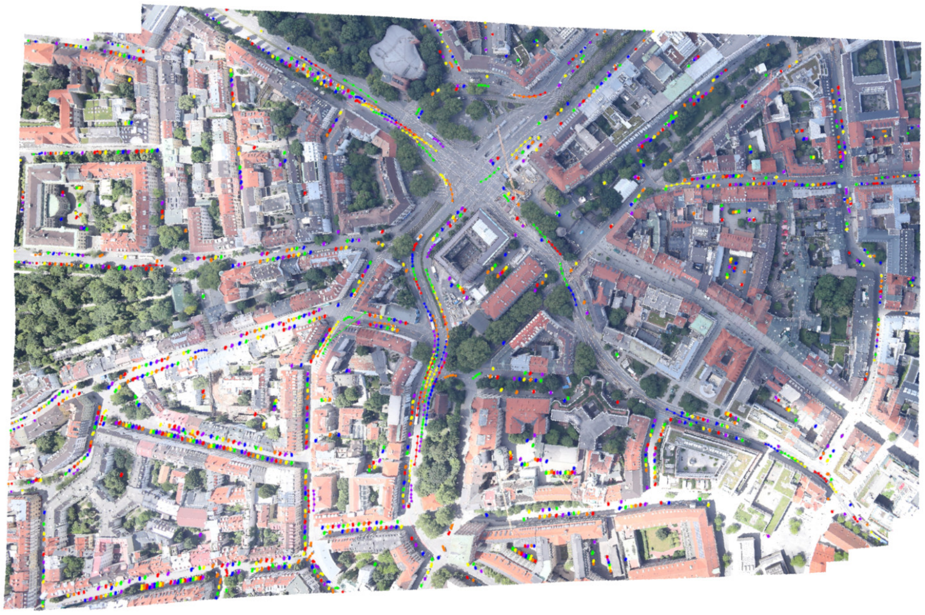


Fig. 11: Mosaic and annotations of sequence 07.

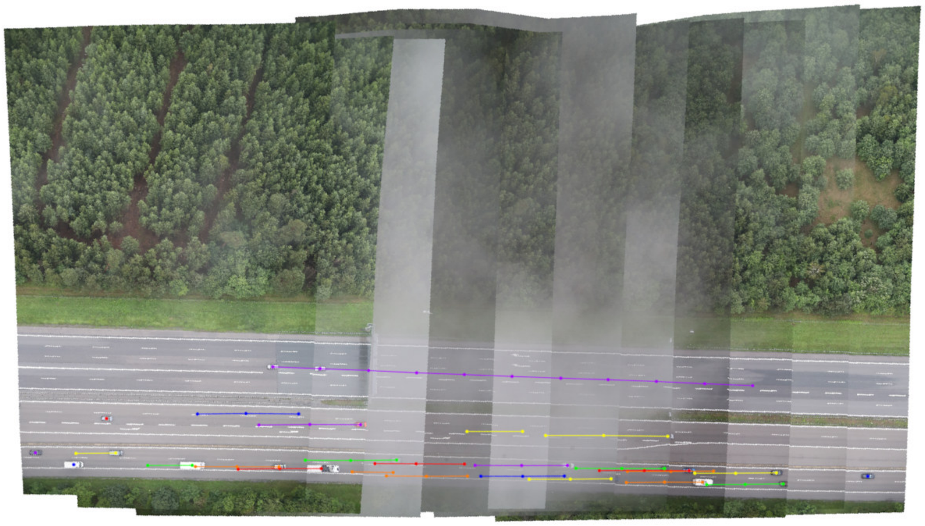


Fig. 12: Mosaic and annotations of sequence 09.

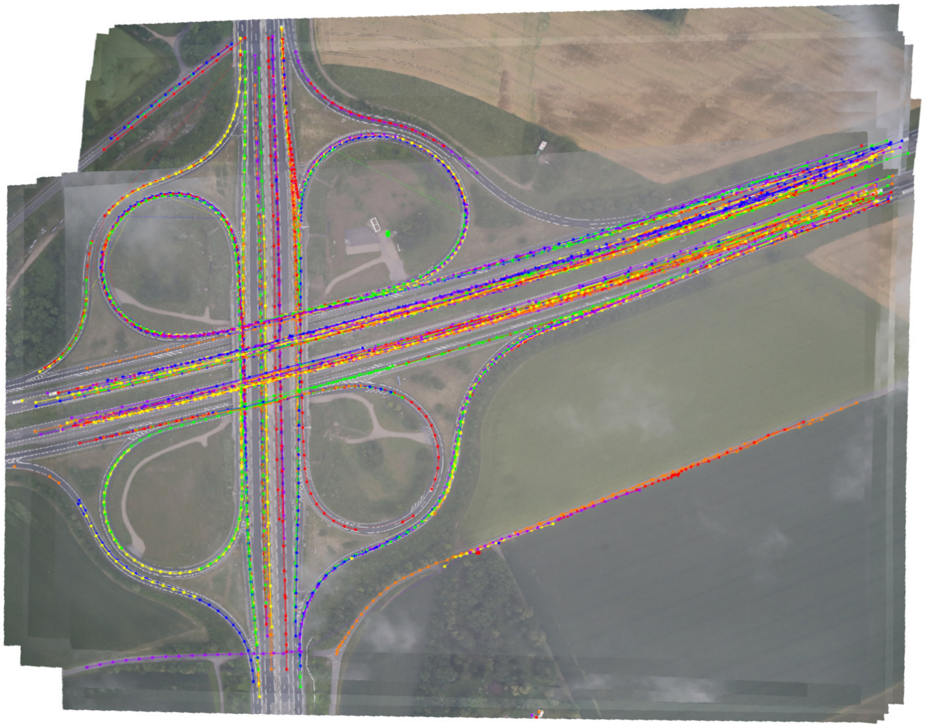


Fig. 13: Mosaic and annotations of sequence 10.

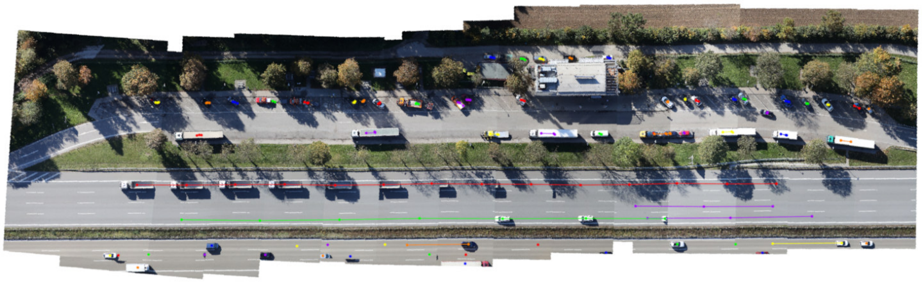


Fig. 14: Mosaic and annotations of sequence 14.

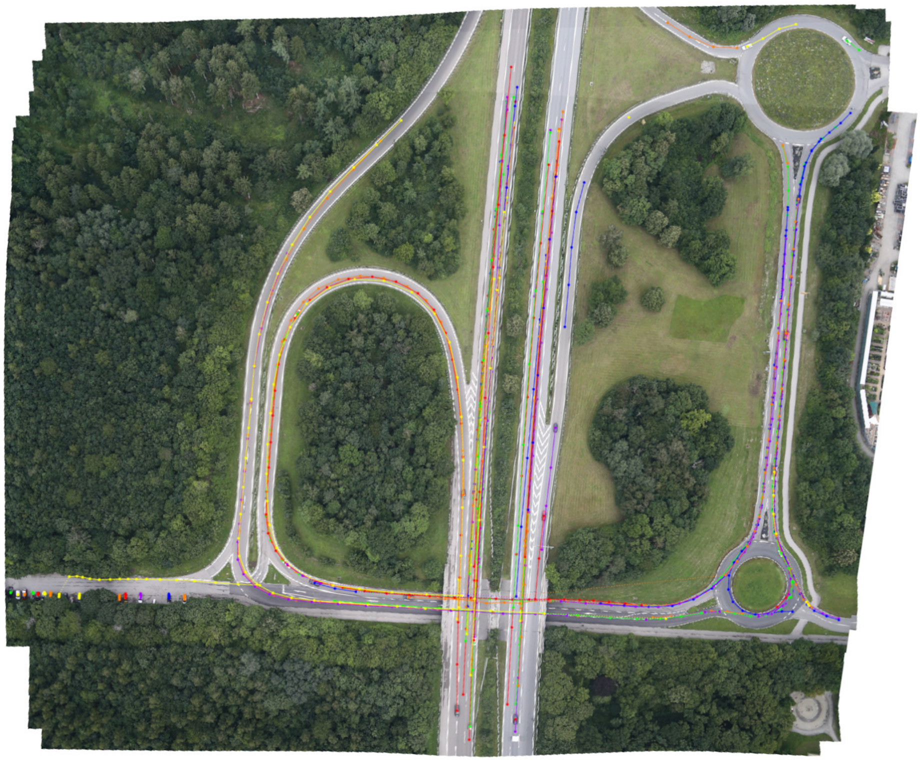


Fig. 15: Mosaic and annotations of sequence 16.

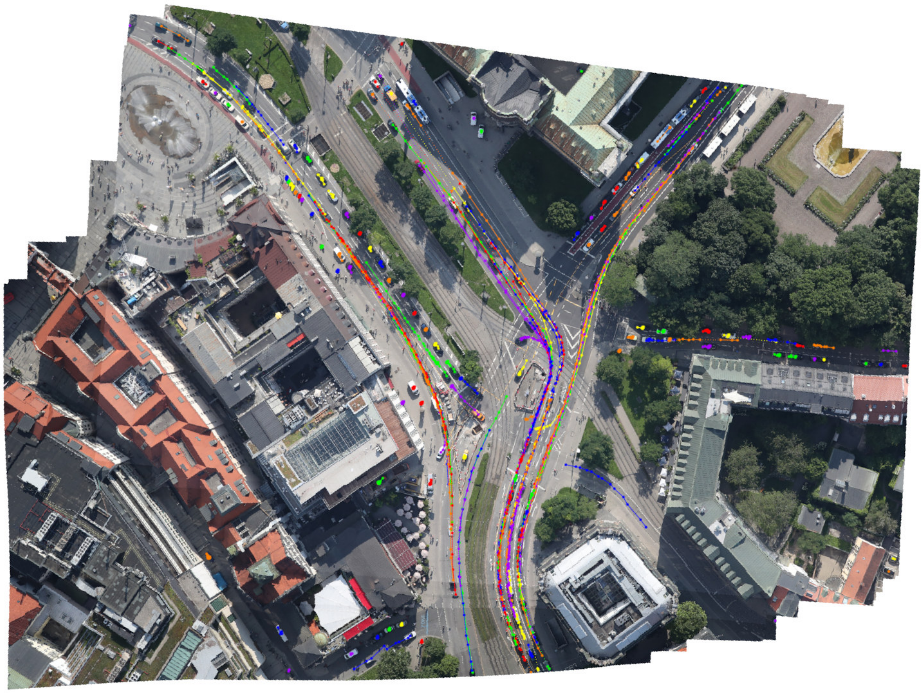


Fig. 16: Mosaic and annotations of sequence 17.

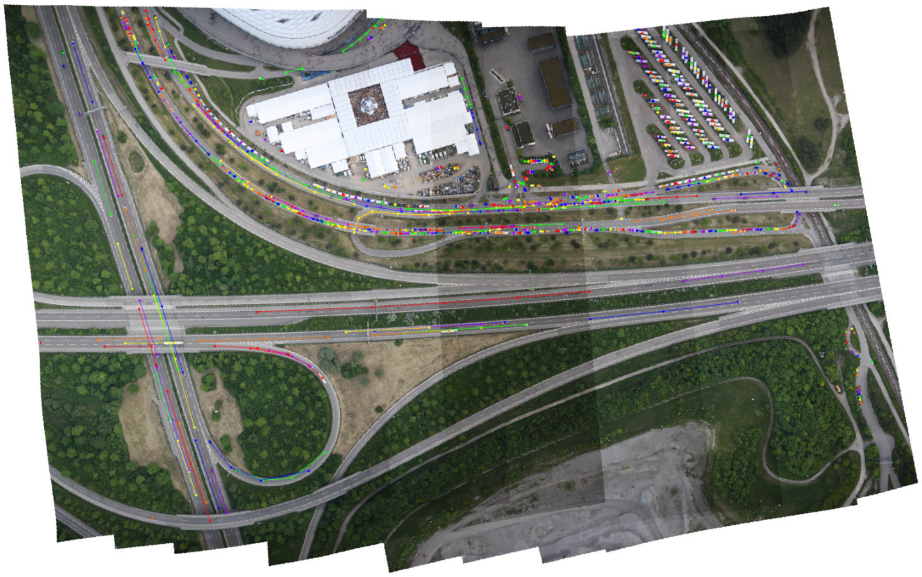


Fig. 17: Mosaic and annotations of sequence 18.

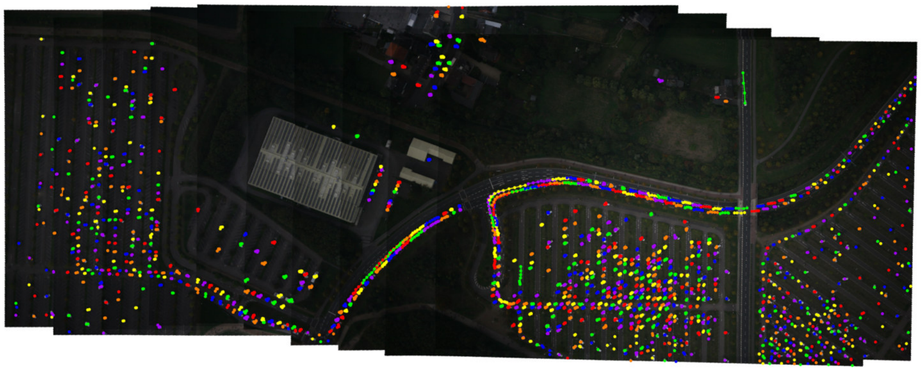


Fig. 18: Mosaic and annotations of sequence 20.

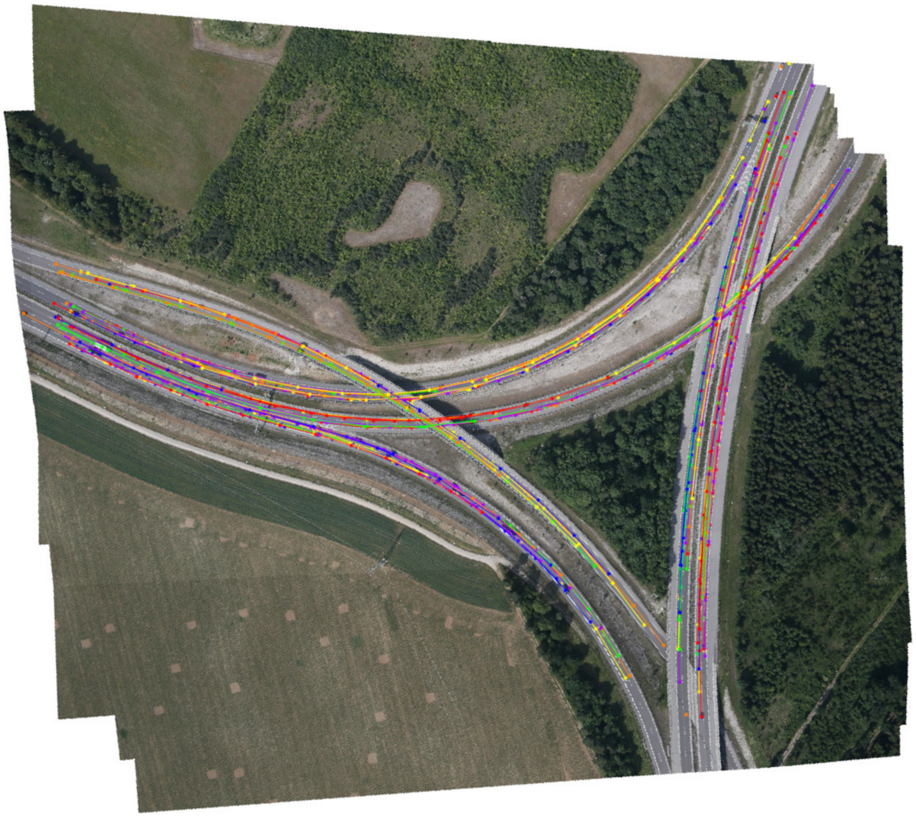


Fig. 19: Mosaic and annotations of sequence 21.

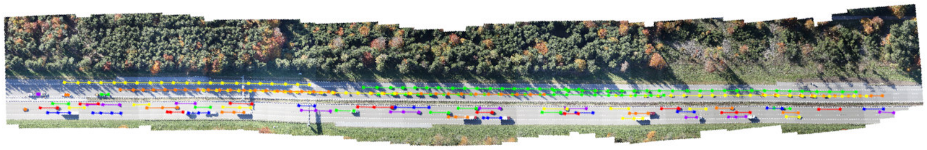


Fig. 20: Mosaic and annotations of sequence 23.

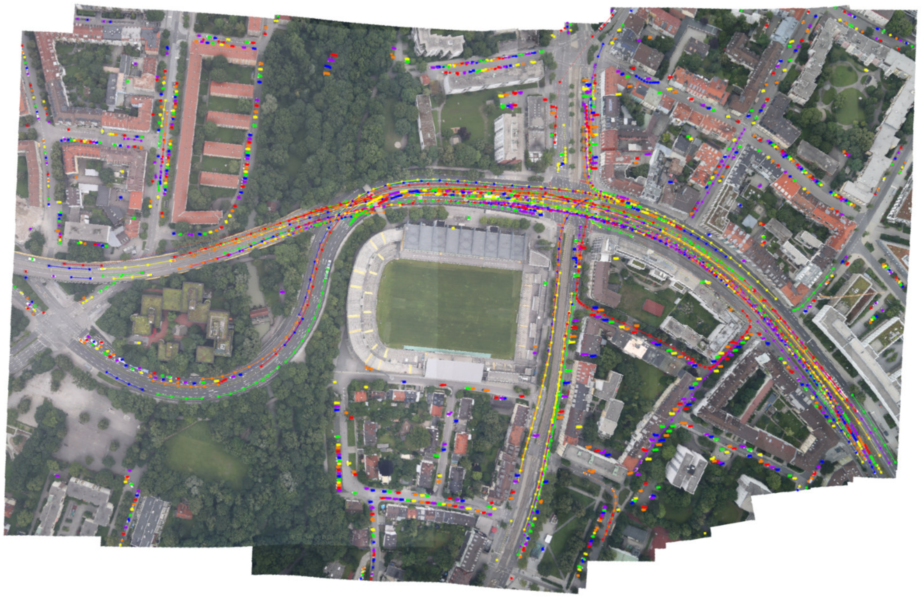
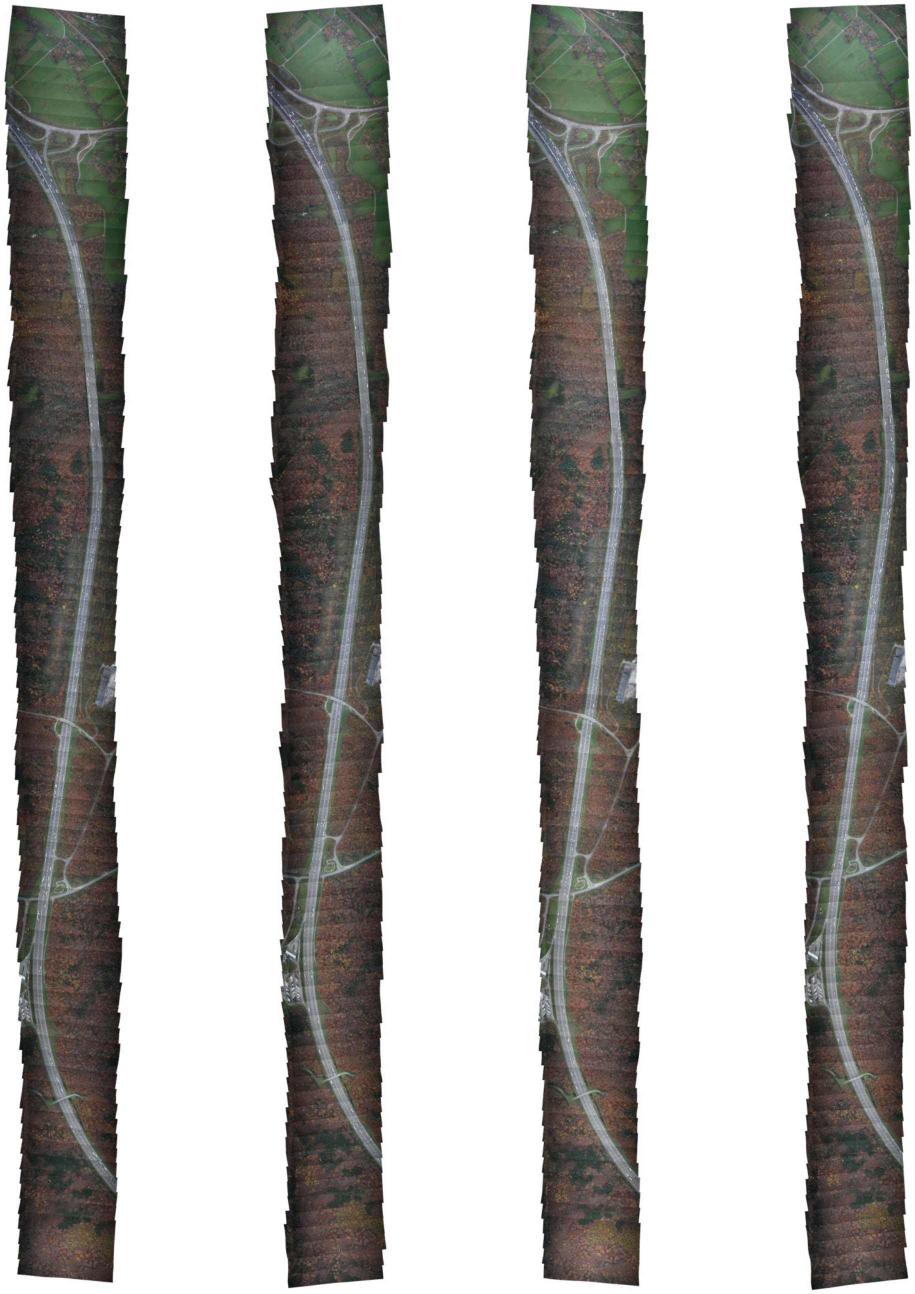


Fig. 21: Mosaic and annotations of sequence 24.

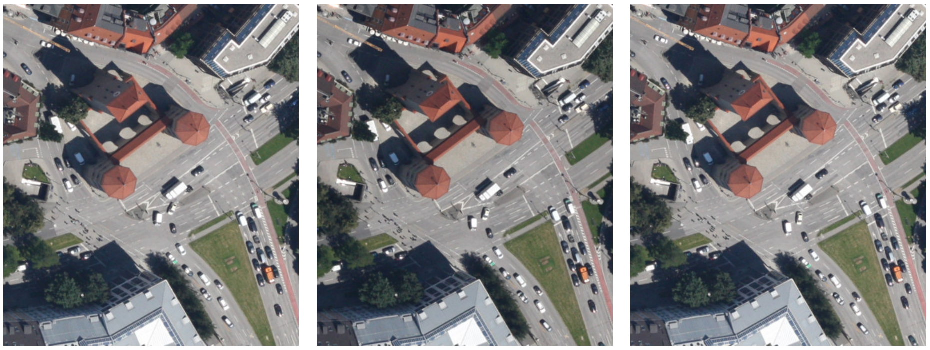


(a) Revisit 1 **(b)** Revisit 2 **(c)** Revisit 3 **(d)** Revisit 4

Fig. 22: Mosaics of LAM test sequences 41 to 44.

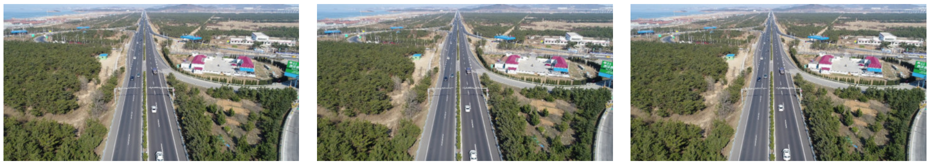
Table 7: Sequence-wise statistics of VETRA dataset.

Set	ID	Sequence	GSD (cm)	FPS (Hz)	Overlap (%)	Length (s)
Training	01	Cologne_lake	20	0.56	55	22
	02	Magdeburg_pool	14	0.40	80	16
	03	Munich_office_buildings	8	0.98	84	18
	04	Munich_tennis_BMW	8	0.98	77	22
	05	Holzkirchen_VHR_gas_station	2	2.38	73	34
	06	Munich_pavement_renewal	6	1.27	81	12
	07	Munich_city_center	14	1.58	97	4
	08	Munich_raised_highway	8	12.10	98	2
	09	Munich_aquaplaning	4	1.46	80	8
	10	Munich_cloverleaf	12	1.45	98	49
	11	Greifenberg_camera_rotation	5	1.45	95	17
	12	Gilching_motorway_ramp	5	0.74	77	40
Validation	13	Landsberg_roundabout	13	0.56	80	29
	14	Munich_VHR_motorway_service	2	1.45	64	10
	15	Hamburg_river_bridge	11	1.01	78	11
	16	Munich_test_vehicle	8	0.98	91	50
	17	Munich_multimodal_crossing	4	1.87	83	33
Test	18	Munich_stadium	10	0.42	70	19
	19	Munich_railroad	14	1.11	88	6
	20	Moenchengladbach_parking	8	1.67	89	7
	21	Munich_Y_interchange	10	0.76	90	27
	22	Kufstein_river	16	1.00	86	18
	23	Holzkirchen_VHR_motorway	3	2.39	80	23
	24	Munich_tunnel	9	0.98	88	33
LAM test	25	LAM_Augsburg_revisit_1	9	1.00	79	279
	26	LAM_Augsburg_revisit_2	9	1.00	79	280
	27	LAM_Augsburg_revisit_3	9	1.00	79	278
	28	LAM_Augsburg_revisit_4	9	1.00	79	280
	29	LAM_Nesselwang_revisit_1	9	1.00	79	101
	30	LAM_Nesselwang_revisit_2	9	1.00	79	100
	31	LAM_Nesselwang_revisit_3	9	1.00	79	99
	32	LAM_Nesselwang_revisit_4	9	1.00	79	101
	33	LAM_Kempten_revisit_1	9	1.00	79	71
	34	LAM_Kempten_revisit_2	9	1.00	79	72
	35	LAM_Kempten_revisit_3	9	1.00	79	72
	36	LAM_Kempten_revisit_4	9	1.00	79	76
	37	LAM_Grunbach_revisit_1	10	1.00	79	75
	38	LAM_Grunbach_revisit_2	10	1.00	79	73
	39	LAM_Grunbach_revisit_3	10	1.00	79	74
	40	LAM_Grunbach_revisit_4	10	1.00	79	73
	41	LAM_Stuttgart_revisit_1	9	1.00	79	80
	42	LAM_Stuttgart_revisit_2	9	1.00	79	81
	43	LAM_Stuttgart_revisit_3	9	1.00	79	82
	44	LAM_Stuttgart_revisit_4	9	1.00	79	79
	45	LAM_Ammelshain_revisit_1	10	1.00	79	104
	46	LAM_Ammelshain_revisit_2	10	1.00	79	107
	47	LAM_Ammelshain_revisit_3	10	1.00	79	105
	48	LAM_Ammelshain_revisit_4	10	1.00	79	106
	49	LAM_Rothschoenberg_revisit_1	10	1.00	79	103
	50	LAM_Rothschoenberg_revisit_2	10	1.00	79	103
	51	LAM_Rothschoenberg_revisit_3	10	1.00	79	101
	52	LAM_Rothschoenberg_revisit_4	10	1.00	79	109



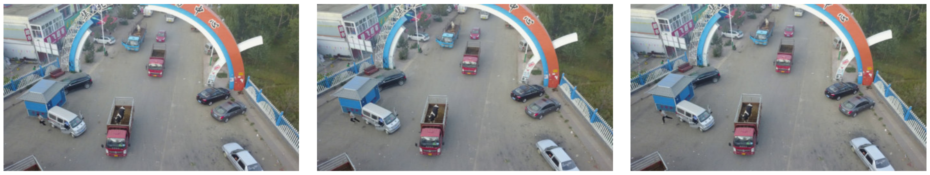
(a) Frame 'MOS37' **(b)** Frame 'MOS38' **(c)** Frame 'MOS39'

Fig. 23: Segment of sequence 'MunichCrossroad02' of KIT AIS dataset [37].



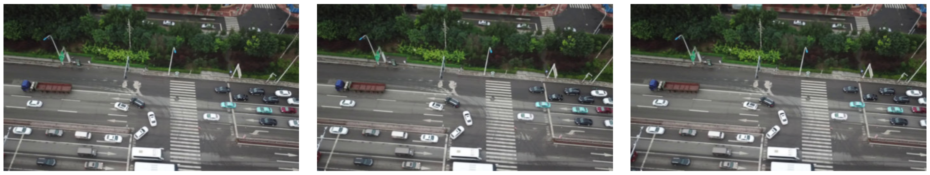
(a) Frame 'img000169' **(b)** Frame 'img000170' **(c)** Frame 'img000171'

Fig. 24: Segment of sequence 'M0201' of UAVDT dataset [49].



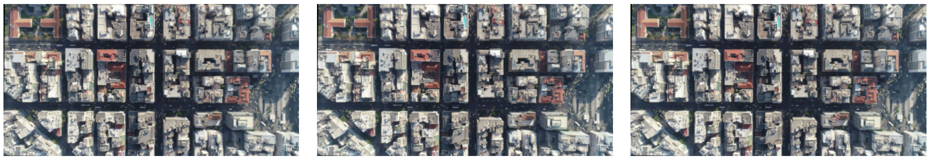
(a) Frame ‘0000144’ **(b)** Frame ‘0000145’ **(c)** Frame ‘0000146’

Fig. 25: Segment of sequence ‘uav0000077_02880_v’ of VisDrone dataset [57].



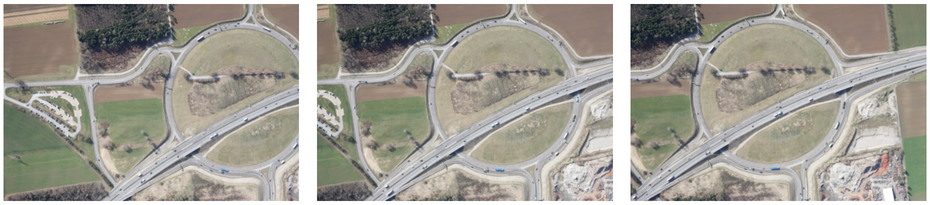
(a) Frame ‘00000044’ **(b)** Frame ‘00000045’ **(c)** Frame ‘00000046’

Fig. 26: Segment of sequence ‘48-1’ of MDMT dataset [29].



(a) Frame ‘00578’ **(b)** Frame ‘00579’ **(c)** Frame ‘00580’

Fig. 27: Segment of sequence ‘20181029_D6_0900_0930’ of pNEUMA dataset [17].



(a) Frame ‘13_005’ **(b)** Frame ‘13_006’ **(c)** Frame ‘13_007’

Fig. 28: Segment of sequence ‘13_Landsberg_roundabout’ of VETRA dataset (ours).