ROAD: The ROad event Awareness Dataset for Autonomous Driving

Gurkirt Singh[3], Stephen Akrigg[1], Manuele Di Maio[5], Valentina Fontana[2], Reza Javanmard Alitappeh[4] Salman Khan[1], Suman Saha[3], Kossar Jeddisaravi[4] Farzad Yousefi[4] Jacob Culley[1], Tom Nicholson[1], Jordan Omokeowa[1], Stanislao Grazioso[2], Andrew Bradley[1], Giuseppe Di Gironimo[2], Fabio Cuzzolin[1]

Abstract—Humans drive in a holistic fashion which entails, in particular, understanding dynamic road events and their evolution. Injecting these capabilities in autonomous vehicles can thus take situational awareness and decision making closer to human-level performance. To this purpose, we introduce the ROad event Awareness Dataset (ROAD) for Autonomous Driving, to our knowledge the first of its kind. ROAD is designed to test an autonomous vehicle's ability to detect road events, defined as triplets composed by an active agent, the action(s) it performs and the corresponding scene locations. ROAD comprises videos originally from the Oxford RobotCar Dataset, annotated with bounding boxes showing the location in the image plane of each road event. We benchmark various detection tasks, proposing as a baseline a new incremental algorithm for online road event awareness termed 3D-RetinaNet. We also report the performance on the ROAD tasks of Slowfast and YOLOv5 detectors, as well as that of the winners of the ICCV2021 ROAD challenge, which highlight the challenges faced by situation awareness in autonomous driving. ROAD is designed to allow scholars to investigate exciting tasks such as complex (road) activity detection, future event anticipation and continual learning. The dataset is available at https://github.com/gurkirt/road-dataset; the baseline can be found at https://github.com/gurkirt/3D-RetinaNet.

Index Terms—Autonomous driving, action detection, road agents, situation awareness, decision making.

1 Introduction

In recent years, *autonomous driving* (or *robot-assisted driving*) has emerged as a fast-growing research area. The race towards fully autonomous vehicles pushed many large companies, such as Google, Toyota and Ford, to develop their own concept of *robot-car* [1], [2], [3]. While self-driving cars are widely considered to be a major development and testing ground for the real-world application of artificial intelligence, major reasons for concern remain in terms of safety, ethics, cost, and reliability [4]. From a safety standpoint, in particular, smart cars need to robustly interpret the behaviour of the humans (drivers, pedestrians or cyclists) they share the environment with, in order to cope with their decisions. *Situation awareness* and the ability to understand the behaviour of other road users are thus crucial for the safe deployment of autonomous vehicles (AVs).

The latest generation of robot–cars is equipped with a range of different sensors (i.e., laser rangefinders, radar, cameras, GPS) to provide data on what is happening on the road [5]. The information so extracted is then fused to suggest how the vehicle should move [6], [7], [8], [9]. Some authors, however, maintain that vision is a sufficient sense for AVs to navigate their environment, supported by humans' ability to do just so. Without enlisting ourselves as supporters of the latter point of view, in this paper we consider the context of *vision-based* autonomous driving [10] from video sequences captured by cameras mounted on the vehicle in a streaming, online fashion.

While detector networks [11] are routinely trained to facilitate object and actor recognition in road scenes, this simply allows the vehicle to 'see' what is around it. The philosophy of this work is that robust self-driving capabilities require a deeper, more human-like understanding of dynamic road environments (and of the evolving behaviour of other road users over time) in the form of semantically meaningful concepts, as a stepping stone for intention prediction and automated decision making. One advantage of this approach is that it allows the autonomous vehicle to focus on a much smaller amount of relevant information when learning how to make its decisions, in a way arguably closer to how decision making takes place in humans.

On the opposite side of the spectrum lies end-to-end reinforcement learning. There, the behaviour of a human driver in response to road situations is used to train, in an imitation learning setting [12], an autonomous car to respond in a more 'human-like' manner to road scenarios. This, however, requires an astonishing amount of data from a myriad of road situations. For highway driving only, a relatively simple task when compared to city driving, Fridman et al. in [13] had to use a whole fleet of vehicles to collect 45 million frames. Perhaps more importantly, in this approach the network learns a mapping from the scene to control inputs, without attempting to model the significant facts taking place in the scene or the reasoning of the agents therein. As discussed in [14], many authors [15], [16] have recently highlighted the insufficiency of models which

- [1]*VAIL, Oxford Brookes University, UK,* [2] *University of Naples Federico II, Italy,* [3] *CVL, ETH Zurich,* [4] *University of Science and Technology of Mazandaran, Behshahr, Iran,* [5]*Siemens SpA, Bologna, Italy. This collaborative work took place at Oxford Brookes University when V. Fontana and M. Di Maio were visiting students, and S. Akrigg and G. Singh were students there. E-mail: gurkirt.singh@vision.ee.ethz.ch, fabio.cuzzolin@brookes.ac.uk.*

Manuscript received XXX, 2020.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2022.3150906, IEEE Transactions on Pattern Analysis and Machine Intelligence

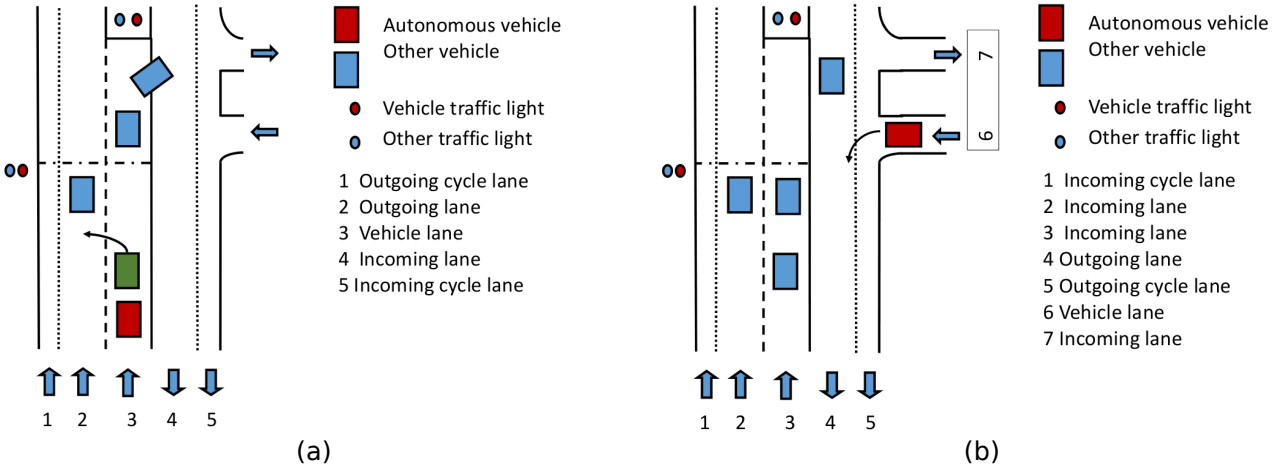


Fig. 1. Use of labels in ROAD to describe typical road scenarios. (a) A green car is in front of the AV while changing lanes, as depicted by the arrow symbol. The associated event will then carry the following labels: *in vehicle lane* (location), *moving left* (action). Once the event is completed, the location label will change to: *in outgoing lane*. (b) Autonomous vehicle turning left from lane 6 into lane 4: lane 4 will be the *outgoing lane* as the traffic is moving in the same direction as the AV. However, if the AV turns right from lane 6 into lane 4 (a wrong turn), then lane 4 will become the *incoming lane* as the vehicle will be moving into the incoming traffic. The overall philosophy of ROAD is to use suitable combinations of multiple label types to fully describe a road situation, and allow a machine learning algorithm to learn from this information.

directly map observations to actions [17], specifically in the self-driving cars scenario.

1.1 ROAD: a multi-label, multi-task dataset

Concept. This work aims to propose a new framework for situation awareness and perception, departing from the disorganised collection of object detection, semantic segmentation or pedestrian intention tasks which is the focus of much current work. We propose to do so in a "holistic", multi-label approach in which agents, actions and their locations are all ingredients in the fundamental concept of *road event* (RE). Road events are defined as triplets $E = (Ag, Ac, Loc)$ composed by an active road agent Ag, the action(s) Ac it performs (possibly more than one at the same time), and the location(s) Loc in which this takes place (which may vary from the start to the end of the event itself), as seen from the point of the view of an autonomous vehicle.
This takes the problem to a higher conceptual level, in which AVs are tested on their *understanding of what is going on* in a dynamic scene rather than their ability to describe what the scene *looks like*, putting them in a position to use that information to make decisions and a plot course of action. Modelling dynamic road scenes in terms of road events can also allow us to model the causal relationships between what happens; these causality links can then be exploited to predict further future consequences.

To transfer this conceptual paradigm into practice, this paper introduces ROAD, the first *ROad event Awareness in Autonomous Driving Dataset*, as an entirely new type of dataset designed to allow researchers in autonomous vehicles to test the situation awareness capabilities of their stacks in a manner impossible until now. Unlike all existing benchmarks, ROAD provides ground truth for the action performed by all road agents, not just humans. In this sense ROAD is unique in the richness and sophistication of its annotation, designed to support the proposed conceptual shift. We are confident this contribution will be very useful moving forward for both the autonomous driving and the computer vision community.

Features. ROAD is built upon (a fraction of) the Oxford RobotCar Dataset [18], by carefully annotating 22 carefully selected, relatively long-duration videos. Road events are represented as 'tubes', i.e., time series of frame-wise bounding box detections. ROAD is a dataset of significant size, most notably in terms of the richness and complexity of its annotation rather than the raw number of video frames. A total of $122K$ video frames are labelled for a total of $560K$ detection bounding boxes in turn associated with $1.7M$ unique individual labels, broken down into $560K$ agent labels, $640K$ action labels and $499K$ location labels.

The dataset was designed according to the following principles.

- A *multi-label* benchmark: each road event is composed by the label of the (moving) agent responsible, the label(s) of the type of action(s) being performed, and labels describing where the action is located.
- Each event can be assigned *multiple instances* of the same label type whenever relevant (e.g., an RE can be an instance of both *moving away* and *turning left*).
- The labelling is done *from the point of view of the AV*: the final goal is for the autonomous vehicle to use this information to make the appropriate decisions.
- The meta-data is intended to contain all the information required to fully describe a road scenario: an illustration of this concept is given in Figure 1. After closing one's eyes, the set of labels associated with the current video frame should be sufficient to recreate the road situation in one's head (or, equivalently, sufficient for the AV to be able to make a decision).

In an effort to take action detection into the real world, ROAD moves away from human body actions almost entirely, to consider (besides pedestrian behaviour) actions performed by humans as drivers of various types of ve-

hicles, shifting the paradigm from *actions performed by human bodies* to *events caused by agents*. As shown in our experiments, ROAD is more challenging than current action detection benchmarks due to the complexity of road events happening in real, non-choreographed driving conditions, the number of active agents present and the variety of weather conditions encompassed.

Tasks. ROAD allows one to validate manifold tasks associated with situation awareness for self-driving, each associated with a label type (agent, action, location) or combination thereof: *spatiotemporal* (i) *agent detection*, (ii) *action detection*, (iii) *location detection*, (iv) *agent-action detection*, (v) *road event detection*, as well as the (vi) *temporal segmentation of AV actions*. For each task one can assess both *frame-level* detection, which outputs independently for each video frame the bounding box(es) (BBs) of the instances there present and the relevant class labels, and *video-level* detection, which consists in regressing the whole series of temporally-linked bounding boxes (i.e., in current terminology, a 'tube') associated with an instance, together with the relevant class label. In this paper we conduct tests on both. All tasks come with both the necessary annotation and a shared baseline, which is described in Section 4.

1.2 Contributions

The major contributions of the paper are thus the following.

- A conceptual shift in situation awareness centred on a formal definition of the notion of road event, as a triplet composed by a road agent, the action(s) it performs and the location(s) of the event, seen from the point of view of the AV.
- A new ROad event Awareness Dataset for Autonomous Driving (ROAD), the first of its kind, designed to support this paradigm shift and allow the testing of a range of tasks related to situation awareness for autonomous driving: agent and/or action detection, event detection, ego-action classification.

Instrumental to the introduction of ROAD as the benchmark of choice for semantic situation awareness, we propose a robust baseline for online action/agent/event detection (termed *3D-RetinaNet*) which combines state-of-the-art single-stage object detector technology with an online tube construction method [19], with the aim of linking detections over time to create *event tubes* [20], [21]. Results for two additional baselines based on a Slowfast detector architecture [22] and YOLOv5[1] (for agent detection only) are also reported and critically assessed.

We are confident that this work will lay the foundations upon which much further research in this area can be built.

1.3 Outline

The remainder of the paper is organised as follows. Section 2 reviews related work concerning existing datasets, both for autonomous driving (Sec. 2.1) and action detection (Sec. 2.2), as well as action detection methods (Sec. 2.3). Section 3 presents our ROAD dataset in full detail, including: its multi-label nature (Sec. 3.1), data collection (Sec. 3.2), annotation (Sec. 3.3), the tasks it is designed to validate (Sec. 3.4), and a quantitative summary (Sec. 3.5). Section 4 presents an overview of the proposed 3D-RetinaNet baseline, and recalls the ROAD challenge organised by some of us at ICCV 2021 to disseminate this new approach to situation awareness within the autonomous driving and computer vision communities, using ROAD as the benchmark. Experiments are described in Section 5, where a number of ablation studies are reported and critically analysed in detail, together with the results of the ROAD challenge's top participants. Section 6 outlines additional exciting tasks the dataset can be used as a benchmark for in the near future, such as future event anticipation, decision making and machine theory of mind [14]. Conclusions and future work are outlined in Section 7.

The Supplementary material reports detailed class-wise results, a qualitative analysis of success and failure cases, and a link to a 30-minute footage visually illustrating the baseline's predictions versus the ground truth.

2 RELATED WORK

2.1 Autonomous driving datasets

In recent years a multitude of AV datasets have been released, mostly focusing on object detection and scene segmentation. We can categorise them into two main bins: (1) RGB without range data (single modality) and (2) RGB with range data (multimodal).

Single-modality datasets. Collecting and annotating RGB data only is relatively less time-consuming and expensive than building multimodal datasets including range data from LiDAR or radar. Most single-modality datasets [23], [24], [25], [26], [27], [28] provide 2D bounding box and scene segmentation labels for RGB images. Examples include Cityscapes [24], Mapillary Vistas [25], BDD100k [26] and Apolloscape [27]. To allow the studying of how vision algorithms generalise to different unseen data, [25], [26], [28] collect RGB images under different illumination and weather conditions. Other datasets only provide pedestrian detection annotation [29], [30], [31], [32], [33], [34], [35]. Recently, MIT and Toyota have released DriveSeg, which comes with pixel-level semantic labelling for 12 agent classes [36].

Multimodal datasets. KITTI [37] was the first-ever multimodal dataset. It provides depth labels from front-facing stereo images and dense point clouds from LiDAR alongside GPS/IMU (inertial) data. It also provides bounding-box annotations to facilitate improvements in 3D object detection. H3D [38] and KAIST [39] are two more examples of multimodal datasets. H3D provides 3D box annotations, using real-world LiDAR-generated 3D coordinates, in crowded scenes. Unlike KITTI, H3D comes with object detection annotations in a full 360^o view. KAIST provides thermal camera data alongside RGB, stereo, GPS/IMU and LiDAR-based range data. Among other notable multimodal datasets [18], [40] only consist of raw data without semantic labels, whereas [41] and [42] provide labels for location category and driving behaviour, respectively. The most recent multimodal large-scale AV datasets [43], [44], [45], [46], [47], [48] are significantly larger in terms of both data (also captured under varying weather conditions, e.g. by night or in the

1. https://github.com/ultralytics/yolov5.

rain) and annotations (RGB, LiDAR/radar, 3D boxes). For instance, Argovers [43] doubles the number of sensors in comparison to KITTI [37] and nuScenes [49], providing 3D bounding boxes with tracking information for 15 objects of interest. Similarly, Lyft [44] provides 3D bounding boxes for cars and location annotation including lane segments, pedestrian crosswalks, stop signs, parking zones, speed bumps, and speed humps. In a setup similar to KITTI's [37], in KITTI-360 [48] two fisheye cameras and a pushbroom laser scanner are added to have a full 360^o field of view. KITTI-360 contains semantic and instance annotations for both 3D point clouds and 2D images, which include 19 objects. IMU/GPS sensors are added for localisation purposes. Both 3D bounding boxes based on LiDAR data and 2D annotation on camera data for 4 objects classes are provided in Waymo [45]. In [46], using similar 3D annotation for 5 objects classes, the authors provide a more challenging dataset by adding more night-time scenarios using a faster-moving car. Amongst large-scale multimodal datasets, nuScenes [49], Lyft L5 [44], Waymo Open [45] and A*3D [46] are the most dominant ones in terms of number of instances, the use of high-quality sensors with different types of data (e.g., point clouds or 360° RGB videos), and richness of the annotation providing both semantic information and 3D bounding boxes. Furthermore, nuScenes [49], Argoverse [43] Lyft L5 [44] and KITTI-360 [48] provide contextual knowledge through human-annotated rich semantic maps, an important prior for scene understanding.

Trajectory prediction. Another line of work considers the problem of pedestrian trajectory prediction in the autonomous driving setting, and rests on several influential RGB-based datasets. To compile these datasets, RGB data were captured using either stationary surveillance cameras [50], [51], [52] or drone-mounted ones [53] for aerial view. [54], [55] use RGB images capturing an egocentric view from a moving car for future trajectory forecasting. Recently, the multimodal 3D point cloud-based datasets [37], [38], [43], [44], [45], [49], initially introduced for the benchmarking of 3D object detection and tracking, have been taken up for trajectory prediction as well. A host of interesting recent papers [56], [57], [58], [59] do propose datasets to study the intentions and actions of agents using cameras mounted on vehicles. However, they encompass a limited set of action labels (e.g. walking, standing, looking or crossing), wholly insufficient for a thorough study of road agent behaviour. Among them, TITAN [59] is arguably the most promising.
Our ROAD dataset is similar to TITAN in the sense that both consider actions performed by humans present in the road scene and provide spatiotemporal localisation for each person using multiple action labels. However, TITAN's action labels are restricted to humans (pedestrians), rather than extending to all road agents (with the exception of vehicles with 'stopped' and 'moving' actions). The dataset is a collection of much shorter videos which only last 10-20 seconds, and does not not contemplate agent location (a crucial source of information). Finally, the size of its vocabulary in terms of number of agents and actions is much smaller (see Table 1).

As mentioned, our ROAD dataset is built upon the multimodal Oxford RobotCar dataset, which contains both visual and 3D point cloud data. Here, however, we only process a number of its videos to describe and annotate road events. Note that it is indeed possible to map the 3D point clouds from RobotCar's LiDAR data onto the 2D images to enable true multi-modal action detection. However, a considerable amount would be required to do this, and will be considered in future extensions.

TABLE 1
Comparison of ROAD with similar datasets for perception in autonomous driving in terms of diversity of labels. The comparison is based on the number of classes portrayed and the availability of action annotations and tube tracks for both pedestrians and vehicles, as well as location information. Most competitor datasets do not provide action annotation for either pedestrians or vehicles.

Dataset	Class Num.	Location label	Action Ann		Tube Ann	
			Ped.	Veh.	Ped.	Veh.
SYNTHIA [60]	13	pixelwise ann.	-	-	-	-
SemKITTI [61]	28	3D sem. seg.	-	-	-	-
Cityscapes [24]	30	pixel level sem.	-	-	-	-
A2D2 [47]	14	3D sem. seg.	-	-	-	-
Waymo [45]	4	-	-	-	✓	✓
Apolloscape [27]	25	pixel level sem.	-	-	✓	✓
PIE [58]	6	-	✓	-	✓	-
TITAN [59]	50	-	✓	✓	✓	✓
KITTI360 [48]	19	sem. ann.	-	-	-	-
A*3D [46]	7	-	-	-	-	-
H3D [38]	8	-	-	-	✓	✓
Argoverse [43]	15	-	-	-	✓	✓
NuScense [49]	23	3D sem. seg.	-	-	✓	✓
DriveSeg [36]	12	sem. ann.	-	-	-	-
Spatiotemporal action detection datasets						
UCF24 [62]	24	-	✓	-	✓	-
AVA [63]	80	-	✓	-	✓	-
Multisports [64]	66	-	✓	-	✓	-
ROAD (ours)	43	✓	✓	✓	✓	✓

Ped. Pedestrian, Veh. Vehicle, ann. annotation, sem. seg. semantic segmentation

ROAD departs substantially from all previous efforts, as: (1) it is designed to formally introduce the notion of *road event* as a combination of three semantically-meaningful labels such as agent, action and location; (2) it provides both bounding-box-level *and* tube-level annotation (to validate methods that exploit the dynamics of motion patterns) on long-duration videos (thus laying the foundations for future work on event anticipation and continual learning); (3) it provides temporally dense annotation; (4) it labels the actions not only of physical humans but also of other relevant road agents such as vehicles of different kinds.

Table 1 compares our ROAD dataset with the other state-of-the-art datasets in perception for autonomous driving, in terms of the number and type of labels. As it can be noted in the table, the unique feature of ROAD is its diversity in terms of the types of actions and events portrayed, for all types of road agents in the scene. With 12 agent classes, 30 action classes and 15 location classes ROAD provides (through a combination of these three elements) a much more refined description of road scenes.

2.2 Action detection datasets

Providing annotation for action detection datasets is a painstaking process. Specifically, the requirement to track actors through the temporal domain makes the manual labelling of a dataset an extremely time consuming exercise, requiring frame-by-frame annotation. As a result, action detection benchmarks are fewer and smaller than, say, image classification, action recognition or object detection datasets.

Action recognition research can aim for robustness thanks to the availability of truly large scale datasets such

as Kinetics [65], Moments [66] and others, which are the de-facto benchmarks in this area. The recent 'something-something' video database focuses on more complex actions performed by humans using everyday objects [67], exploring a fine-grained list of 174 actions. More recently, temporal activity detection datasets like ActivityNet [68] and Charades [69] have come to the fore. Whereas the latter still do not address the spatiotemporal nature of the action detection problem, however, datasets such as J-HMDB-21 [70], UCF24 [71], LIRIS-HARL [72], DALY [73] or the more recent AVA [63] have been designed to provide spatial and temporal annotations for human action detection.

In fact, most action detection papers are validated on the rather dated and small LIRIS-HARL [72], J-HMDB-21 [70], and UCF24 [71], whose level of challenge in terms of presence of different source domains and nuisance factors is quite limited. Although recent additions such as DALY [73] and AVA [63] have somewhat improved the situation in terms of variability and number of instances labelled, the realistic validation of action detection methods is still an outstanding issue. AVA is currently the biggest action detection dataset with $1.6M$ label instances, but it is annotated rather sparsely (at a rate of one frame per second).

Overall, the main objective of these datasets is to validate the localisation of human actions in short, untrimmed videos. ROAD, in opposition, goes beyond the detection of actions performed by physical humans to extend the notion of other forms of intelligent agents (e.g., human- or AI-driven vehicles on the road). Furthermore, in contrast with the short clips considered in, e.g., J-HMDB-21 and UCF24, our new dataset is composed of 22 very long videos (around 8 minutes each), thus stressing the dynamical aspect of events and the relationship between distinct but correlated events. Crucially, it is geared towards online detection rather than traditional offline detection, as these videos are streamed in using a vehicle-mounted camera.

2.3 Online action detection

We believe advances in the field of human action recognition [22], [74], [75], [76] can be useful when devising a general approach to the situation awareness problem. We are particularly interested in the *action detection* problem [21], [63], [77], [78], in particular *online* action detection [19], given the incremental processing needs of an autonomous vehicle. Recent work in this area [19], [79], [80], [81], [82], [83] demonstrates very competitive performance compared to (generally more accurate) offline action detection methods [20], [63], [75], [84], [85], [86], [87], [88] on UCF-101-24 [71]. As mentioned, UCF-101-24 is the main benchmark for online action detection research, as it provides annotation in the form of action tubes and every single frame of the untrimmed videos in it is annotated (unlike AVA [63], in which videos are only annotated at one frame per second).

A short review of the state-of-the-art in online action detection is in place. Singh *et al.* [19]'s method was perhaps the first to propose an online, real-time solution to action detection in untrimmed videos, validated on UCF-101-24, and based on an innovative incremental tube construction method. Since then, many other papers [81], [82], [87] have made use of the online tube-construction method in [19]. A common trait of many recent online action detection methods is the reliance on 'tubelet' [81], [82], [84] predictions from a stack of frames. This, however, leads to processing delays proportional to the number of frames in the stack, making these methods not quite applicable in pure online settings. In the case of [81], [82], [84] the frame stack is usually 6-8 frames long, leading to a latency of more than half a second.

For these reasons, inspired by the frame-wise (2D) nature of [19] and the success of the latest single-stage object detectors (such as RetinaNet [89]), here we propose a simple extension of [19] termed '3D-RetinaNet' as a baseline algorithm for ROAD tasks. The latter is completely online when using a 2D backbone network. One, however, can also insert a 3D backbone to make it even more accurate, while keeping the prediction heads online. We benchmark our proposed 3D-RetinaNet architecture against the above-mentioned online and offline action detection methods on the UCF-101-24 dataset to show its effectiveness, twinned with its simplicity and efficiency. We also compare it on our new ROAD dataset against the state-of-the-art action detection Slowfast [22] network. We omit, however, to reproduce other state-of-the-art action detectors such as [90] and [91], for [90] is affected by instability at training time which makes it difficult to reproduce its results, whereas [91] is too complicated to be suitable as a baseline because of its sparse tracking and memory banks features. Nevertheless, both methods rely on the Slowfast detector as a backbone and baseline action detector.

3 THE DATASET

3.1 A multi-label benchmark

The ROAD dataset is specially designed from the perspective of self-driving cars, and thus includes actions performed not just by humans but by all road agents in specific locations, to form *road events* (REs). REs are annotated by drawing a bounding box around each active road agent present in the scene, and linking these bounding boxes over time to form 'tubes'. As explained, to this purpose three different types of labels are introduced, namely: (i) the category of *road agent* involved (e.g. *Pedestrian*, *Car*, *Bus*, *Cyclist*); (ii) the *type of action* being performed by the agent (e.g. *Moving away*, *Moving towards*, *Crossing* and so on), and (iii) the *location* of the road user relative the autonomous vehicle perceiving the scene (e.g. *In vehicle lane*, *On right pavement*, *In incoming lane*). In addition, ROAD labels the actions performed by the vehicle itself. Multiple agents might be present at any given time, and each of them may perform multiple actions simultaneously (e.g. a *Car* may be *Indicating right* while *Turning right*). Each agent is always associated with at least one action label.

The full lists of agent, action and location labels are given in the Supplementary material, Tables 1, 2, 3 and 4.

Agent labels. Within a road scene, the objects or people able to perform actions which can influence the decision made by the autonomous vehicle are termed *agents*. We only annotate *active* agents (i.e., a parked vehicle or a bike or a person visible to the AV but located away from the road are not considered to be 'active' agents). Three types of agent are considered to be of interest, in the sense defined above, to

the autonomous vehicle: people, vehicles and traffic lights. For simplicity, the AV itself is considered just like another agent: this is done by labelling the vehicle's bonnet. People are further subdivided into two sub-classes: pedestrians and cyclists. The vehicle category is subdivided into six sub–classes: car, small–size motorised vehicle, medium–size motorised vehicle, large–size motorised vehicle, bus, motorbike, emergency vehicle. Finally, the 'traffic lights' category is divided into two sub–classes: *Vehicle traffic light* (if they apply to the AV) and *Other traffic light* (if they apply to other road users). Only one agent label can be assigned to each active agent present in the scene at any given time.

Action labels. Each agent can perform one or more *actions* at any given time instant. For example, a traffic light can only carry out a single action: it can be either red, amber, green or 'black'. A car, instead, can be associated with two action labels simultaneously, e.g., *Turning right* and *Indicating right*. Although some road agents are inherently multitasking, some action combinations can be suitably described by a single label: for example, pushing an object (e.g. a pushchair or a trolley-bag) while walking can be simply labelled as *Pushing object*. The latter was our choice.

AV own actions. Each video frame is also labelled with the action label associated with what the AV is doing. To this end, a bounding box is drawn on the bonnet of the AV. The AV can be assigned one of the following seven action labels: *AV-move*, *AV-stop*, *AV-turn-left*, *AV-turn-right*, *AV-overtake*, *AV-move-left* and *AV-move-right*. The full list of AV own action classes is given in the Supplementary material, Table 4. Note that these are separate classes only applicable to the AV, with a different semantics than the similar-sounding classes. For instance, the regular *Moving* action label means 'moving in the perpendicular direction to the AV', whereas *AV-move* means that the AV is on the move along its normal direction of travel. These labels mirror those used for the autonomous vehicle in the Honda Research Institute Driving Dataset (HDD) [92].

Location labels. Agent *location* is crucial for deciding what action the AV should take next. As the final, long-term objective of this project is to assist autonomous decision making, we propose to label the location of each agent from the perspective of the autonomous vehicle. For example, a pedestrian can be found on the right or the left pavement, in the vehicle's own lane, while crossing or at a bus stop. The same applies to other agents and vehicles as well. There is no location label for the traffic lights as they are not movable objects, but agents of a static nature and well-defined location. To understand this concept, Fig. 1 illustrates two scenarios in which the location of the other vehicles sharing the road is depicted from the point of view of the AV. *Traffic light* is the only agent type missing location labels, all the other agent classes are associated with at least one location label. A complete table with location classes and their description is provided in Supplementary material.

3.2 Data collection

ROAD is composed of 22 videos from the publicly available Oxford RobotCar Dataset [18] (OxRD) released in 2017 by the Oxford Robotics Institute[2], covering diverse road scenes under various weather conditions. The OxRD dataset, collected from the narrow streets of the historic city of Oxford, was selected because it presents challenging scenarios for an autonomous vehicle due to the diversity and density of various road users and road events. The OxRD dataset was gathered using 6 cameras, as well as LIDAR (Light Detection and Ranging), GPS (Global Positioning System) and INS (Inertial Navigation System) sensors mounted on a Nissan LEAF vehicle [18]. To construct ROAD we only annotated videos from the frontal camera view.

Note, however, that our labelling process (described below) is not limited to OxRD. In principle, other autonomous vehicle datasets (e.g. [26], [93]) may be labelled in the same manner to further enrich the ROAD benchmark,: we plan to do exactly so in the near future.

Video selection. Within OxRD, videos were selected with the objective of ensuring diversity in terms of weather conditions, times of the day and types of scenes recorded. Specifically, the 22 videos have been recorded both during the day (in strong sunshine, rain or overcast conditions, sometimes with snow present on the surface) and at night. Only a subset of the large number of videos available in OxRD was selected. The presence of semantically meaningful content was the main selection criterion. This was done by manually inspecting the videos in order to cover all types of labels and label classes and to avoid 'deserted' scenarios as much as possible. Each of the 22 videos is 8 minutes and 20 seconds long, barring three videos whose duration is 6:34, 4:10 and 1:37, respectively. In total, ROAD comprises 170 minutes of video content.

Preprocessing. Some preprocessing was conducted. First, the original sets of video frames were downloaded and demosaiced, in order to convert them to red, green, and blue (RGB) image sequences. Then, they were encoded into proper video sequences using `ffmpeg`[3] at the rate of 12 frames per second (fps). Although the original frame rate in the considered frame sequences varies from 11 fps to 16 fps, we uniformised it to keep the annotation process consistent. As we retained the original time stamps, however, the videos in ROAD can still be synchronised with the LiDAR and GPS data associated with them in the OxRD dataset, allowing future work on multi-modal approaches.

3.3 Annotation process

Annotation tool. Annotating tens of thousands of frames rich in content is a very intensive process; therefore, a tool is required which can make this process both fast and intuitive. For this work, we adopted Microsoft's VoTT[4]. The most useful feature of this annotation tool is that it can copy annotations (bounding boxes and their labels) from one frame to the next, while maintaining a unique identification for each box, so that boxes across frames are automatically linked together. Moreover, VoTT also allows for multiple labels, thus lending itself well to ROAD's multi-label annotation concept. A number of examples of annotated frames from two videos using VOTT tool is provided in supplementary material.

2. http://robotcar-dataset.robots.ox.ac.uk/

3. https://www.ffmpeg.org/

4. https://github.com/Microsoft/VoTT/

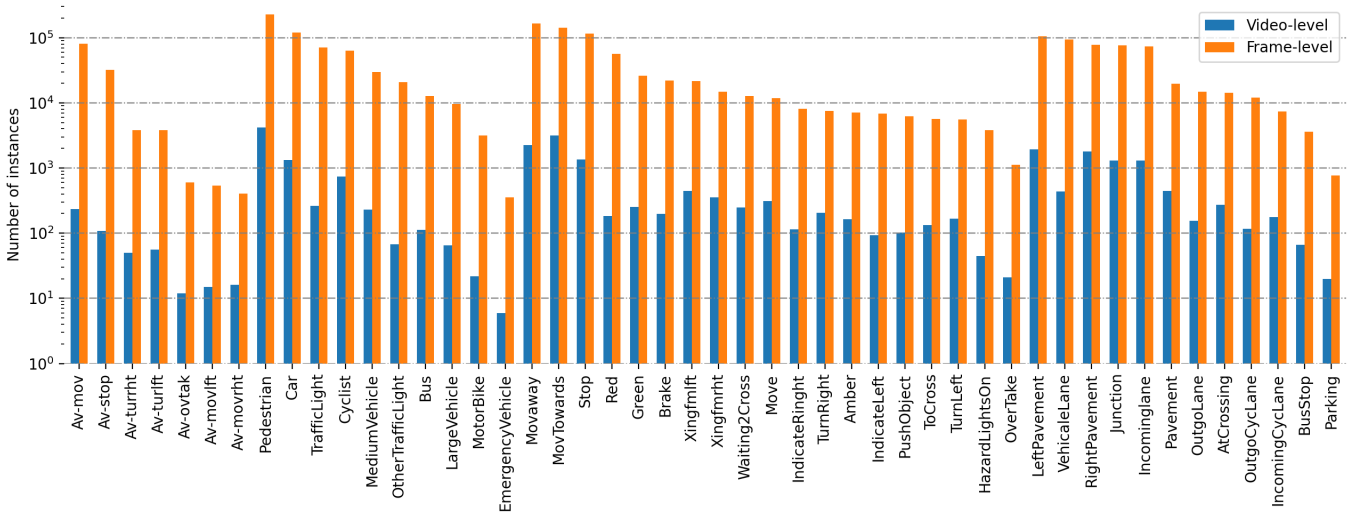


Fig. 2. Number of instances of each class of individual label-types, in logarithmic scale.

Annotation protocol. All salient objects and actors within the frame were labelled, with the exception of inactive participants (mostly parked cars) and objects / actors at large distances from the ego vehicle, as the latter were judged to be irrelevant to the AV's decision making. This can be seen in the attached 30-minute video[5] portraying ground truth and predictions. As a result, pedestrians, cyclists and traffic lights were always labelled. Vehicles, on the other hand, were only labelled when active (i.e., moving, indicating, being stopped at lights or stopping with hazard lights on on the side of road). As mentioned, only parked vehicles were not considered active (as they do not arguably influence the AV's decision making), and were thus not labelled.

Event label generation. Using the annotations manually generated for actions and agents in the multi-label scenario as discussed above it is possible to generate *event-level* labels about agents, e.g. *Pedestrian / Moving towards* the AV *On right pavement* or *Cyclist / Overtaking / In vehicle lane*. Any combinations of location, action and agent labels are admissible. If location labels are ignored, the resulting event labels become location-invariant.

In addition to event tubes, in this work we do explore *agent-action* pair instances (see Sec. 5). Namely, given an agent tube and the continuous temporal sequence of action labels attached to its constituent bounding box detections, we can generate action tubes by looking for changes in the action label series associated with each agent tube. For instance, a *Car* appearing in a video might be first *Moving away* before *Turning left*. The agent tube for the car will then be formed by two contiguous agent-action tubes: a first tube with label pair *Car / Moving away* and a second one with pair *Car / Turning left*.

3.4 Tasks

ROAD is designed as a sandbox for validating the six tasks relevant to situation awareness in autonomous driving outlined in Sec. 1.1. Five of these tasks are detection tasks, while the last one is a frame-level action recognition task sometimes referred to as 'temporal action segmentation' [69], Table 2 shows the main attributes of these tasks.

5. https://www.youtube.com/watch?v=CmxPjHhiarA.

TABLE 2
ROAD tasks and attributes.

Task type	Problem type	Output	Multiple labels
Active agent	Detection	Box&Tube	No
Action	Detection	Box&Tube	Yes
Location	Detection	Box&Tube	Yes
Duplex	Detection	Box&Tube	Yes
Event	Detection	Box&Tube	Yes
AV-action	Temp segmentation	Start/End	No

All detection tasks are evaluated both at frame-level and at video- (tube-)level. *Frame-level detection* refers to the problem of identifying in each video frame the bounding box(es) of the instances there present, together with the relevant class labels. *Video-level detection* consists in regressing a whole series of temporally-linked bounding boxes (i.e., in current terminology, a 'tube') together with the relevant class label. In our case, the bounding boxes will mark a specific active agent in the road scene. The labels may issue (depending on the specific task) either from one of the individual label types described above (i.e., agent, action or location) or from one of the meaningful combinations described in 3.3 (i.e., either agent-action pairs or events).

Below we list all the tasks for which we currently provide a baseline, with a short description.

1) *Active agent detection* (or *agent detection*) aims at localising an active agent using a bounding box (frame-level) or a tube (video-level) and assigning a class label to it.
2) *Action detection* seeks to localise an active agent occupied in performing a specific action from the list of action classes.
3) In *agent location detection* (or *location detection*) a label from the relevant list of locations (as seen from the AV) is sought and attached to the relevant bounding box or tube.

4) In *agent-action detection* the bounding box or tube is assigned a pair agent-action as explained in 3.3. We sometimes refer to this task as 'duplex detection'.
5) *Road event detection* (or *event detection*) consist in assigning to each box or tube a triplet of class labels.
6) *Autonomous vehicle temporal action segmentation* is a frame-level action classification task in which each video frame is assigned a label from the list of possible AV own actions. We refer to this task as 'AV-action segmentation', similarly to [69].

3.5 Quantitative summary

Overall, $122K$ frames extracted from 22 videos were labelled, in terms of both AV own actions (attached to the entire frame) and bounding boxes with attached one or more labels of each of the three types: agent, action, location. In total, ROAD includes $560K$ bounding boxes with $1.7M$ instances of individual labels. The latter figure can be broken down into $560K$ instances of agent labels, $640K$ instances of action labels, and $499K$ instances of location labels. Based on the manually assigned individual labels, we could identify $603K$ instances of duplex (agent-action) labels and $454K$ instances of triplets (event labels).

The number of instances for each individual class from the three lists is shown in Fig. 2 (frame-level, in orange). The $560K$ bounding boxes make up $7,029$, $9,815$, $8,040$, $9,335$ and $8,394$ tubes for the label types agent, action, location, agent-action and event, respectively. Figure 2 also shows the number of tube instances for each class of individual label types as number of video-level instances (in blue).

4 BASELINE AND CHALLENGE

Inspired by the success of recent 3D CNN architectures [74] for video recognition and of feature-pyramid networks (FPN) [94] with focal loss [89], we propose a simple yet effective 3D feature pyramid network (3D-FPN) with focal loss as a baseline method for ROAD's detection tasks. We call this architecture *3D-RetinaNet*.

4.1 3D-RetinaNet architecture

The data flow of 3D-RetinaNet is shown in Figure 3. The input is a sequence of T video frames. As in classical FPNs [94], the initial block of 3D-RetinaNet consists of a backbone network outputting a series of forward feature pyramid maps, and of lateral layers producing the final feature pyramid composed by T feature maps. The second block is composed by two sub-networks which process these features maps to produce both bounding boxes (4 coordinates) and C classification scores for each anchor location (over A possible locations). In the case of ROAD, the integer C is the sum of the numbers of agent, action, location, action-agent (duplex) and agent-action-location (event) classes, plus one reserved for an *agentness* score. The extra class agentness is used to describe the presence or absence of an active agent. As in FPN [94], we adopt ResNet50 [95] as the backbone network.

2D versus 3D backbones. In our experiments we show results obtained using three different backbones: frame-based ResNet50 (2D), inflated 3D (I3D) [74] and Slowfast [22], in the manner also explained in [22], [75]. Choosing a 2D backbone makes the detector completely online [19], with a delay of a single frame. Choosing an I3D or a Slowfast backbone, instead, causes a 4-frame delay at detection time. Note that, as Slowfast and I3D networks makes use of a max-pool layer with stride 2, the initial feature pyramid in the second case contains $T/2$ feature maps. Nevertheless, in this case we can simply linearly upscale the output to T feature maps.

AV action prediction heads. In order for the method to also address the prediction of the AV's own actions (e.g. whether the AV is stopping, moving, turning left etc.), we branch out the last feature map of the pyramid (see Fig. 3, bottom) and apply spatial average pooling, followed by a temporal convolution layer. The output is a score for each of the C_a classes of AV actions, for each of the T input frames.

Loss function. As for the choice of the loss function, we adopt a binary cross-entropy-based focal loss [89]. We choose a binary cross entropy because our dataset is multi-label in nature. The choice of a focal-type loss is motivated by the expectation that it may help the network deal with long tail and class imbalance (see Figure 2).

4.2 Online tube generation via agentness score

The autonomous driving scenario requires any suitable method for agent, action or event tube generation to work in an *online* fashion, by incrementally updating the existing tubes as soon as a new video frame is captured. For this reason, this work adopts a recent algorithm proposed by Singh *et al.* [19], which incrementally builds action tubes in an online fashion and at real-time speed. To be best of our knowledge, [19] was the first online multiple action detection approach to appear in the literature, and was later adopted by almost all subsequent works [81], [82], [87] on action tube detection.

Linking of detections. We now briefly review the tube-linking method of Singh *et al.* [19], and show how it can be adapted to build agent tubes based on an 'agentness' score, rather than build a tube separately for each class as proposed in the original paper. This makes the whole detection process faster, since the total number of classes is much larger than in the original work [19]. The proposed 3D-RetinaNet is used to regress and classify detection boxes in each video frame potentially containing an active agent of interest. Subsequently, detections whose score is lower than 0.025 are removed and non-maximal suppression is applied based on the agentness score.

At video start, each detection initialises an agentness tube. From that moment on, at any time instance t the highest scoring tubes in terms of mean agentness score up to $t-1$ are linked to the detections with the highest agentness score in frame t which display an Intersection-over-Union (IoU) overlap with the latest detection in the tube above a minimum threshold λ. The chosen detection is then removed from the pool of frame-t detections. This continue until the tubes are either assigned or not assigned a detection from current frame. Remaining detections at time t are used to initiate new tubes. A tube is terminated after no suitable detection is found for n consecutive frames. As the linking process takes place, each tube carries scores

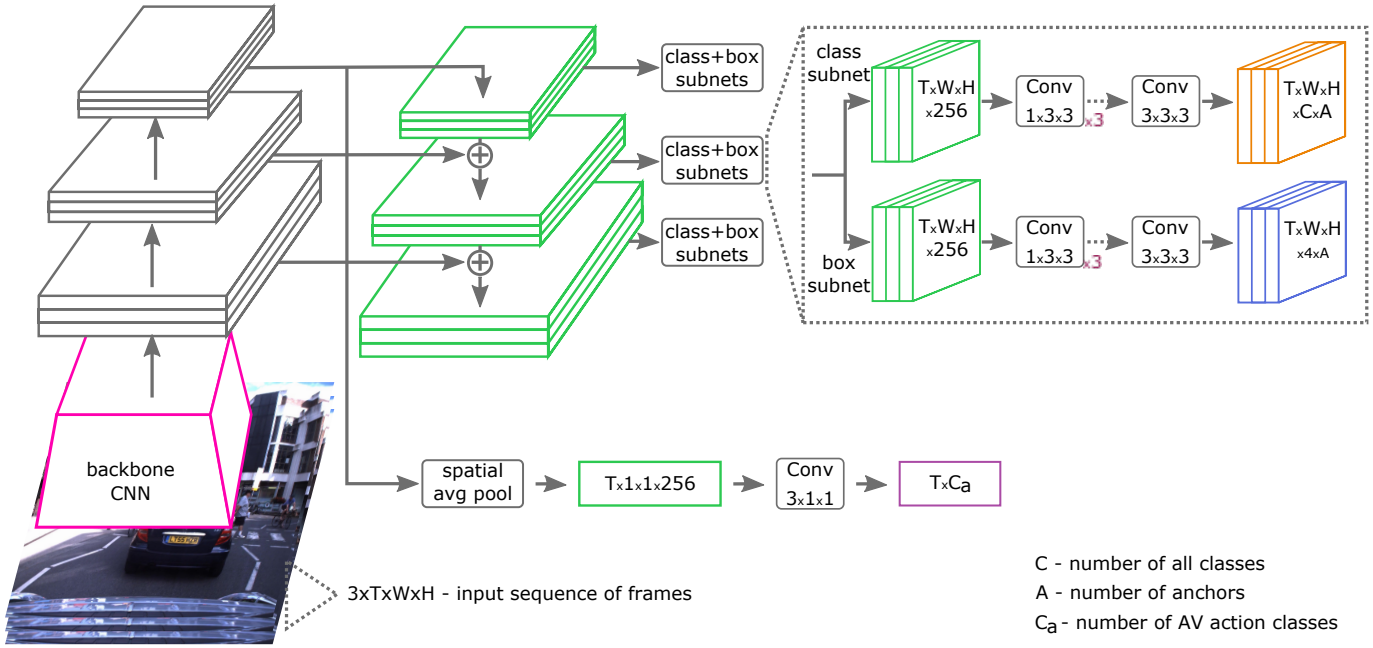


Fig. 3. Proposed 3D-RetinaNet architecture for online video processing.

for all the classes of interest for the task at hand (e.g., action detection rather than event detection), as produced by the classification subnet of 3D-RetinaNet. We can then label each agentness tube using the k classes that show the highest mean score over the duration of the tube.

Temporal trimming. Most tubelet based methods [81], [82], [96] do not perform any temporal trimming of the action tubes generated in such a way (i.e., they avoid deciding when they should start or end). Singh *et al.* [19] proposed to pose the problem in a label consistency formulation solved via dynamic programming. However, as it turns out, temporal trimming [19] does not actually improve performance, as shown in [87], except in some settings, for instance in the DALY [73] dataset.

The situation is similar for our ROAD dataset as opposed to what happens on UCF-101-24, for which temporal trimming based on solving the label consistency formulation in terms of the actionness score, rather than the class score, does help improve localisation performance. Therefore, in our experiments we only use temporal trimming on the UCF-101-24 dataset but not on ROAD.

4.3 The ROAD challenge

To introduce the concept of road event, our new approach to situation awareness and the ROAD dataset to the computer vision and AV communities, some of us have organised in October 2021 the workshop "The ROAD challenge: Event Detection for Situation Awareness in Autonomous Driving"[6]. For the challenge, we selected (among the tasks described in Sec. 3.4) only three tasks: agent detection, action detection and event detection, which we identified as the most relevant to autonomous driving.

As standard in action detection, evaluation was done in terms of video mean average precision (video-mAP). 3D-RetinaNet was proposed as the baseline for all three tasks. Challenge participants had 18 videos available for training and validation. The remaining 4 videos were to be used to test the final performance of their model. This split was applied to all the three challenges (split 3 of the ROAD evaluation protocol, see Section 5.3).

The challenge opened for registration on April 1 2021, with the training and validation folds released on April 30, the test fold released on July 20 and the deadline for submission of results set to September 25. For each stage and each Task the maximum number of submissions was capped at 50, with an additional constraint of 5 submissions per day. The workshop, co-located with ICCV 2021, took place on October 16 2021.

In the validation phase we had between three and five teams submit between 15 and 17 entries to each of three challenges. In the test phase, which took place after the summer, we noticed a much higher participation with 138 submissions from 9 teams to the agent challenge, 98 submissions from 8 teams to the action challenge, and 93 submission from 6 teams to the event detection challenge.

The methods proposed by the winners of each challenge are briefly recalled in Section 5.4.

Benchmark maintenance. After the conclusion of the ROAD @ ICCV 2021 workshop, the challenge has been reactivated to allow for submissions indefinitely. The ROAD benchmark will be maintained by withholding the test set from the public on the eval.ai platform[7], where teams can submit their predictions for evaluation. Training and validation sets can be downloaded from https://github.com/gurkirt/road-dataset.

6. https://sites.google.com/view/roadchallangeiccv2021/.

7. https://eval.ai/web/challenges/challenge-page/1059/overview

5 EXPERIMENTS

In this section we present results on the various task the ROAD dataset is designed to benchmark (see Sec. 3.4), as well as the action detection results delivered by our 3D-RetinaNet model on UCF-101-24 [62], [97].

We first present the evaluation metrics and implementation details specific to ROAD in Section 5.1. In Section 5.2 we benchmark our 3D-RetinaNet model for the action detection problem on UCF-101-24. The purpose is to show that this baseline model is competitive with the current state of the art in action tube detection while only using RGB frames as input, and to provide a sense of how challenging ROAD is when compared to standard action detection benchmarks. Indeed, the complex nature of the real-world, non-choreographed road events, often involving large numbers of actors simultaneously responding to a range of scenarios in a variety of weather conditions makes ROAD a dataset which poses significant challenges when compared to other, simpler action recognition benchmarks.

In Section 5.3 we illustrate and discuss the baseline results on ROAD for the different tasks (Sec. 5.3.2), using a 2D ResNet50, an I3D and a Slowfast backbone, as well as the agent detection performance of the standard YOLOv5 model. Different training/testing splits encoding different weather conditions are examined using the I3D backbone (Sec. 5.3.3). In particular, in Sec. 5.3.4 we show the results one can obtain when predicting composite labels as products of single-label predictions as opposed to training a specific model for them, as this can provide a crucial advantage in terms of efficiency, as well as give the system the flexibility to be extended to new composite labels without retraining. Finally, in Sec. 5.3.5 we report our baseline results on the temporal segmentation of AV actions.

5.1 Implementation details

The results are evaluated in terms of both frame-level bounding box detection and of tube detection. In the first case, the evaluation measure of choice is *frame mean average precision* (f-mAP). We set the Intersection over Union (IoU) detection threshold to 0.5 (signifying a 50% overlap between predicted and true bounding box). For the second set of results we use *video mean average precision* (video-mAP), as information on how the ground-truth BBs are temporally connected is available. These evaluation metrics are standard in action detection [19], [81], [98], [99], [100].
We also evaluate actions performed by AV, as described in 3.1. Since this is a temporal segmentation problem, we adopt the mean average precision metric computed at frame-level, as standard on the Charades [69] dataset.

We use sequences of $T = 8$ frames as input to 3D-RetinaNet. Input image size is set to 512×682. This choice of T is the result of GPU memory constraints; however, at test time, we unroll our convolutional 3D-RetinaNet for sequences of 32 frames, showing that it can be deployed in a streaming fashion. We initialise the backbone network with weights pretrained on Kinetics [65]. For training we use an SGD optimiser with step learning rate. The initial learning rate is set to 0.01 and drops by a factor of 10 after 18 and 25 epochs, up to an overall 30 epochs. For tests on the UCF-101-24 dataset the learning rate schedule is shortened to a maximum 10 epochs, and the learning rate drop steps are set to 6 and 8.

The parameters of the tube-building algorithm (Sec. 4.2) are set by cross validation. For ROAD we obtain $\lambda = 0.5$ and $k = 4$. For UCF-101-24, we get $\lambda = 0.25$ and $k = 4$. Temporal trimming is only performed on UCF-101-24.

TABLE 3
Comparison of the action detection performance (frame-mAP@0.5 (f-mAP) and video-mAP at different IoU thresholds) of the proposed 3D-RetinaNet baseline model with the state-of-the-art on the UCF-101-24 dataset.

Methods / δ =	f-mAP	0.2	0.5	0.75	0.5:0.9
RGB + FLOW methods					
MR-TS Peng *et al.* [85]	–	73.7	32.1	00.9	07.3
FasterRCNN Saha *et al.* [98]	–	66.6	36.4	07.9	14.4
SSD + OJLA Behl *et al.* [80]*	–	68.3	40.5	14.3	18.6
SSD Singh *et al.* [19]*	–	76.4	45.2	14.4	20.1
AMTnet Saha *et al.* [84]*	–	78.5	49.7	22.2	24.0
ACT Kalogeiton *et al.* [81]*	–	76.5	49.2	19.7	23.4
TraMNet Singh *et al.* [87]*	–	79.0	50.9	20.1	23.9
Song *et al.* [101]	72.1	77.5	52.9	21.8	24.1
Zhao *et al.* [86]	–	78.5	50.3	22.2	24.5
I3D Gu *et al.* [102]	76.3	–	**59.9**	–	–
Li *et al.* [82]*	**78.0**	**82.8**	53.8	**29.6**	**28.3**
RGB only methods					
RGB-SSD Singh *et al.* [19]*	65.0	72.1	40.6	14.1	18.5
RGB-AMTNet Saha *et al.* [84]*	–	75.8	45.3	19.9	22.0
3D-RetinaNet / 2D (ours)*	65.2	73.5	48.6	22.0	22.8
3D-RetinaNet / I3D (ours)	**75.2**	**82.4**	**58.2**	**25.5**	**27.1**

* online methods

5.2 Baseline performance on UCF-101-24

Firstly, we benchmarked 3D-RetinaNet on UCF-101-24 [62], [97], using the corrected annotations from [19]. We evaluated both frame-mAP and video-mAP and provided a comparison with state-of-the-art approaches in Table 3. It can be seen that our baseline is competitive with the current state-of-the-art [82], [102], even as those methods use both RGB and optical flow as input, as opposed to ours. As shown in the bottom part of Table 3, 3D-RetinaNet outperforms all the methods solely relying on appearance (RGB) by large margins. The model retains the simplicity of single-stage methods, while sporting, as we have seen, the flexibility of being able to be reconfigured by changing the backbone architecture. Note that its performance could be further boosted using the simple optimisation technique proposed in [103].

5.3 Experimental results on ROAD

5.3.1 Three splits: modelling weather variability

For the benchmarking of the ROAD tasks, we divided the dataset into two sets. The first set contains 18 videos for training and validation purposes, while the second set contains 4 videos for testing, equally representing the four types of weather conditions encountered.

The group of training and validation videos is further subdivided into three different ways ('splits'). In each split, 15 videos are selected for training and 3 for validation. Details on the number of videos for each set and split are shown in Table 4. All 3 validation videos for Split-1 are

TABLE 4
Splits of training, validation and test sets for the ROAD dataset with respect to weather conditions. The table shows the number of videos in each set or split. For splits, the first figure is the number of training videos, the second number that of validation videos.

Condition	sunny	overcast	snow	night
Training and validation	7	7	1	3
Split-1	7/0	4/3	1/0	3/0
Split-2	7/0	7/0	1/0	0/3
Split-3	4/3	7/0	1/0	3/0
Testing	1	1	1	1

overcast; 4 overcast videos are also present in the training set. As such, Split-1 is designed to assess the effect of different overcast conditions. Split-2 has all 3 night videos in the validation subset, and none in the training set. It is thus designed to test model robustness to day/night variations. Finally, Split-3 contains 4 training and 3 validation videos for sunny weather: it is thus designed to evaluate the effect of different sunny conditions, as camera glare can be an issue when the vehicle is turning or facing the sun directly.

Note that there is no split to simulate a bias towards snowy conditions, as the dataset only contains one video of that kind. The test set (bottom row) is more uniform, as it contains one video from each environmental condition.

5.3.2 Results on the various tasks

Results are reported for the tasks discussed in Section 3.4.

Frame-level results across the five detection tasks are summarised in Table 5 using the frame-mAP (f-mAp) metric, for a detection threshold of $\delta = 0.5$. The reported figures are averaged across the three splits described above, in order to assess the overall robustness of the detectors to domain variations. Performance within each split is evaluated on both the corresponding validation subset and test set. Each row in the Table shows the result of a particular combination of backbone network (2D, I3D, or Slowfast) and test-time sequence length (in number of frames, 8 and 32). Frame-level results vary between 16.8% (events) and 65.4% (agentness) for I3D, and between 23.9% and 69.2% for Slowfast. Clearly, for each detection task except agentnness (which amounts to agent detection on ROAD) the performance is quite lower than the 75.2% achieved by our I3D baseline network on UCF-101-24 (Table 3, last row). This is again due to the numerous nuisance factors present in ROAD, such as significant camera motion, weather conditions, etc. For a fair comparison, note that there are only 11 agent classes, as opposed to e.g. 23 action classes and 15 location classes.

Video-level results are reported in terms of video-mAP in Table 6. As for the frame-level results, tube detection performance (see Sec. 4.2) is averaged across the three splits. One can appreciate the similarities between frame- and video-level results, which follow a similar trend albeit at a much lower absolute level. Again, results are reported for different backbone networks and sequence lengths. Not considering the YOLOv5 numbers, video-level results at detection threshold $\delta = 0.2$ vary between a minimum of 20.5% (actions) to a maximum of 33.0% (locations), compared to the 82.4% achieved on UCF-101-24. For a detection threshold δ equal to 0.5, the video-level results lie between 4.7% (actions) and 11% (locations) compared to the 58.2% achieved on UCF-101-24 for the same IoU threshold. The difference is quite dramatic, and highlights the order of magnitude of the challenge involved by perception in autonomous driving compared to a standard benchmark portraying only human actions. Furthermore, we can notice a few important facts.

TABLE 5
Frame-level results (mAP %) averaged across the three splits of ROAD. The considered models differ in terms of backbone network (2D, I3D, and Slowfast) and clip length (08 vs 32). The performance of YOLOv5 on agent detection is also reported. Detection threshold $\delta = 0.5$. Both validation and test performance are reported for each entry.

Model	Agentness	Agents	Actions	Locations	Duplexes	Events
2D-08	51.8/63.4	30.9/39.5	15.9/22.0	23.2/30.8	18.1/25.1	10.6/12.8
2D-32	52.4/64.2	31.5/39.8	16.3/22.6	23.6/31.4	18.7/25.8	10.8/13.0
I3D-08	52.3/65.1	32.2/39.5	19.3/25.4	24.5/34.9	21.5/30.8	12.3/16.5
I3D-32	52.7/65.4	32.3/39.2	19.7/25.9	24.7/35.3	21.9/31.0	12.6/16.8
Slowfast-08	68.8/**69.2**	41.9/47.5	26.9/31.1	34.6/**37.3**	31.6/36.0	**18.1**/23.7
Slowfast-32	**69.3**/68.7	42.6/43.7	**27.3**/**31.7**	**34.8**/36.4	**32.0**/**36.1**	18.0/**23.9**
YOLOv5	-	**57.9**/**56.9**	-	-	-	-

TABLE 6
Video-level results (mAP %) averaged across the three ROAD splits. The models differ in terms of backbone network (2D, I3D and Slowfast) and test time clip length (08 vs 32). The performance of YOLOv5 on agent detection is also reported. Both validation and test performance are reported for each entry.

Model	Agents	Actions	Locations	Duplexes	Events
Detection threshold $\delta = 0.2$					
2D-08	22.2/25.1	10.3/13.9	18.2/24.8	16.1/21.9	12.8/14.7
2D-32	22.6/25.0	11.2/14.5	18.5/25.9	16.2/22.7	13.0/15.3
I3D-08	23.2/26.5	14.1/15.8	20.8/25.8	21.1/24.0	14.9/17.4
I3D-32	24.4/26.9	14.3/17.5	21.3/27.1	21.4/25.5	15.9/17.9
Slowfast-08	24.1/29.0	**16.0**/**20.5**	28.3/**33.0**	24.0/**27.3**	18.9/22.4
Slowfast-32	24.2/28.6	**16.0**/19.55	**29.0**/29.7	**24.3**/26.1	**19.1**/**22.5**
YOLOv5	**38.8**/**43.3**	-	-	-	-
Detection threshold $\delta = 0.5$					
2D-08	8.9/7.5	2.3/3.0	5.2/6.1	6.5/6.1	5.1/5.3
2D-32	8.3/8.0	2.7/3.3	5.6/7.1	6.3/6.8	5.0/5.7
I3D-08	9.2/9.6	**4.0**/4.3	5.8/6.9	7.2/7.4	4.6/5.4
I3D-32	9.7/10.2	**4.0**/4.6	6.4/7.7	7.1/8.3	4.8/6.1
Slowfast-08	7.1/8.9	3.9/**4.7**	7.1/**11.0**	**7.3**/7.7	**6.5**/6.6
Slowfast-32	8.3/9.8	3.7/4.4	**8.4**/10.0	7.1/**9.0**	5.3/**7.3**
YOLOv5	**18.7**/**13.9**	-	-	-	-

Streaming deployment. Increasing test sequence length from 8 to 32 does not much impact performance. This indicates that, even though the network is trained on 8-frame clips, being fully convolutional (including the heads in the temporal direction), it can be easily unrolled to process longer sequences at test time, making it easy to deploy in a streaming fashion. Being deployable in an incremental fashion is a must for autonomous driving applications; this is a quality that other tubelet-based online action detection methods [81], [82], [87] fail to exhibit, as they can only be deployed in a sliding window fashion. Interestingly, the latest work on streaming object detection [104] proposes an approach that integrates latency and accuracy into a single metric for real-time online perception, termed 'streaming accuracy'. We will consider adopting this metric in the future evolution of ROAD.

Impact of the backbone. Broadly speaking, the Slowfast [22] and I3D [74] versions of the backbone perform as expected, much better than the 2D version. A Slowfast backbone can particularly help with tasks which require the system to 'understand' movement, e.g. when detecting actions, agent-

TABLE 7
Number of video- and frame-level instances for each label (individual or composite), left. Corresponding frame-/video-level results (mAP@%) for each of the three ROAD splits (right). Val-n denotes the validation set for Split n. Results produced by an I3D backbone.

	Number of instance					Frame-mAP@0.5/Video-mAP@0.2					
Train subset	#Boxes/#Tubes					Train-1		Train-2		Train-3	
Eval subset	All	Val-1	Val-2	Val-3	Test	Val-1	Test	Val-2	Test	Val-3	Test
Agent	559142/7029	60103/781	79119/761	83750/809	82465/1138	44.5/30.1	34.0/25.7	17.2/16.0	40.9/27.4	35.3/27.1	42.6/27.5
Action	639740/9815	69523/1054	89142/1065	95760/1111	94669/1548	26.2/17.0	26.6/17.4	11.7/11.4	25.3/17.3	21.2/14.6	25.7/17.9
Location	498566/8040	56594/851	67116/864	77084/914	70473/1295	34.9/28.6	35.2/26.4	13.7/12.1	33.9/26.3	25.4/23.2	36.7/28.6
Duplex	603274/9335	60000/965	85730/1032	88960/1050	89080/1471	28.2/25.3	28.7/23.4	13.6/17.3	31.4/24.8	23.9/21.6	33.0/28.4
Event	453626/8394	43569/883	65965/963	72152/967	64545/1301	17.7/18.6	15.9/15.8	6.4/11.8	16.4/18.9	13.7/17.2	18.1/18.9
	Number of instances					Frame-AP					
AV-action	122154/490	17929/67	18001/56	16700/85	20374/82	57.9	45.7	33.5	43.6	43.7	48.2

actions pairs and road events, at least at 0.2 IoU. Under more stringent localisation requirements ($\delta = 0.5$), it is interesting to notice how Slowfast's advantage is quite limited, with the I3D version often outperforming it. This shows that by simply switching backbone one can improve on performance or other desirable properties, such as training speed (as in or X3D [76]). The 3D CNN encoding can be made intrinsically online, as in RCN [105]. Finally, even stronger backbones using transformers [106], [107] can be plugged in.

Level of task challenge. The overall results on event detection (last column in both Table 5 and Table 6) are encouraging, but they remain in the low 20s at best, showing how challenging situation awareness is in road scenarios.

Comparison across tasks. From a superficial comparison of the mAPs obtained, action detection seems to perform worse than agent-action detection or even event detection. However, the headline figures are not really comparable since, as we know, the number of class per task varies. More importantly, within-class variability is often lower for composite labels. For example, the score for *Indicating right* is really low, whereas *Car / Indicating-right* has much better performance (see Supplementary material, Tables 11–13 for class-specific performance). This is because the within-class variability of the pair *Car / Indicating-right* is much lower than that of *Indicating right*, which puts together instances of differently-looking types of vehicles (e.g. buses, cars and vans) all indicating right. Interestingly, results on agents are comparable among the four baseline models (especially for f-mAP and v-mAP at 0.2, see Tables 5 and 6).

YOLOv5 for Agent detection. For completeness, we also trained YOLOv5[8] for the detection of active agents. The results are shown in the last row of both Table 5 and Table 6. Keeping is mind that YOLOv5 is trained only on single input frames, it shows a remarkable improvement over the other baseline methods for active agent detection. We believe that is because YOLOv5 is better at the regression part of the detection problem – namely, Slowfast has a recall of 71% compared to the 94% of YOLOv5, so that Slowfast has a 10% lower mAP for active agent detection. We leave the combination of YOLOv5 for bounding box proposal generation and Slowfast for proposal classification as a promising future extension, which could lead to a general improvement across all tasks.

8. https://github.com/ultralytics/yolov5

Validation vs test results. Results on the test set are, on average, superior to those on the validation set. This is because the test set includes data from all weather/visibility conditions (see Table 4), whereas for each split the validation set only contains videos from a single weather condition. E.g., in Split 2 all validation videos are nighttime ones.

5.3.3 Results under different weather conditions

Table 7 shows, instead, the results obtained under the three different splits we created on the basis of the weather/environmental conditions of the ROAD videos, discussed in Section 5.3.1 and summarised in Table 4. Note that the total number of instances (boxes for frame-level results or tubes for video-level ones) of the five detection tasks is comparable for all the three splits.

We can see how Split-2 (for which all three validation videos are taken at night and no nighttime videos are used for training, see Table 4) has the lowest validation results, as seen in Table 7 (Train-2, Val-2). When the network trained on Split-2's training data is evaluated on the (common) test set, instead, its performance is similar to that of the networks trained on the other splits (see Test columns). Split-1 has three overcast videos in the validation set, but also four overcast videos in the training set. The resulting network has the best performance across the three validation splits. Also, under overcast conditions one does not have the typical problems with night-time vision, nor glares issues as in sunny days. Split-3 is in a similar situation to Split-1, as it has sunny videos in both train and validation sets.

These results seem to attest a certain robustness of the baseline to weather variations, for no matter the choice of the validation set used to train the network parameters (represented by the three splits), the performance on test data (as long as the latter fairly represents a spectrum of weather conditions) is rather stable.

5.3.4 Joint versus product of marginals

One of the crucial points we wanted to test is weather the manifestation of composite classes (e.g., agent-action pairs or road events) can be estimated by separately training models for the individual types of labels, to then combine the resulting scores by simple multiplication (under an implicit, naive assumption of independence). This would have the advantage of not having to train separate networks on all sort of composite labels, an obvious positive in terms of efficiency, especially if we imagine to further extend in the

TABLE 8
Comparison of joint vs product of marginals approaches with I3D backbone. Number of video-/frame-level instances for each composite label ('No instances' column) and corresponding frame-/video-level results (mAP@%) averaged across all three splits, on both validation and test sets.

	No instances	Frame-mAP@0.5/Video-mAP@0.2			
Eval-method		Joint		Prod. of marginals	
Eval subset	All	Val	Test	Val	Test
Duplexes	603274/9335	21.9/21.4	31.0/25.5	21.6/21.2	30.8/24.3
Event	453626/8394	12.6/15.9	16.8/17.9	13.7/15.4	16.3/16.1

future the set of labels to other relevant aspects of the scene, such as attributes (e.g. vehicle speed). This would also give the system the flexibility to be extended to new composite events in the future without need for retraining.

For instance, we may want to test the hypothesis that the score for the pair *Pedestrian / Moving away* can be approximated as $P_{Ag}(\text{Pedestrian}) \times P_{Ac}(\text{Moving away})$, where P_{Ag} and P_{Ac} are the likelihood functions associated with the individual agent and action detectors[9]. This boils down to testing whether we need to explicitly learn a model for the joint distribution of the labels, or we can approximate that joint as a product of marginals. Learning-wise, the latter task involves a much smaller search space, so that marginal solutions (models) can be obtained more easily.

Table 8 compares the detection performance on composite (duplex or event) labels obtained by expressly training a detection network for those ('Joint' column) as opposed to simply multiplying the detector scores generated by the networks trained on individual labels ('Prod. of marginals'). The results clearly validate the hypothesis that it is possible to model composite labels using predictions for individual labels without having to train on the former. In most cases, the product of marginals approach achieves results similar or even better than those of joint prediction, although in some case (e.g. *Traffic light red* and *Traffic light red*, see Supplementary material again) we can observe a decrease in performance. We believe this to be valuable insight for further research.

5.3.5 Results of AV-action segmentation

Finally, Table 9 shows the results of using 3D-RetinaNet to temporally segment AV-action classes, averaged across all three splits on both validation and test set. As we can see, the results for classes *AV-move* and *AV-stop* are very good, we think because these two classes are predominately present in the dataset. The performance of the 'turning' classes is reasonable, but the results for the bottom three classes are really disappointing. We believe this is mainly due the fact that the dataset is very heavily biased (in terms of number of instances) towards the other classes. As we do intend to further expand this dataset in the future by including more and more videos, we hope the class imbalance issue can be mitigated over time. A measure of performance weighing mAP using the number of instances per class could be considered, but this is not quite standard in the action detection literature. At the same time, ROAD provides an opportunity for testing methods designed to address class imbalance.

9. Technically the networks output scores, not probabilities, but those can be easily calibrated to probability values.

TABLE 9
AV-action temporal segmentation results (frame mAP%) averaged across all three splits.

	No instances	Frame-mAP@0.5			
Model		I3D		2D	
Eval subset	All	Val	Test	Val	Test
Av-move	81196/233	92.0	96.6	83.0	87.8
Av-stop	31801/108	92.2	98.5	65.3	68.4
Av-turn-right	3826/50	46.1	63.0	35.0	57.7
Av-turn-left	3787/56	69.0	59.8	55.1	42.9
Av-overtake	599/12	4.9	1.1	2.7	2.5
Av-move-left	537/15	0.5	0.8	0.5	0.5
Av-move-right	408/16	10.5	0.6	4.0	2.0
Total/Mean	122154/490	45.0	45.8	35.1	37.4

TABLE 10
Results (in video-mAP) of the winning entries to the ICCV 2021 ROAD challenge compared with the Slowfast and YOLOv5 baselines, at a detection threshold of 0.2.

Task	Top team	Slowfast	YOLOv5	Winners
Agent detection	Xidian	29.0	43.3	**52.4**
Action detection	CMU-INF	20.5	-	**25.6**
Event detection	IFLY	22.4	-	**24.7**

5.4 Challenge Results

Table 10 compares the results of the top teams participating in our ROAD @ ICCV 2021 challenge with those of the Slowfast and YOLOv5 baselines, at a tube detection threshold of 0.2. The challenge server remains open at https://eval.ai/web/challenges/challenge-page/1059/overview, where one can consult the latest entries.

Agent detection. The agent detection challenge was won by a team formed by Chenghui Li, Yi Cheng, Shuhan Wang, Zhongjian Huang, Fang Liu of Xidian University, with an entry using YOLOv5 with post-processing. In their approach, agents are linked by evaluating their similarity between frames and grouping them into a tube. Discontinuous tubes are completed through frame filling, using motion information. Also, the authors note that YOLOv5 generates some incorrect bounding boxes, scattered in different frames, and take advantage of this by filtering out the shorter tubes. As shown in Table 10, the postprocessing applied by the winning entry significantly outperforms our off-the-shelf implementation of YOLOv5 on agent detection.

Action detection. The action detection challenge was won by Lijun Yu, Yijun Qian, Xiwen Chen, Wenhe Liu and Alexander G. Hauptmann of team CMU-INF, with an entry called "ArgusRoad: Road Activity Detection with Connectionist Spatiotemporal Proposals", based on their Argus++ framework for real-time activity recognition in extended videos in the NIST ActEV (Activities in Extended Video ActEV) challenge[10]. The had to adapt their system to be run on ROAD, e.g. to construct tube proposals rather than frame-level proposals. The approach is a rather complex cascade of object tracking, proposal generation, activity recognition and temporal localisation stages [108]. Results

10. https://actev.nist.gov/.

show a significant (5%) improvement over the Slowfast baseline, which is close to state-of-the-art in action detection, but still at a relatively low level (25.6%)

Event detection. The event detection challenge was won by team IFLY (Yujie Hou and Fengyan Wang, from the University of Science and Technology of China and IFLYTEK). The entry consisted in a number of amendments to the 3D-RetinaNet baseline, namely: bounding box interpolation, tuning of the optimiser, ensemble feature extraction with RCN, GRU and LSTM units, together with some data augmentation. Results show an improvement of above 2% over Slowfast, which suggests event better performance could be achieved by applying the ensemble technique to the latter.

6 FURTHER EXTENSIONS

By design, ROAD is an open project which we expect to evolve and grow over time.

Extension to other datasets and environments. In the near future we will work towards completing the multi-label annotation process for a larger number of frames coming from videos spanning an even wider range of road conditions. Further down the line, we plan to extend the benchmark to other cities, countries and sensor configurations, to slowly grow towards an even more robust, 'in the wild' setting. In particular, we will initially target the Pedestrian Intention Dataset (PIE, [58]) and Waymo [109]. The latter one comes with spatiotemporal tube annotation for pedestrian and vehicles, much facilitating the extension of ROAD-like event annotation there.

Event anticipation/intent prediction. ROAD is an oven-ready playground for action and event anticipation algorithms, a topic of growing interest in the vision community [110], [111], as it already provides the kind of annotation that allows researchers to test predictions of both future event labels and future event locations, both spatial and temporal. Anticipating the future behaviour of other road agents is crucial to empower the AV to react timely and appropriately. The output of this Task should be in the form of one or more future tubes, with the scores of the associated class labels and the future bounding box locations in the image plane [88]. We will shortly propose a baseline method for this Task, but we encourage researchers in the area to start engaging with the dataset from now.

Autonomous decision making. In accordance with our overall philosophy, we will design and share a baseline for AV decision making from intermediate semantic representations. The output of this Task should be the decision made by the AV in response to a road situation [112], represented as a collection of events as defined in this paper. As the action performed by the AV at any given time is part of the annotation, the necessary meta-data is already there. Although we did provide a simple temporal segmentation baseline for this task seen as a classification problem, we intend in the near future to propose a baseline from a decision making point of view, making use of the intermediate semantic representations produced by the detectors.

Machine theory of mind [113] refers to the attempt to provide machines with (limited) ability to guess the reasoning process of other intelligent agents they share the environment with. Building on our efforts in this area [14], we will work with teams of psychologists and neuroscientists to provide annotations in terms of mental states and reasoning processes for the road agents present in ROAD. Note that theory of mind models can also be validated in terms of how close the predictions of agent behaviour they are capable of generating are to their actual observed behaviour. Assuming that the output of a theory of mind model is intention (which is observable and annotated) the same baseline as for event anticipation can be employed.

Continual event detection. ROAD's conceptual setting is intrinsically incremental, one in which the autonomous vehicle keeps learning from the data it observes, in particular by updating the models used to estimate the intermediate semantic representations. The videos forming the dataset are particularly suitable, as they last 8 minutes each, providing a long string of events and data to learn from. To this end, we plan to set a protocol for the continual learning of event classifiers and detectors and propose ROAD as the first continual learning benchmark in this area [114].

7 CONCLUSIONS

This paper proposed a strategy for situation awareness in autonomous driving based on the notion of road events, and contributed a new ROad event Awareness Dataset for Autonomous Driving (ROAD) as a benchmark for this area of research. The dataset, built on top of videos captured as part of the Oxford RobotCar dataset [18], has unique features in the field. Its rich annotation follows a multi–label philosophy in which road agents (including the AV), their locations and the action(s) they perform are all labelled, and road events can be obtained by simply composing labels of the three types. The dataset contains 22 videos with 122K annotated video frames, for a total of 560K detection bounding boxes associated with 1.7M individual labels.

Baseline tests were conducted on ROAD using a new 3D-RetinaNet architecture, as well as a Slowfast backbone and a YOLOv5 model (for agent detection). Both frame–mAP and video–mAP were evaluated. Our preliminary results highlight the challenging nature of ROAD, with the Slowfast baseline achieving a video-mAP on the three main tasks comprised between 20% and 30%, at low localisation precision (20% overlap). YOLOv5, however, was able to achieve significantly better performance. These findings were reinforced by the results of the ROAD @ ICCV 2021 challenge, and support the need for an even broader analysis, while highlighting the significant challenges specific to situation awareness in road scenarios.

Our dataset is extensible to a number of challenging tasks associated with situation awareness in autonomous driving, such as event prediction, trajectory prediction, continual learning and machine theory of mind, and we pledge to further enrich it in the near future by extending ROAD-like annotation to major datasets such as PIE and Waymo.

ACKNOWLEDGMENTS

This project has received funding from the European Union's Horizon 2020 research and innovation programme, under grant agreement No. 964505 (E-pi). The authors would like to thank Petar Georgiev, Adrian Scott, Alex Bruce

and Arlan Sri Paran for their contribution to video annotation. We also wish to acknowledge the members of the ROAD challenge's winning teams: Chenghui Li, Yi Cheng, Shuhan Wang, Zhongjian Huang, Fang Liu, Lijun Yu, Yijun Qian, Xiwen Chen, Wenhe Liu, Alexander G. Hauptmann, Yujie Hou and Fengyan Wang.

REFERENCES

[1] J. Winn and J. Shotton, "The layout consistent random field for recognizing and segmenting partially occluded objects," in *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, vol. 1. IEEE, 2006, pp. 37–44.

[2] K. Korosec, "Toyota is betting on this startup to drive its self-driving car plans forward," Available at: http://fortune.com/2017/09/27/toyota-self-driving-car-luminar/.

[3] G. Pandey, J. R. McBride, and R. M. Eustice, "Ford campus vision and lidar data set," *International Journal of Robotics Research*, vol. 30, no. 13, pp. 1543–1552, 2011.

[4] M. e. a. Maurer, *Autonomous driving: technical, legal and social aspects*. Springer Nature, 2016.

[5] A. Broggi, Alberto et al. C. Laugier, "Intelligent vehicles," in *Springer Handbook of Robotics*. Springer, 2016, pp. 1627–1656.

[6] S. Azam, F. Munir, A. Rafique, Y. Ko, A. M. Sheri, and M. Jeon, "Object modeling from 3d point cloud data for self-driving vehicles," in *2018 IEEE Intelligent Vehicles Symposium (IV)*, June 2018, pp. 409–414.

[7] Z. Fang and A. M. López, "Is the pedestrian going to cross? answering by 2d pose estimation," in *2018 IEEE Intelligent Vehicles Symposium (IV)*, June 2018, pp. 1271–1276.

[8] P. Wang, C. Chan, and A. d. L. Fortelle, "A reinforcement learning based approach for automated lane change maneuvers," in *2018 IEEE Intelligent Vehicles Symposium (IV)*, June 2018, pp. 1379–1384.

[9] J. Chen, C. Tang, L. Xin, S. E. Li, and M. Tomizuka, "Continuous decision making for on-road autonomous driving under uncertain and interactive environments," in *2018 IEEE Intelligent Vehicles Symposium (IV)*, June 2018, pp. 1651–1658.

[10] M. Bertozzi, A. Broggi, and A. Fascioli, "Vision-based intelligent vehicles: State of the art and perspectives," *Robotics and Autonomous Systems*, vol. 32, no. 1, pp. 1 – 16, 2000.

[11] J. Redmon and A. Farhadi, "Yolo9000: Better, faster, stronger," in *IEEE Int. Conf. on Computer Vision and Pattern Recognition*, 2017, pp. 6517—-6525.

[12] M. Codevilla, Felipe Dosovitskiy, "End-to-end driving via conditional imitation learning," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2018, pp. 1–9.

[13] L. F. et al., "Arguing machines: Perception-control system redundancy and edge case discovery in real-world autonomous driving," ArXiv preprint ArXiv:1710.04459, 2017.

[14] F. Cuzzolin, A. Morelli, B. Cirstea, and B. J. Sahakian, "Knowing me, knowing you: Theory of mind in AI," *Psychological Medicine*, vol. 50, no. 7, pp. 1057–1061, May 2020.

[15] A. Rasouli and J. K. Tsotsos, "Autonomous vehicles that interact with pedestrians: A survey of theory and practice," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 3, pp. 900—-918, 2020.

[16] A. Rudenko, L. Palmieri, M. Herman, K. M. Kitani, D. M. Gavrila, and K. O. Arras, "Human motion trajectory prediction: A survey," arXiv preprint arXiv1905.06113, 2019.

[17] S. Armstrong and S. Mindermann, "Occam's razor is insufficient to infer the preferences of irrational agents," in *Advances in Neural Information Processing Systems*, vol. 31, 2018, pp. 5603—-5614.

[18] W. Maddern, G. Pascoe, C. Linegar, and P. Newman, "1 year, 1000 km: The oxford robotcar dataset," *The International Journal of Robotics Research*, vol. 36, no. 1, pp. 3–15, 2017.

[19] G. Singh, S. Saha, M. Sapienza, P. Torr, and F. Cuzzolin, "Online real-time multiple spatiotemporal action localisation and prediction," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 3637–3646.

[20] S. Saha, G. Singh, M. Sapienza, P. H. Torr, and F. Cuzzolin, "Deep learning for detecting multiple space-time action tubes in videos," *arXiv preprint arXiv:1608.01529*, 2016.

[21] G. Gkioxari and J. Malik, "Finding action tubes," in *IEEE Int. Conf. on Computer Vision and Pattern Recognition*, 2015.

[22] C. Feichtenhofer, H. Fan, J. Malik, and K. He, "Slowfast networks for video recognition," in *Proceedings of the IEEE international conference on computer vision*, 2019, pp. 6202–6211.

[23] G. J. Brostow, J. Shotton, J. Fauqueur, and R. Cipolla, "Segmentation and recognition using structure from motion point clouds," in *European conference on computer vision*, 2008, pp. 44–57.

[24] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *Proceedings of CVPR 2016*, 2016, pp. 3213–3223.

[25] G. Neuhold, T. Ollmann, S. Rota Bulo, and P. Kontschieder, "The mapillary vistas dataset for semantic understanding of street scenes," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 4990–4999.

[26] F. Yu, W. Xian, Y. Chen, F. Liu, M. Liao, V. Madhavan, and T. Darrell, "Bdd100k: A diverse driving video database with scalable annotation tooling," *arXiv preprint arXiv:1805.04687*, 2018.

[27] P. Wang, X. Huang, X. Cheng, D. Zhou, Q. Geng, and R. Yang, "The apolloscape open dataset for autonomous driving and its application," *IEEE transactions on pattern analysis and machine intelligence*, 2019.

[28] Z. Che, G. Li, T. Li, B. Jiang, X. Shi, X. Zhang, Y. Lu, G. Wu, Y. Liu, and J. Ye, "D ^{2}-city: A large-scale dashcam video dataset of diverse traffic scenarios," *arXiv preprint arXiv:1904.01975*, 2019.

[29] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1. IEEE, 2005, pp. 886–893.

[30] A. Ess, B. Leibe, K. Schindler, and L. Van Gool, "A mobile vision system for robust multi-person tracking," in *2008 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2008, pp. 1–8.

[31] C. Wojek, S. Walk, and B. Schiele, "Multi-cue onboard pedestrian detection," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2009, pp. 794–801.

[32] M. Enzweiler and D. M. Gavrila, "Monocular pedestrian detection: Survey and experiments," *IEEE transactions on pattern analysis and machine intelligence*, vol. 31, no. 12, pp. 2179–2195, 2008.

[33] S. Zhang, R. Benenson, and B. Schiele, "Citypersons: A diverse dataset for pedestrian detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 3213–3221.

[34] L. Neumann, M. Karg, S. Zhang, C. Scharfenberger, E. Piegert, S. Mistr, O. Prokofyeva, R. Thiel, A. Vedaldi, A. Zisserman *et al.*, "Nightowls: A pedestrians at night dataset," in *Asian Conference on Computer Vision*. Springer, 2018, pp. 691–705.

[35] V. T. Covello and M. W. Merkhofer, "An evaluation of the state of the art," in *Risk Assessment Methods*. Springer, 1993, pp. 239–265.

[36] L. Ding, J. Terwilliger, R. Sherony, B. Reimer, and L. Fridman, "MIT DriveSeg (Manual) Dataset," 2020.

[37] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2012, pp. 3354–3361.

[38] A. Patil, S. Malla, H. Gang, and Y.-T. Chen, "The h3d dataset for full-surround 3d multi-object detection and tracking in crowded urban scenes," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 9552–9557.

[39] Y. Choi, N. Kim, S. Hwang, K. Park, J. S. Yoon, K. An, and I. S. Kweon, "Kaist multi-spectral day/night data set for autonomous and assisted driving," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 3, pp. 934–948, 2018.

[40] J.-L. Blanco-Claraco, F.-Á. Moreno-Dueñas, and J. González-Jiménez, "The málaga urban dataset: High-rate stereo and lidar in a realistic urban scenario," *The International Journal of Robotics Research*, vol. 33, no. 2, pp. 207–214, 2014.

[41] H. Jung, Y. Oto, O. M. Mozos, Y. Iwashita, and R. Kurazume, "Multi-modal panoramic 3d outdoor datasets for place categorization," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2016, pp. 4545–4550.

[42] Y. Chen, J. Wang, J. Li, C. Lu, Z. Luo, H. Xue, and C. Wang, "Lidar-video driving dataset: Learning driving policies effectively," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5870–5878.

[43] J. Chang, Ming-Fang D. Wang, P. Carr, S. Lucey, D. Ramanan, and others et al., "Argoverse: 3d tracking and forecasting with rich

maps," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 8748–8757.

[44] R. Kesten, M. Usman, J. Houston, T. Pandya, K. Nadhamuni, A. Ferreira, M. Yuan, B. Low, A. Jain, P. Ondruska *et al.*, "Lyft level 5 av dataset 2019," *urlhttps://level5. lyft. com/dataset*, 2019.

[45] P. Sun, H. Kretzschmar, X. Dotiwalla, A. Chouard, V. Patnaik, P. Tsui, J. Guo, Y. Zhou, Y. Chai, B. Caine, V. Vasudevan, W. Han, J. Ngiam, H. Zhao, A. Timofeev, S. Ettinger, M. Krivokon, A. Gao, A. Joshi, Y. Zhang, J. Shlens, Z. Chen, and D. Anguelov, "Scalability in perception for autonomous driving: Waymo open dataset," 2019.

[46] Q.-H. Pham, P. Sevestre, R. S. Pahwa, H. Zhan, C. H. Pang, Y. Chen, A. Mustafa, V. Chandrasekhar, and J. Lin, "A* 3d dataset: Towards autonomous driving in challenging environments," *arXiv preprint arXiv:1909.07541*, 2019.

[47] J. Geyer, Y. Kassahun, M. Mahmudi, X. Ricou, R. Durgesh, A. S. Chung, L. Hauswald, V. H. Pham, M. Mühlegg, S. Dorn *et al.*, "A2d2: Aev autonomous driving dataset," *Note: http://www. a2d2. audi Cited by*, vol. 1, no. 4, 2019.

[48] Y. Liao, J. Xie, and A. Geiger, "KITTI-360: A novel dataset and benchmarks for urban scene understanding in 2d and 3d," *arXiv.org*, vol. 2109.13410, 2021.

[49] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuscenes: A multimodal dataset for autonomous driving," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 11 621–11 631.

[50] A. Lerner, Y. Chrysanthou, and D. Lischinski, "Crowds by example," in *Computer graphics forum*, vol. 26, no. 3. Wiley Online Library, 2007, pp. 655–664.

[51] S. Pellegrini, A. Ess, and L. Van Gool, "Improving data association by joint modeling of pedestrian trajectories and groupings," in *European conference on computer vision*. Springer, 2010, pp. 452–465.

[52] S. Oh, A. Hoogs, A. Perera, N. Cuntoor, C.-C. Chen, J. T. Lee, S. Mukherjee, J. Aggarwal, H. Lee, L. Davis *et al.*, "A large-scale benchmark dataset for event recognition in surveillance video," in *CVPR 2011*. IEEE, 2011, pp. 3153–3160.

[53] A. Robicquet, A. Sadeghian, A. Alahi, and S. Savarese, "Learning social etiquette: Human trajectory understanding in crowded scenes," in *European conference on computer vision*. Springer, 2016, pp. 549–565.

[54] Y. Yao, M. Xu, C. Choi, D. J. Crandall, E. M. Atkins, and B. Dariush, "Egocentric vision-based future vehicle localization for intelligent driving assistance systems," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 9711–9717.

[55] R. Chandra, U. Bhattacharya, A. Bera, and D. Manocha, "Traphic: Trajectory prediction in dense and heterogeneous traffic using weighted interactions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 8483–8492.

[56] A. Rasouli, I. Kotseruba, and J. K. Tsotsos, "Are they going to cross? a benchmark dataset and baseline for pedestrian crosswalk behavior," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2017, pp. 206–213.

[57] R. Q. Mínguez, I. P. Alonso, D. Fernández-Llorca, and M. Á. Sotelo, "Pedestrian path, pose, and intention prediction through gaussian process dynamical models and pedestrian activity recognition," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 5, pp. 1803–1814, 2018.

[58] A. Rasouli, I. Kotseruba, T. Kunic, and J. K. Tsotsos, "Pie: A large-scale dataset and models for pedestrian intention estimation and trajectory prediction," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 6262–6271.

[59] S. Malla, B. Dariush, and C. Choi, "Titan: Future forecast using action priors," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 11 186–11 196.

[60] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. M. Lopez, "The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 3234–3243.

[61] J. Behley, M. Garbade, A. Milioto, J. Quenzel, S. Behnke, C. Stachniss, and J. Gall, "Semantickitti: A dataset for semantic scene understanding of lidar sequences," 2019.

[62] Y. Jiang, J. Liu, A. Roshan Zamir, G. Toderici, I. Laptev, M. Shah, and R. Sukthankar, "Thumos challenge: Action recognition with a large number of classes," *http://crcv.ucf.edu/THUMOS14*, 2014.

[63] C. Gu, C. Sun, S. Vijayanarasimhan, C. Pantofaru, D. A. Ross, G. Toderici, Y. Li, S. Ricco, R. Sukthankar, C. Schmid *et al.*, "Ava: A video dataset of spatio-temporally localized atomic visual actions," *arXiv preprint arXiv:1705.08421*, 2017.

[64] Y. Li, L. Chen, R. He, Z. Wang, G. Wu, and L. Wang, "Multisports: A multi-person video dataset of spatio-temporally localized sports actions," *arXiv preprint arXiv:2105.07404*, 2021.

[65] J. Kay, Will S. Vijayanarasimhan, F. Viola, T. Green, T. Back, P. Natsev, and others et al., "The kinetics human action video dataset," *arXiv preprint arXiv:1705.06950*, 2017.

[66] M. Monfort, A. Andonian, B. Zhou, K. Ramakrishnan, S. A. Bargal, T. Yan, L. Brown, Q. Fan, D. Gutfruend, C. Vondrick *et al.*, "Moments in time dataset: one million videos for event understanding," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–8, 2019.

[67] R. Goyal, S. E. Kahou, V. Michalski, J. Materzyńska, S. Westphal, H. Kim, V. Haenel, I. Fruend, P. Yianilos, M. Mueller-Freitag, F. Hoppe, C. Thurau, I. Bax, and R. Memisevic, "The "something something" video database for learning and evaluating visual common sense," 2017.

[68] F. Caba Heilbron, V. Escorcia, B. Ghanem, and J. Carlos Niebles, "Activitynet: A large-scale video benchmark for human activity understanding," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 961–970.

[69] G. A. Sigurdsson, A. Gupta, C. Schmid, A. Farhadi, and K. Alahari, "Charades-ego: A large-scale dataset of paired third and first person videos," 2018.

[70] H. Jhuang, J. Gall, S. Zuffi, C. Schmid, and M. J. Black, "Towards understanding action recognition," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2013, pp. 3192–3199.

[71] K. Soomro, A. R. Zamir, and M. Shah, "Ucf101: A dataset of 101 human actions classes from videos in the wild," 2012.

[72] C. Wolf, J. Mille, E. Lombardi, O. Celiktutan, M. Jiu, M. Baccouche, E. Dellandréa, C.-E. Bichot, C. Garcia, and B. Sankur, "The LIRIS Human activities dataset and the ICPR 2012 human activities recognition and localization competition," LIRIS UMR 5205 CNRS/INSA de Lyon/Université Claude Bernard Lyon 1/Université Lumière Lyon 2/École Centrale de Lyon, Tech. Rep., 2012. [Online]. Available: http://liris.cnrs.fr/publis/?id=5498

[73] P. Weinzaepfel, X. Martin, and C. Schmid, "Human action localization with sparse spatial supervision," *arXiv preprint arXiv:1605.05197*, 2016.

[74] J. Carreira and A. Zisserman, "Quo vadis, action recognition? a new model and the kinetics dataset," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 4724–4733.

[75] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *CVPR*, 2018.

[76] C. Feichtenhofer, "X3d: Expanding architectures for efficient video recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 203–213.

[77] R. Girdhar, J. Carreira, C. Doersch, and A. Zisserman, "A better baseline for ava," *arXiv preprint arXiv:1807.10066*, 2018.

[78] C.-Y. Wu, C. Feichtenhofer, H. Fan, K. He, P. Krahenbuhl, and R. Girshick, "Long-term feature banks for detailed video understanding," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 284–293.

[79] K. Soomro, H. Idrees, and M. Shah, "Predicting the where and what of actors and actions through online action localization," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2648–2657.

[80] H. S. Behl, M. Sapienza, G. Singh, S. Saha, F. Cuzzolin, and P. H. Torr, "Incremental tube construction for human action detection," *arXiv preprint arXiv:1704.01358*, 2017.

[81] V. Kalogeiton, P. Weinzaepfel, V. Ferrari, and C. Schmid, "Action tubelet detector for spatio-temporal action localization," in *Proc. Int. Conf. Computer Vision*, 2017.

[82] Y. Li, Z. Wang, L. Wang, and G. Wu, "Actions as moving points," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

[83] X. Yang, X. Yang, M.-Y. Liu, F. Xiao, L. S. Davis, and J. Kautz, "Step: Spatio-temporal progressive learning for video action detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 264–272.

[84] S. Saha, G. Singh, and F. Cuzzolin, "Amtnet: Action-micro-tube regression by end-to-end trainable deep architecture," in *Proc. Int. Conf. Computer Vision*, 2017.
[85] X. Peng and C. Schmid, "Multi-region two-stream r-cnn for action detection," in *European Conference on Computer Vision*, 2016, pp. 744–759.
[86] J. Zhao and C. G. Snoek, "Dance with flow: Two-in-one stream action detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 9935–9944.
[87] G. Singh, S. Saha, and F. Cuzzolin, "Tramnet-transition matrix network for efficient action tube proposals," in *Asian Conference on Computer Vision*. Springer, 2018, pp. 420–437.
[88] ——, "Predicting action tubes," in *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, 2018, pp. 0–0.
[89] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2980–2988.
[90] J. Pan, S. Chen, M. Z. Shou, Y. Liu, J. Shao, and H. Li, "Actor-context-actor relation network for spatio-temporal action localization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 464–474.
[91] J. Tang, J. Xia, X. Mu, B. Pang, and C. Lu, "Asynchronous interaction aggregation for action detection," in *European Conference on Computer Vision*. Springer, 2020, pp. 71–87.
[92] V. Ramanishka, Y.-T. Chen, T. Misu, and K. Saenko, "Toward driving scene understanding: A dataset for learning driver behavior and causal reasoning," in *Conference on Computer Vision and Pattern Recognition*, 2018.
[93] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The kitti dataset," *The International Journal of Robotics Research*, vol. 32, no. 11, pp. 1231–1237, 2013.
[94] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *CVPR*, vol. 1, no. 2, 2017, p. 4.
[95] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.
[96] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Advances in neural information processing systems*, 2014, pp. 568–576.
[97] K. Soomro, A. Zamir, and M. Shah, "Ucf101: a dataset of 101 human action classes from videos in the wild (2012)," *arXiv preprint arXiv:1212.0402*, 2012.
[98] S. Saha, G. Singh, M. Sapienza, P. H. S. Torr, and F. Cuzzolin, "Deep learning for detecting multiple space-time action tubes in videos," in *British Machine Vision Conference*, 2016.
[99] P. Weinzaepfel, Z. Harchaoui, and C. Schmid, "Learning to track for spatio-temporal action localization," in *IEEE Int. Conf. on Computer Vision and Pattern Recognition*, June 2015.
[100] Z. Li, Dong et al. proposal and recognition networks for action detection," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 303–318.
[101] L. Song, S. Zhang, G. Yu, and H. Sun, "Tacnet: Transition-aware context network for spatio-temporal action detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 11 987–11 995.
[102] C. Gu, C. Sun, D. A. Ross, C. Vondrick, C. Pantofaru, Y. Li, S. Vijayanarasimhan, G. Toderici, S. Ricco, R. Sukthankar, C. Schmid, and J. Malik, "Ava: A video dataset of spatio-temporally localized atomic visual actions," 2018.
[103] T. Kong, F. Sun, H. Liu, Y. Jiang, and J. Shi, "Consistent optimization for single-shot object detection," 2019.
[104] M. Li, Y.-X. Wang, and D. Ramanan, "Towards streaming perception," in *European Conference on Computer Vision*. Springer, 2020, pp. 473–488.
[105] G. SingH and F. Cuzzolin, "Recurrent convolutions for causal 3d cnns," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2019, pp. 0–0.
[106] Z. Liu, J. Ning, Y. Cao, Y. Wei, Z. Zhang, S. Lin, and H. Hu, "Video swin transformer," *arXiv preprint arXiv:2106.13230*, 2021.
[107] H. Fan, B. Xiong, K. Mangalam, Y. Li, Z. Yan, J. Malik, and C. Feichtenhofer, "Multiscale vision transformers," *arXiv preprint arXiv:2104.11227*, 2021.
[108] W. Liu, G. Kang, P.-Y. Huang, X. Chang, Y. Qian, J. Liang, L. Gui, J. Wen, and P. Chen, "Argus: Efficient activity detection system for extended video analysis," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision Workshops*, 2020, pp. 126–133.
[109] P. Sun, H. Kretzschmar, X. Dotiwalla, A. Chouard, V. Patnaik, P. Tsui, J. Guo, Y. Zhou, Y. Chai, B. Caine *et al.*, "Scalability in perception for autonomous driving: Waymo open dataset," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 2446–2454.
[110] Y. Kong, Z. Tao, and Y. Fu, "Deep sequential context networks for action prediction," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1473–1481.
[111] ——, "Adversarial action prediction networks," *IEEE transactions on pattern analysis and machine intelligence*, vol. 42, no. 3, pp. 539–553, 2018.
[112] C. Hubmann, M. Becker, D. Althoff, D. Lenz, and C. Stiller, "Decision making for autonomous driving considering interaction and uncertain prediction of surrounding vehicles," in *2017 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2017, pp. 1671–1678.
[113] N. Rabinowitz, F. Perbet, F. Song, C. Zhang, S. A. Eslami, and M. Botvinick, "Machine theory of mind," in *International conference on machine learning*. PMLR, 2018, pp. 4218–4227.
[114] G. I. Parisi, R. Kemker, J. L. Part, C. Kanan, and S. Wermter, "Continual lifelong learning with neural networks: A review," *Neural Networks*, vol. 113, pp. 54–71, 2019.

Gurkirt Singh received his Bachelor of Technology degree in Electronics and Instrumentation Engineering from VIT University, Vellore, India. He completed his master's thesis under the supervision of Dr. Georgios Evangelidis and Dr. Radu HORAUD at INRIA, Grenoble. He graduated from MOSIG master program at Grenoble-INPG (School ENSIMAG) with specialization in Graphics Vision and Robotics. After two years as a research engineer with Siemens Research India, he received a PhD from Oxford Brookes University under the supervision of Prof Fabio Cuzzolin. He is now a postdoctoral researcher with the Computer Vision Lab of ETH Zurich.

Stephen Akrigg is a computer scientist currently working in the medical science division at the University of Oxford. He has mainly been a generalist in the field of IT, but has recently focused his studies on computer vision, due to his interests in photography and how machines perceive and learn in contrast to us, as well as the enormous potential and opportunities this field could contribute to shaping the future.

Manuele Di Maio works as Automation Engineer for Siemens in Italy. He studied Control Engineering at University of Naples Federico II (Italy), where he graduated with the highest marks. During his Master's degree he spent six months at Oxford Brookes University (UK), where he kickstarted the collaboration with Prof Fabio Cuzzolin on ROAD. In 2020 his dissertation won the "Smart Solution For Sustainable Mobility Award" promoted by University of Naples Federico II in Memory of Davide Natale.

Valentina Fontana received both her Master and Bachelor degrees in Automation Engineering from University of Naples Federico II, Italy, in 2016 and 2018, respectively. For her Master dissertation she worked with Prof Fabio Cuzzolin on action recognition for autonomous vehicles at the Visual AI Lab in Oxford Brookes University (UK), under the supervision of Prof Fabio Cuzzolin (Oxford Brookes), Prof Giuseppe Di Gironimo and Dr Stanislao Grazioso (Dept. of Industrial Engineering, Federico II University).

Reza Javanmard Alitappeh is currently an Assistant Professor at the University of Science and Technology of Mazandaran, Iran. In 2019 he worked as a research fellow at Visual Artificial Intelligence Laboratory in Oxford Brookes University (UK) with Prof Fabio Cuzzolin and Dr Bradley. He completed his PhD in 2016 in artificial intelligence and robotics under the supervision of Prof Pimenta and Chaimowicz at the Federal University of Minas Gerais, Brazil. He is a reviewer for various journals and conferences.

Suman Saha is a post doctoral research fellow at the Computer Vision Lab (CVL), ETH Zurich, Switzerland. He works with Prof Luc Van Gool at CVL. Before joining ETH, he was a Research Associate at the Visual AI Lab, Oxford Brookes University (UK). He received his PhD degree in Computer Science and Mathematics there under the supervision of Prof Fabio Cuzzolin. Earlier he had received an MSc degree in Computer Science from University of Bedfordshire (UK).

Kossar Jeddisaravi received her MSc degree in Computer Systems and Robotics from the Federal University of Minas Gerias, Brazil, in 2014, with a dissertation on multi-objective robot exploration. She obtained a PhD from the same university in 2017, under the supervision of Prof Frederico Guimaraes. Since 2017 she is a lecturer in programming language and image processing at the University of Science and Technology of Mazandaran. Her research interests include multi-agent systems and machine vision.

Farzad Yousefi recently obtained a bachelor in Computer Engineering from the University of Science and Technology of Mazandaran. One of the 10 top students in his course, he also was Teaching Assistant for Computational intelligence and Natural language processing. He worked on several projects including facial expressions recognition and license plate detection For his bachelor project he developed a chatbot for the Persian language under the supervision of Prof Reza Javanmard.

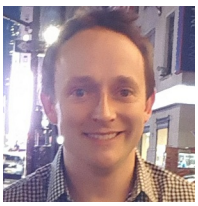
Jacob Culley recently graduated from Oxford Brookes with a BSc in Computer Science, where he led the University's *OBR Autonomous* team in the development of a self driving racing car, supervised by Dr. Bradley. He led the team to victory in the 2020 IMechE *Formula Student: Artificial Intelligence* competition, and now specialises in developing unmanned ground vehicles for off-highway applications.

Tom Nicholson is a final year MEng student in Mechanical Engineering at Oxford Brookes University, UK. He joined the ROAD project through his involvement in OBR EV, the University's Electric Vehicle racing team working closely with OBR Autonomous under Dr Bradley. He recently completed a dissertation on the use of disbondable structural adhesives for battery electric vehicles, and is now part of the group designing the new motors for the car.

Jordan Omokeowa recently graduated from Oxford Brookes University with a BSc in Motorsport Technology, having joined the ROAD team through Dr Bradley's Vehicle Dynamics lectures on real-time vehicle control. Jordan currently works as a .NET software developer at the GreatPlaces housing group, and as a freelance web developer.

Salman Khan received his Master's degree in Computer Vision from Sejong University, Seoul, Republic of Korea in 2020 with research in vision-based fire/smoke detection. Currently, he is pursuing PhD degree in Computer Vision (Deep Learning for Modelling Complex Video Activities) from Oxford Brookes University, Oxford, United Kingdom. He is working as a research assistant at Visual Artificial Intelligence Laboratory (VAIL) from February 2020.

Stanislao Grazioso received both a Laurea in Mechanical Engineering and a PhD in Industrial Engineering from the University of Naples Federico II in 2014 and 2018, respectively. Since 2014 he is a PostDoc at the Department of Industrial Engineering of University of Naples Federico II. In 2016 he was a Visiting Scholar at both the University of Maryland and GeorgiaTech under a EUROfusion research grant. In 2019 he received the "Georges Giralt PhD Award" for the best doctorate thesis in robotics in Europe.



Andrew Bradley leads the Autonomous Driving Research Group at Oxford Brookes University, UK. His research interests are in autonomous driving, vehicle dynamics and real-time vehicle control. He completed a PhD in computer vision for vehicle dynamic analysis and collaborates with Prof Fabio Cuzzolin to bring enhanced perception capabilities to autonomous driving. Andrew oversees Oxford Brookes' award-winning autonomous racing team OBR Autonomous.

Giuseppe Di Gironimo is Professor of Virtual Prototyping and Head of the Virtual Reality ("MARTE"), Motion Analysis ("ERGOS") and "IDEAinVR" Labs at University of Naples Federico II. His main research interests are virtual and augmented reality, ergonomics and human-centered design, biomechanics and human motion analysis, soft robotics, human-robot interaction. He is the Head of Mechanical Design Unit, member of the Scientific Board of the CREATE Consortium and founder and President of BeyondShape, a spinoff company of the University of Naples active in the design and development of 3D Medical Scanning Systems.

Fabio Cuzzolin received a *laurea* degree magna cum laude in Computer Engineering from the University of Padua, Italy, in 1997. He was awarded a PhD by the same institution in 2001 for the thesis *Visions of a generalized probability theory*. After conducting research at Politecnico di Milano, the Washington University in St Louis, UCLA and INRIA Rhone-Alpes, he joined Oxford Brookes University (UK) where he is currently a Professor of Artificial Intelligence and the Director of the Visual Artificial Intelligence Laboratory. He is a world expert in the theory of belief functions, and the author of about 110 peer-reviewed publications.