DoTA: Unsupervised Detection of Traffic Anomaly in Driving Videos

Yu Yao, *Member, IEEE*, Xizi Wang, *Member, IEEE*, Mingze Xu, *Member, IEEE*, Zelin Pu, Yuchen Wang, *Member, IEEE*, Ella Atkins, *Senior Member, IEEE*, and David J. Crandall, *Member, IEEE*

Abstract—Video anomaly detection (VAD) has been extensively studied for static cameras but is much more challenging in egocentric driving videos where the scenes are extremely dynamic. This paper proposes an unsupervised method for traffic VAD based on future object localization. The idea is to predict future locations of traffic participants over a short horizon, and then monitor the accuracy and consistency of these predictions as evidence of an anomaly. Inconsistent predictions tend to indicate an anomaly has occurred or is about to occur. To evaluate our method, we introduce a new large-scale benchmark dataset called Detection of Traffic Anomaly (DoTA)containing 4,677 videos with temporal, spatial, and categorical annotations. We also propose a new VAD evaluation metric, called spatial-temporal area under curve (STAUC), and show that it captures how well a model detects both temporal and spatial locations of anomalies unlike existing metrics that focus only on temporal localization. Experimental results show our method outperforms state-of-the-art methods on DoTA in terms of both metrics. We offer rich categorical annotations in DoTA to benchmark video action detection and online action detection methods. *The DoTA dataset has been made available at: https://github.com/MoonBlvd/Detection-of-Traffic-Anomaly*

Index Terms—Video anomaly detection, Traffic accident detection, Future object localization, Video action recognition

1 Introduction

Autonomous driving has the potential to transform the world as we know it, revolutionizing transportation to be faster, safer, cheaper, and less labor intensive. But building autonomous systems that can accurately perceive and safely react to the huge diversity of situations that are encountered on real-world roadways is a major challenge. Driving obeys a long-tailed distribution, such that a few common situations make up the vast majority of what a driver encounters, while a virtually infinite variety of rare scenarios — animals running into the roadway, cars driving on the wrong side of the street, etc. — make up the rest. While each of these individual unusual scenarios is rare, they can and do happen. In fact, the chances that *one* of them will occur on any given day are actually quite high.

Much work in computer vision has studied detecting anomalous events from dashboard-mounted cameras [1], [2], [3]. Most of these methods focus on identifying frames in which an anomaly is occurring, but do not attempt to localize where in the frame the anomaly is happening or to identify which road participants are involved — critical information for real-world driving assistance systems. Other work on spatio-temporal action recognition (STAR) explores simultaneously predicting the bounding boxes (called "action tubes") of action performers and their associated classes [4], [5], [6], [7]. Applying these approaches to detecting anomalous events would require large-scale training data for each of a pre-defined set of events. However, the long-tailed distribution of driving events means that it may be impossible to collect training data for rare scenarios, or to even anticipate which scenarios might occur [8]. In fact, some studies indicate that driverless cars would need to be tested for *billions* of miles before enough of these rare situations occur to even accurately measure system safety [9], much less to collect sufficient training data to make them work well.

This paper proposes an alternative approach based on unsupervised video anomaly detection (VAD), which avoids modeling all possible driving events by recognizing "normal" roadway conditions and then signaling an anomaly when events that do not fit the model are observed. Any observation that does not fit a normal model is assigned a high anomaly score, regardless of its anomaly type. Unlike fully-supervised classification-based work or STAR, this unsupervised approach may not be able to identify exactly which anomaly has occurred, but it can potentially capture any type of anomalous event, even those not previously observed during training [10]. Unsupervised VAD has been widely applied to static surveillance camera datasets [8], [11], [12], [13], [14], [15] by training deep neural networks to reconstruct or predict video

- *Yu Yao, Zelin Pu, and Ella Atkins are with the Robotics Institute, University of Michigan, Ann Arbor, MI 48109 USA. E-mail: {brianyao, foxtony, ematkins}@umich.edu.*
- *Xizi Wang, Mingze Xu, Yuchen Wang, and David J. Crandall are with the Luddy School of Informatics, Computing and Engineering, Indiana University, Bloomington, IN 47408 USA. E-mail: {xiziwang, mx6, wang617, djcran}@iu.edu.*

Manuscript received 12 September 2020; revised 20 October 2021; accepted 26 January 2022. Date of publication 14 February 2022; date of current version 5 December 2022.
This work was supported in part by the National Science Foundation under Grants CNS-1544844 and CAREER IIS-1253549, in part by the U.S. Navy under Grant N00164-21-1-1002, and in part by the IU Office of the Vice Provost for Research and the Luddy School of Informatics, Computing, and Engineering through the Emerging Areas of Research Project "Learning: Brains, Machines, and Children."
(Corresponding author: Yu Yao.)
Recommended for acceptance by B. Ghanem.
Digital Object Identifier no. 10.1109/TPAMI.2022.3150763

0162-8828 © 2022 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.
Authorized licensed use limited to: Sungkyunkwan University. Downloaded on June 21,2023 at 11:45:09 UTC from IEEE Xplore. Restrictions apply.

frames and computing the reconstruction or prediction errors as anomaly scores. However, these methods do not generalize well to driving videos since frame prediction and reconstruction are extremely difficult when cameras are rapidly moving, as in the driving scenario.

This paper side-steps the difficulty of predicting whole future frames for traffic VAD by tracking objects and predicting their future locations as opposed to attempting prediction of whole frames. We propose a novel approach that learns a future object localization (FOL) network for traffic participants (e.g., cars, bikes, pedestrians, etc.) in the field of view of a dashboard-mounted camera on a moving ego-vehicle. Our FOL model consists of two modules: an ego-motion Recurrent Neural Network (RNN) encoder-decoder to predict future odometry of the ego-vehicle,and a two-stream RNN encoder-decoder incorporating predicted ego-motion into future object bounding box predictions. This model can be easily learned from massive collections of dashboard-mounted video of normal driving, and no manual labeling is required thanks to well-performing off-the-shelf object detectors and trackers.

Existing unsupervised VAD methods [8], [11], [13], [16], [17], [18] compute prediction error with respect to ground truth as the anomaly score, which may cause problems in our case due to imperfect object detection and tracking. To address this issue, we propose two alternatives. First, we take object boxes as the foreground to generate binary foreground-background masks and compute the IoU between predicted and ground truth masks as anomaly scores to reduce the impact of imperfect tracking. Second, we predict an object's location at time t based on the observed data from multiple previous time steps (e.g., $t-1$, $t-2$, *etc.*), collect all these predicted boxes of an object, and compute their consistency as the anomaly score. Such a prediction consistency metric does not compare prediction with detected objects, and therefore reduces the influence of imperfect detection and tracking. We compare with prediction accuracy methods and show the effectiveness of prediction consistency in traffic VAD experiments.

This paper also introduces a new large-scale benchmark dataset for traffic VAD called Detection of Traffic Anomaly (DoTA). DoTA contains 4,677 videos with 18 anomaly categories [19] and multiple anomaly participants in different driving scenarios. DoTA provides rich annotation for each anomaly: type (category), temporal annotation, and anomalous object bounding box tracklets. Current anomaly datasets contain only temporal annotations, so they cannot be used to evaluate the accuracy of spatial localization — where in the frame an anomaly is occurring. However, accurately locating the anomalous region is essential for model explainability and downstream applications such as collision avoidance. Taking advantage of this large-scale dataset with rich anomalous object annotations, we propose a novel VAD evaluation metric called Spatio-temporal Area Under Curve (STAUC) as a complement to frame-level AUC metrics. While AUC uses a per-frame anomaly score which is usually averaged from a pixel-level or object-level score map, STAUC takes the score map and computes its overlap with the annotated anomalous region. This overlap ratio is used as a weighting factor for true positive predictions with STAUC such that AUC is an upper bound on STAUC. Our STAUC can be considered as a combination of the mean average-precision (mAP) metric used in STAR or semantic segmentation evaluation and the traditional AUC metric used in VAD evaluation. STAUC has the advantage of mAP by computing the spatial overlap between prediction and ground truth, and it also has the advantage of AUC so that it is feasible for VAD evaluation tasks where the dataset is highly imbalanced due to the rareness of anomalous events. We benchmark existing VAD baselines and state-of-the-art methods on DoTA using both AUC and STAUC, and show the advantage of using STAUC.

The DoTA dataset can also be used for video action recognition and online action detection given its categorical annotations. Video action recognition takes a video clip as input to predict its anomaly type, e.g., oncoming collision or out-of-control vehicle, while online action detection processes a video frame-by-frame to classify each frame as normal or one type of anomaly. We provide benchmarks of state-of-the-art methods such as SlowFast [20] and TRN [21] on these two tasks. Experiments show that applying generalized video action recognition and online action detection methods to traffic anomaly understanding is far from perfect, motivating more research in this area.

This paper offers four contributions. First, we propose a future object localization-based unsupervised traffic video anomaly detection method and two anomaly score computation methods to address problems caused by imperfect object detection and tracking. Second, we introduce DoTA, a large-scale egocentric traffic video dataset which, to the best of our knowledge, is the largest traffic video anomaly dataset to date and the first containing detailed temporal, spatial, and categorical annotations. Third, we identify problems with the commonly-used AUC metric and propose a new spatio-temporal evaluation metric (STAUC) to address them. We benchmark state-of-the-art VAD methods with both AUC and STAUC and show the advantages of our new metric. Finally, we provide benchmarks of state-of-the-art video action recognition and online action detection algorithms on DoTA, which we hope will encourage further research on challenging egocentric traffic video scenarios.

This paper is based on a preliminary version that was published at the International Conference on Intelligent Robots and Systems (IROS) [22]. We extend the previous version by: 1) Providing a large-scale traffic VAD dataset, called DoTA, with temporal, spatial, and categorical annotations; 2) Introducing a new spatio-temporal evaluation metric for VAD, called STAUC, with a number of advantages over existing metrics; 3) Providing benchmarks of state-of-the-art VAD, video action recognition, and online action detection methods on the DoTA dataset; and 4) Introducing FOL-Ensemble, a new and strong baseline for traffic VAD.

2 Related Work

Existing Video Anomaly Detection (VAD) datasets. Are typically from surveillance cameras. For example, UCSD Ped1/Ped2 [23], CUHK Avenue [11], and ShanghaiTech [8] were collected from campus surveillance cameras and include anomalies like prohibited objects and abnormal movements, while UCF-Crime [24] includes accidents, robbery, and theft. Anomaly detection in egocentric traffic videos has very recently attracted attention. Chan *et al.* [1] propose the StreetAccident dataset of on-road accidents with 620 video clips

Authorized licensed use limited to: Sungkyunkwan University. Downloaded on June 21,2023 at 11:45:09 UTC from IEEE Xplore. Restrictions apply.

collected from dashboard cameras. The last ten frames of each clip are annotated as anomalous. Yao *et al.* [22] propose the A3D dataset containing 1,500 anomalous videos in which abnormal events are annotated with start and end times. Fang *et al.* [3] introduce the DADA dataset for driver attention prediction in accidents, while Herzig *et al.* [2] extract a collision dataset with 803 videos from BDD100K [25]. In very recent work, conducted contemporaneously to ours, Bao *et al.* [26] collected a car crash dataset with 1,500 videos for traffic accident anticipation which contains environmental attributes and cause-of-accident annotations. Our DoTA dataset is much larger (4,677 videos) and, more importantly, contains richer annotations that support traffic video anomaly analysis from spatial (location of anomalies), temporal (start and end of anomalies), and categorical (type of anomalies) perspectives.

Existing VAD models. Mainly focus on detecting the start and end of anomalous events and only implicitly relate to spatial localization. Hasan *et al.* [12] propose a convolutional Auto-Encoder to model the normality of video frames by reconstructing stacked input frames. A Convolutional LSTM Auto-Encoder is used in [16], [17], [18] to capture regular visual and motion patterns. Luo *et al.* [13] propose a stacked RNN for temporally-coherent sparse coding. Liu *et al.* [8] detect anomalies by looking for differences between predicted future frames and actual observations. Gong *et al.* [27] propose a MemAE network to query pre-saved memory units for reconstruction, while Wang *et al.* [28] design generalized one-class sub-spaces for discriminative regularity modeling. Other work has recently studied object-centric approaches. Ionescu *et al.* [15] cluster object features and train multiple support vector machine (SVM) classifiers using the confidence as an anomaly score. Morais *et al.* [14] model human skeleton regularity with local-global autoencoders and compute per-object anomaly scores. Although these methods have achieved promising results on VAD tasks in surveillance cameras, egocentric traffic scenarios are challenging due to the moving camera, dynamic foreground and background, and complex scenes. Instead of modeling the whole scene, we propose to predict future locations of traffic participants and compute prediction consistency as an anomaly score. We benchmark our method and state-of-the-art VAD methods on our new DoTA dataset.

Trajectory Prediction. Has been extensively investigated in computer vision research, and is often posed as a sequence-to-sequence generation problem. In this paper, we propose a VAD method based on trajectory prediction. Much work in trajectory prediction has focused on social interaction modeling [29], [30], multimodal prediction [31], [32], [33], [34], [35], [36], and goal estimation [37], [38], [39]. However, these methods are designed for third-person views from static cameras, whereas trajectory prediction in first-person, egocentric videos in which the camera itself is moving (e.g., a dashboard camera) can be even more challenging. Yagi *et al.* [40] incorporate different kinds of cues into a convolution-deconvolution network to predict pedestrians' future locations in first-person video. Yao *et al.* [41] extend this work to autonomous driving scenarios by proposing a multi-stream RNN encoder-decoder architecture with both past vehicle locations and image features as inputs. Malla *et al.* [42] and Rasouli *et al.* [43] use a similar structure and introduce intention and action priors to boost trajectory prediction accuracy. We propose a VAD method for driving videos based on [41] and introduce a prediction consistency metric to improve anomaly detection performance.

Action Understanding. Techniques can be applied to traffic anomaly classification after an anomaly is detected (e.g., to distinguish front-collision from turning-collision or vehicle-pedestrian collision, *etc.*). Two-stream networks [44] and temporal segment networks (TSN) [45] leverage RGB and optical flow data. Tran *et al.* [46] proposed 3D convolutional networks (C3D) for spatiotemporal modeling, followed by an inflated model [47]. Recent work substitutes 3D convolution with 2D and 1D convolution blocks (R(2+1)D [48]) to improve effectiveness and efficiency. Feichtenhofer *et al.* [20] propose the SlowFast model to extract video features from low and high frame rate streams. For real-time applications in untrimmed, streaming videos, online action detectors have been developed to classify video frames with only past observations [21], [49], [50], [51], [52], [53], [54], [55], [56]. In this paper we benchmark video action recognition (VAR) and online action detection (OAD) methods on our new DoTA dataset to show how anomaly events can be classified after being detected.

Recently, spatio-temporal action recognition (STAR) has been proposed to detect objects and action types simultaneously [4], [5], [6], [7], [57], [58] and has the potential to be applied to driving videos. However, STAR requires detailed annotations of each bounding box and action type during training and only detects pre-defined categories; due to the long-tailed distribution of traffic events, it is nearly impossible to pre-define all the anomalous event categories and to collect enough data for training [10], which makes it difficult to apply STAR to anomaly detection. In contrast, VAD does not require pre-defined anomaly categories because it models normality in the videos without supervision, and then identifies events that do not follow typical patterns as anomalies.

Video Object Detection (VOD). [59] is similar to detecting objects in static images, but incorporates temporal constraints and tries to handle the unique challenges of video such as motion blur. Early methods solved this problem by detecting objects in each individual frame. To capture the spatial-temporal nature of video and to increase the accuracy and efficiency of detection, recent methods [60], [61], [62], [63], [64], [65], [66] use deep learning. Some methods [67], [68] use Convolutional Long Short Term Memories (LSTMs) [69] to model long-term relations and select important features. Liu *et al.* [67] combine an image-based object detector with a convolutional LSTM to add temporal information to input features, in order to detect objects in videos. They improve inference speed by using large and small feature extractors in [68]. Other methods [61], [62], [63], [64], [65], [66] use local feature aggregation for detection in a given frame. Inspired by humans' object localization ability across multiple time steps, Chen *et al.* [70] jointly consider global semantic information and key frame object localization, and propose the Memory Enhanced Global-Local Aggregation module, which achieves 85.4% on the ImageNet VID dataset. Han *et al.* [71] propose the Hierarchical Video Relation Network which models the Inter-Video Proposal Relation in addition to Intra-Video Proposal Relation.

Authorized licensed use limited to: Sungkyunkwan University. Downloaded on June 21,2023 at 11:45:09 UTC from IEEE Xplore. Restrictions apply.

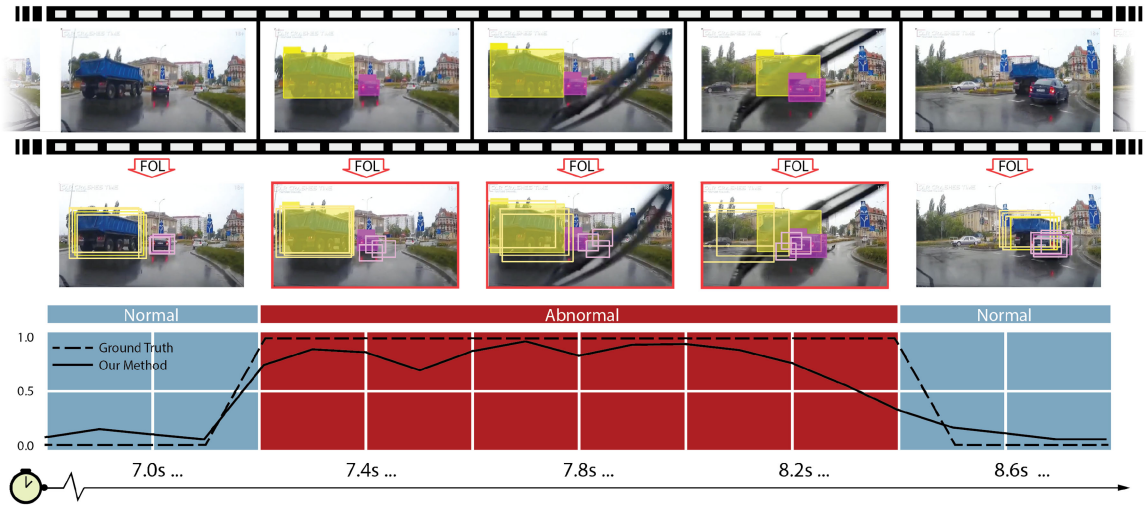


Fig. 1. Overview of our method based on future object localization (FOL) using a sample video from our DoTA dataset. Annotated bounding boxes (filled) and predicted boxes are presented. For each time step, we collect FOL predictions of all traffic participants from different past time steps and compute the bounding box standard deviation, called consistency, as the anomaly score.

Although Video Object Detection is related to the VAD task we consider here because they both involve spatial-temporal modeling, our work is significantly different in several ways: 1) VAD aims at localizing anomalous events in video while VOD detects object bounding boxes and types; 2) our DoTA dataset focuses on the richness of anomalous behaviors rather than only objects; 3) our DoTA dataset provides labels for anomalous behaviors, not normal objects and their behaviors.

3 Method

Autonomous vehicles must monitor the roadway for signs of unexpected activity that may require evasive action. A natural way to detect these anomalies is to look for unexpected or rare movements in the first-person perspective of a front-facing, dashboard-mounted camera on a moving ego-vehicle. Prior work [8] proposes monitoring for unexpected scenarios by using past video frames to predict the current video frame, and then looking for major differences. However, this does not work well for moving cameras on vehicles, where the perceived optical motion in the frame is induced by both moving objects and camera ego-motion. More importantly, anomaly detection systems do *not* need to accurately predict all information in the frame, since anomalies are unlikely to involve peripheral objects such as houses or billboards by the roadside. This paper thus assumes that an anomaly may exist if an object's real-world observed trajectory deviates from the predicted trajectory. For example, when a vehicle should move through an intersection but instead suddenly stops, a collision may have occurred.

Our model is trained with a large-scale dataset of normal, non-anomalous driving videos. This allows the model to learn normal patterns of object and ego motions, then recognize deviations without the need to explicitly train the model with examples of every possible anomaly. This video dataset is easy to obtain and does not require hand labeling since the object trajectories can be estimated using off-the-shelf object detection and tracking algorithms [41]. Considering the influence of ego-motion on perceived object location, we incorporate a future ego-motion prediction module as an additional input. At test time, we use the model to predict the current locations of objects based on the last few frames of data and determine if an abnormal event has happened based on three different anomaly detection strategies, described in Section 3.2.

3.1 Future Object Localization (FOL)

3.1.1 Bounding Box Prediction

Following [41], we denote an observed object's bounding box $X_t = [c_t^x, c_t^y, w_t, h_t]$ at time t, where (c_t^x, c_t^y) is the location of the center of the box and w_t and h_t are its width and height in pixels, respectively. We denote the object's future bounding box trajectory for the δ frames after time t to be $\mathbf{Y_t} = \{Y_{t+1}, Y_{t+2}, \ldots, Y_{t+\delta}\}$, where each Y_t is a bounding box parameterized by center, width, and height. Given the image evidence O_t observed at time t, a visible object's location X_t, and its corresponding historical information H_{t-1}, our future object localization model predicts $\mathbf{Y_t}$. This model is inspired by the multi-stream RNN encoder-decoder framework of Yao *et al.* [41], but with a completely different network structure [21] to allow for online training and inference. For each frame, [41] receives and re-processes the previous ten frames before making a decision, whereas our model only needs to process the current information, making it much faster at inference time.

Our model is shown in Fig. 2. Two encoders (Enc) based on gated recurrent units (GRUs) receive an object's current bounding box and pixel-level spatiotemporal features as inputs, respectively, and update the object's hidden states. In particular, the spatiotemporal features are extracted by a region-of-interest pooling (RoIPool) operation using bilinear interpolation from precomputed optical flow fields. The updated hidden states are used by a location decoder (Dec) to recurrently predict the bounding boxes of the immediate future. We train the FOL model using a mean squared error loss between the predicted and target future bounding boxes.

Authorized licensed use limited to: Sungkyunkwan University. Downloaded on June 21,2023 at 11:45:09 UTC from IEEE Xplore. Restrictions apply.

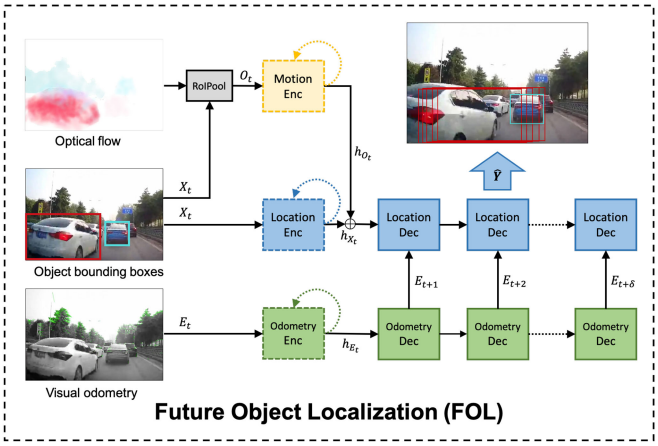


Fig. 2. Overview of the future object localization model. Blocks with dashed outlines are encoders, and solid lines are decoders. The decoder recurrences are unfolded to visualize the prediction horizon.

3.1.2 Ego-Motion Cue

Ego-motion information of the moving camera has been shown necessary for accurate future object localization [41], [72]. Let E_t be the ego-vehicle's pose at time t; $E_t = \{\phi_t, x_t, z_t\}$ where ϕ_t is the yaw angle and x_t and z_t are positions along the ground plane with respect to the vehicle's starting position in the first video frame. We predict the ego-vehicle's odometry by using another GRU encoder-decoder module to encode ego-position change vector $E_t - E_{t-1}$ and decode future ego-position changes $\mathbf{E} = \{\hat{E}_{t+1} - E_t, \hat{E}_{t+2} - E_t, \ldots, \hat{E}_{t+\delta} - E_t\}$. We use the change in ego-position to eliminate accumulated odometry errors. The output $\mathbf{E}$ is then combined with the hidden state of the future object localization decoder using average pooling to form the input into the next time step.

3.1.3 Missed Objects

We build a list of trackers $Trks$ per [73] to record the current bounding box $Trks[i].X_t$, the predicted future boxes $Trks[i].\hat{\mathbf{Y}}_t$, and the tracker age $Trks[i].age$ of each object. We denote all maintained track IDs as D (both observed and missed), all currently observed track IDs as C, and the missed object IDs as $D - C$. At each time step, we update the observed trackers and initialize a new tracker when a new object is detected. We use a temporarily-missing or occluded object's previously predicted bounding boxes to estimate current location, running future object localization with RoIPool features from predicted boxes (Algorithm 1). Missing object handling is essential in our prediction-based anomaly detection method to eliminate the impact of failed object detection or tracking in any given frame. For example, if an object with a normal motion pattern is missed for several frames, the FOL model is still expected to give reasonable predictions except for some accumulated deviations. On the other hand, if an anomalous object is missed during tracking [73], we make a prediction using its previously predicted bounding box whose region can be substantially displaced and can result in inaccurate predictions. In this case, some false alarms and false negatives can be eliminated by using the metrics presented in Section 3.2.3.

Algorithm 1. FOL-Track Algorithm

Input: Observed bounding boxes $\{X_t^{(i)}\}$ where $i \in C$, observed image evidence O_t, trackers of all objects $Trks$ with track IDs D
Output: Updated trackers $Trks$

A is the maximum age of a tracker
for $i \in C$ **do** // update observed trackers
  if $i \notin D$ **then**
    initialize $Trks[i]$
  else
    $Trks[i].X_t = X_t^{(i)}$
    $Trks[i].\hat{\mathbf{Y}}_t = FOL(X_t^{(i)}, O_t)$
  end
end
for $j \in D - C$ **do** // update missed trackers
  if $Trks[j].age > A$ **then**
    remove $Trks[j]$ from $Trks$
  else
    $Trks[j].X_t = Trks[j].\hat{Y}_{t-1}$
    $Trks[j].\hat{\mathbf{Y}}_t = FOL(Trks[j].X_t, O_t)$
  end
end

3.2 Traffic Anomaly Detection using FOL

Unsupervised anomaly detection methods compute anomaly scores based on prediction or reconstruction accuracy [8], [12], [14], [15]. In this section, we first present the basic anomaly metric computed from predicted bounding box accuracy. The key idea is that object trajectories and locations in non-anomalous events can be precisely predicted, while deviations from predicted behaviors suggest an anomaly. Next we propose two different strategies to compute anomaly scores using: 1) the foreground-background mask generated from predictions and 2) the prediction consistency.

3.2.1 Predicted Bounding Box Accuracy

One simple method for recognizing abnormal events is to directly measure the similarity between predicted object bounding boxes and their corresponding observations. The FOL model predicts bounding boxes of the next δ future frames, i.e., at each time t each object has a bounding box predicted at each time from $t - \delta$ to $t - 1$. We first average the positions of the δ bounding boxes, then compute intersection over union (IoU) between the averaged bounding box and the observed box location, where higher IoU means greater agreement between the two boxes. We average computed IoU values over all observed objects, and then compute an aggregate anomaly score $L_{bbox} \in [0, 1]$,

$$L_{bbox} = 1 - \frac{1}{N}\sum_{i=1}^{N} \mathbf{IoU}\left(\left(\frac{1}{\delta}\sum_{j=1}^{\delta} \hat{Y}_{t,t-j}^{i}\right), Y_{t_0}^{i}\right), \tag{1}$$

where N is the total number of observed objects, and $\hat{Y}_{t,t-j}^{i}$ is the predicted bounding box from time $t - j$ of object i at time t. This method, which we call FOL-IoU, relies upon accurate object tracking to match predicted and observed bounding boxes.

Authorized licensed use limited to: Sungkyunkwan University. Downloaded on June 21,2023 at 11:45:09 UTC from IEEE Xplore. Restrictions apply.

3.2.2 Predicted Box Mask Accuracy

Although tracking algorithms such as Deep-SORT [73] offer reasonable accuracy, it is still possible to lose or mis-track objects. We found that inaccurate tracking particularly happens in severe traffic accidents because of the twist and distortion of object appearances. Moreover, severe ego-motion also results in inaccurate tracking due to sudden changes in object locations. This increases the number of false negatives of the metric proposed above, which simply ignores objects that are not successfully tracked in a given frame. To solve this problem, we first convert all areas within the predicted bounding boxes to binary masks, with areas inside the boxes having value 1 and backgrounds having 0, and do the same with the observed boxes. We then calculate an anomaly score as the IoU between these two binary masks,

$$I^{(u,v)} = \begin{cases} 1, \text{ if pixel (u,v) within box Xi, } \forall \text{i}, \\ 0, \qquad\qquad \text{otherwise}, \end{cases} \tag{2}$$

$$L_{mask} = 1 - \mathbf{IoU}\big(\hat{I}_{t,t-1}, I_t\big), \tag{3}$$

where $I^{(u,v)}$ is pixel (u, v) on mask I, X^i is the ith bounding box, $\hat{I}_{t,t-1}$ is the predicted mask from time $t-1$, and I_t is the observed mask at t. In other words, while the bounding box accuracy metric compares bounding boxes on an object-by-object basis, this metric simply compares the bounding boxes of all objects simultaneously. The main idea is that accurate prediction results will still have a relatively large IoU compared to the ground truth observation. We denote the mask accuracy-based method FOL-Mask.

3.2.3 Predicted Bounding Box Consistency

The above methods rely on accurate detection of objects in concurrent frames to compute anomaly scores. However, the detection of anomaly participants is not always accurate due to changes in appearance and mutual occlusions. We hypothesize that visual and motion features related to an anomaly do not only appear once it happens, but are usually accompanied by a salient pre-event. We thus propose another strategy, called FOL-STD, to detect anomalies by computing consistency of future object localization outputs from several previous frames while eliminating the effect of inaccurate detection and tracking.

As discussed in Section 3.2.1, for each object in video frame at time t, we can collect δ bounding boxes predicted from time $t-1, t-2, .., t-\delta$. We compute the standard deviation (STD) between all δ predicted bounding boxes to measure their similarity,

$$L_{pred} = \frac{1}{N}\sum_{i=1}^{N} \max_{\{c^x, c^y, w, h\}} \mathbf{STD}([\hat{Y}^{(i)}_{t,t-j}]_{j=1}^{j=\delta}). \tag{4}$$

where $\hat{Y}^{(i)}_{t,t-j}$ is the bounding box of the ith object in frame at time t predicted from the frame at time $t-j$, and c^x, c^y, w, h are the center coordinates and the width and height of a bounding box. We compute the maximum standard deviation over the four components of the bounding boxes since different anomalies may be indicated by different effects on the bounding box, e.g., suddenly stopped cross traffic may only have large variance along the horizontal axis. A low standard deviation suggests the object is following normal movement patterns thus the predictions are stable, while a high standard deviation suggestions abnormal motion. For all three methods, we follow [8] to normalize computed anomaly scores for evaluation.

3.3 Frame-Object Ensemble Anomaly Detection

Our FOL based methods are object-centric by encoding-decoding object information. Frame-level VAD methods focus on appearance while object-centric methods focus more on object motion. We are not aware of any method combining the two. Appearance-only methods may fail with large changes in lighting conditions, and motion-only methods may fail when trajectory prediction is imperfect. We combine FOL-STD with the frame prediction method AnoPred [8], yielding what we call the FOL-Ensemble method. AnoPred predicts one anomaly score per image pixel while our method predicts one anomaly score per object. We first map our object anomaly score to a per-pixel score by putting a Gaussian function at the center of each object (as introduced in Section 5). We train each module of FOL-Ensemble independently and apply average pooling on the computed per-pixel scores from two modules to compute a final anomaly score. We observed this late fusion is better than fusing hidden features at an early stage and training the two models together, since their hidden features are scaled differently. AnoPred encodes one feature per frame, while FOL-STD has one feature per object.

4 Detection of Traffic Anomaly (DoTA) Dataset

We introduce DoTA, the first publicly-available traffic video anomaly dataset with temporal, spatial, and categorical annotations. To build DoTA, we collected more than 6,000 video clips mainly from two YouTube channels[1] which provides traffic accident videos for driver education purposes. We selected diverse dash camera accident videos from different areas (e.g., East Asia, North America, Europe etc.) under different weather (e.g., sunny, cloudy, raining, snowing, etc.) and lighting conditions (day and night). We avoided videos with accidents that were not visible or where the camera dislodged during the accident, yielding 4,677 videos with 1280×720 resolution, each containing exactly one anomalous event. Although the original videos are at 30 fps, we extracted frames at 10 fps for annotations and experiments in this paper. Table 1 compares DoTA with other ego-centric traffic anomaly datasets. We annotated the dataset using a custom tool based on Scalabel.[2] Labeling traffic anomalies is subjective, especially for properties like start and end times. To produce high quality annotations, each video was labeled by three annotators, and the temporal and spatial (categorical) annotations were merged by taking average (mode) to minimize individual biases.

Temporal Annotations. Each DoTA video is annotated with anomaly start and end times, which separates it into three temporal partitions: precursor, which is normal video preceding the anomaly, the anomaly window, and post-anomaly, which is normal activity following the anomaly. Duration distributions are shown in Fig. 5a. Since early

1. https://youtube.com/user/CarCrashesTime and https://youtube.com/channel/UC-Oa3wml6F3YcptlFwaLgDA
2. https://scalabel.ai/

Authorized licensed use limited to: Sungkyunkwan University. Downloaded on June 21,2023 at 11:45:09 UTC from IEEE Xplore. Restrictions apply.

TABLE 1
Comparison of Published Driving Video Anomaly Datasets

Dataset	type	# anomaly videos	# frames		Annotations
UCSD Ped1/2 [74]	Surveillance	98	18560	(30fps)	temporal
CUHK Avenue [11]		37	30,652	(30fps)	temporal
UCF-Crime [24]		1,900	13,769,300	(30fps)	temporal
ShanghaiTech [13]		437	317,398	(30fps)	temporal
StreetAccident [1]	Dashcam	620	62,000	(20fps)	temporal
A3D [22]		1,500	128,175	(10fps)	temporal
CCD [26]		1,500	75,000	(10fps)	temporal, spatial (tracklets), causation
DADA [3]		2,000	648,476	(30fps)	temporal, spatial (eye-gaze)
DoTA		**4,677**	**731,932**	(10fps)	temporal, spatial (tracklets), categories

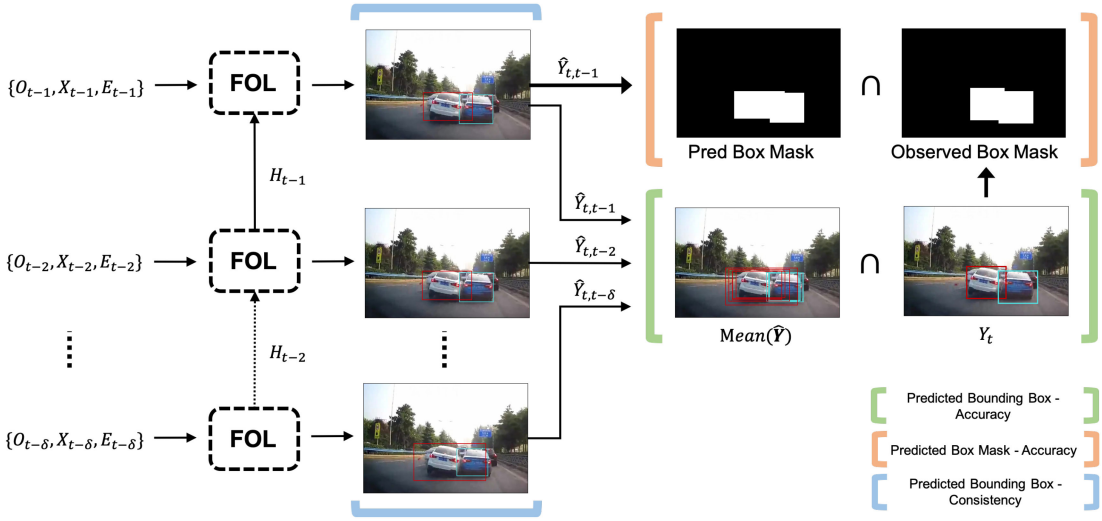


Fig. 3. Overview of our unsupervised VAD methods. The three brackets correspond to: (1) Predicted bounding box accuracy method (green); (2) Predicted box mask accuracy method (orange); (3) Predicted bounding box consistency method (blue). All methods use multiple previous FOL outputs to compute anomaly scores.

detection is essential for on-road anomalies [1], [75], we asked the annotators to estimate the anomaly start as the time when the anomaly was inevitable. The anomaly end was approximated as the time when all anomalous objects are out of the field of view or are stationary. Our annotation is different from [3] where a frame is marked as an anomaly start if half of an anomaly participant appears in the camera view; such a start time can be too early because anomaly participants often appear for a while before they start to behave abnormally. Our annotation is also distinct from [1] and [22] where the anomaly start is marked when a crash happens, which does not support early detection.

Spatial Annotations. DoTA is the first traffic anomaly dataset to provide detailed spatio-temporal annotation of anomalous objects. Each anomaly participant is assigned a unique track ID, and each participant's bounding box is labeled from anomaly start to anomaly end or until the object is out of view. We consider seven common traffic participant categories: person (pedestrian), car, truck, bus, motorcycle, bicycle, and rider, following the BDD100K style [25]. Statistics of object categories and per-video anomalous object numbers are shown in Figs. 5b and 5c. DADA [3] also provides spatial annotations by capturing video observers' eye-gaze for driver attention studies. However, they have shown that eye-gaze does not always coincide with the anomalous region, and that gaze can have a 1-2 second delay from anomaly start. Our tracklets thus provide improved annotation for spatio-temporal anomaly detection studies.

Anomaly Categories. Each DoTA video is assigned one of the 9 categories listed in Table 2 as defined in [19]. We have observed that even within the same anomaly category, different viewpoints cause large visual variation (e.g., whether the ego-vehicle is a participant in the accident or if it is simply observing it), as shown in Fig. 4. Therefore we split each category into ego-involved and non-ego (marked with *), resulting in 18 categories total. Sometimes the category can be ambiguous, particularly when one anomaly is followed by another. For example, an oncoming out-of-control (OO*) vehicle might result in an oncoming collision (OC) with the ego vehicle. In such cases, we annotate the anomaly category as the dominant one in the video, i.e, the one that lasts longer during the anomaly period. The distribution of videos in each category is shown in Fig. 5b.

5 A New VAD Evaluation Metric

5.1 Critique of Current VAD Evaluation

Most VAD methods compute an anomaly score for each frame, and evaluate by plotting receiver operating characteristic (ROC) curves using temporally concatenated scores and

Authorized licensed use limited to: Sungkyunkwan University. Downloaded on June 21,2023 at 11:45:09 UTC from IEEE Xplore. Restrictions apply.

TABLE 2
Traffic Anomaly Categories in the DoTA Dataset

Label	Anomaly Category
ST	Collision with another vehicle that starts, stops, or is stationary
AH	Collision with another vehicle moving ahead or waiting
LA	Collision with another vehicle moving laterally in same direction
OC	Collision with another oncoming vehicle
TC	Collision with another vehicle that turns into or crosses a road
VP	Collision between vehicle and pedestrian
VO	Collision with an obstacle in the roadway
OO	Out-of-control and leaving the roadway to the left or right
UK	Unknown

computing area under curve (AUC). AUC measures how well a VAD method locates an anomaly along the temporal axis but ignores accuracy on spatial axes since an averaged anomaly score lacks spatial information. Frame-level AUC does not evaluate the performance of spatial localization [76], while anomaly region localization is necessary in VAD. We argue AUC is insufficient to fully evaluate VAD performance.

In computing AUC, a true positive occurs when the model predicts a high anomaly score for a positive frame. Fig. 6 shows two positive frames and their corresponding score maps computed by the four benchmarked VAD methods. Although the maps are different, the anomaly scores averaged from these maps are similar, meaning they are treated similarly in AUC evaluation. This results in similar AUCs among all methods, which leads to a conclusion that all perform similarly. However, AnoPred (Fig. 6b) predicts high scores for trees and other noise, while AnoPred+Mask and FOL-STD (Figs. 6c and 6d) predict high scores for unrelated vehicles. Ensemble (Fig. 6d) alleviates these problems but still has high anomaly scores outside the labeled anomalous regions. (Note that FOL-STD and Ensemble are pseudo-maps introduced in Section 5.2.) Although these methods yield similar AUCs, VAD methods should be distinguished by their abilities to localize anomalous regions. Anomalous spatial localization is essential because it improves reaction to anomalies — permitting collision avoidance, for example — and allows for model explainability. This motivates a new metric to evaluate how well a model detects both the temporal and spatial location of the anomaly.

5.2 Spatial-Temporal Area Under Curve (STAUC) Metric

For each positive frame, we first calculate the true anomalous region rate (TARR), which is a scalar describing how much of the anomaly score is located within the true anomalous region,

$$\mathrm{TARR}_t = \frac{\sum_{i \in m_t} \Delta I(i)}{\sum_{i \in M} \Delta I(i)}, \tag{5}$$

where $\Delta I(i)$ is the anomaly score at pixel i, M represents all frame pixels, and m_t is the annotated anomalous frame region (i.e., the union of all annotated bounding boxes) at time t. TARR is inspired by anomaly segmentation tasks where the overlap between prediction and annotation is computed [77]. Another metric related to TARR is intersection over union (IoU) used in object detection, where the predicted box has IoU equal to 1.0 if it is the same as the labeled box. However, in contrast to object detection, not all pixels in an anomalous event region are necessarily anomalous: the tree pixels or street surface pixels in the car bounding boxes in Fig. 6a, for example. The proposed TARR value measures how well a model concentrates on the true anomalous region but does not require it to predict high anomaly scores for all pixels within the true anomalous region, thus making it more appropriate to the anomalous event detection task.

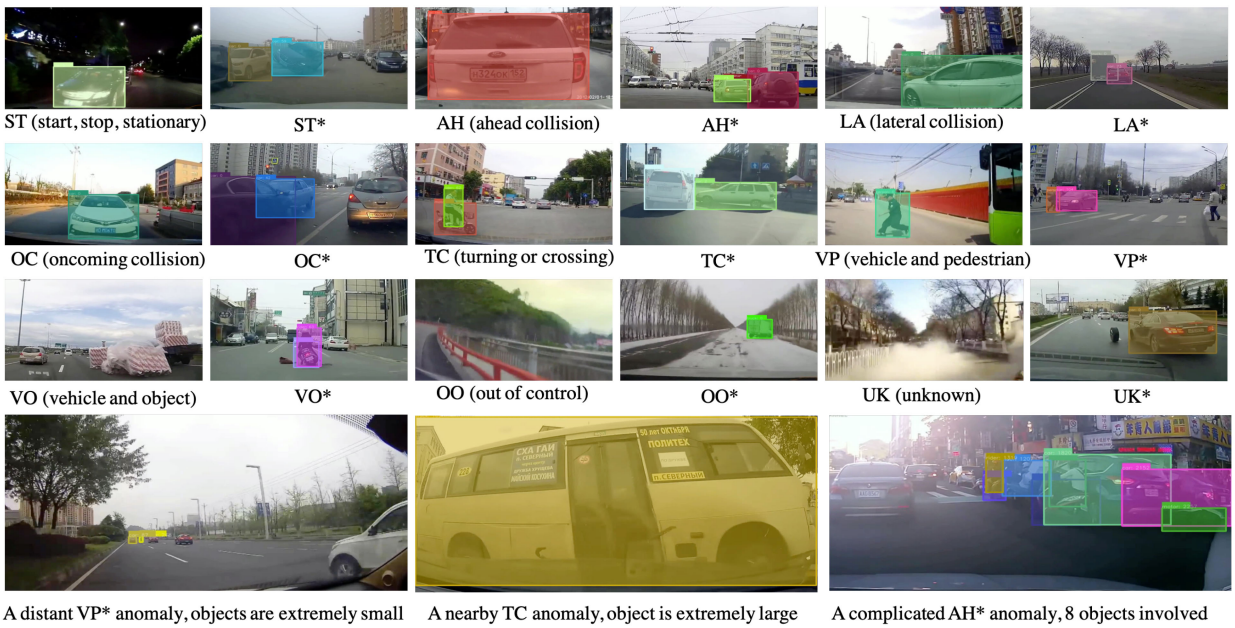
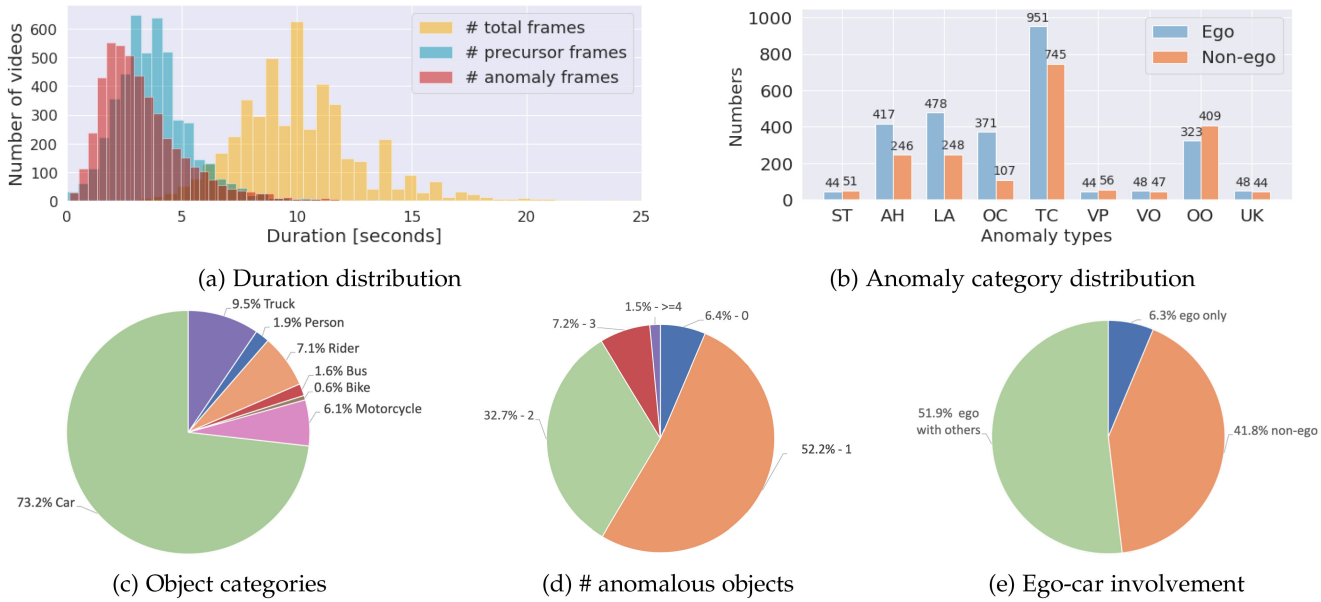


Fig. 4. DoTA Samples. Spatial annotations are shown as shadowed bounding boxes. Short anomaly category labels with * indicate non-ego anomalies.

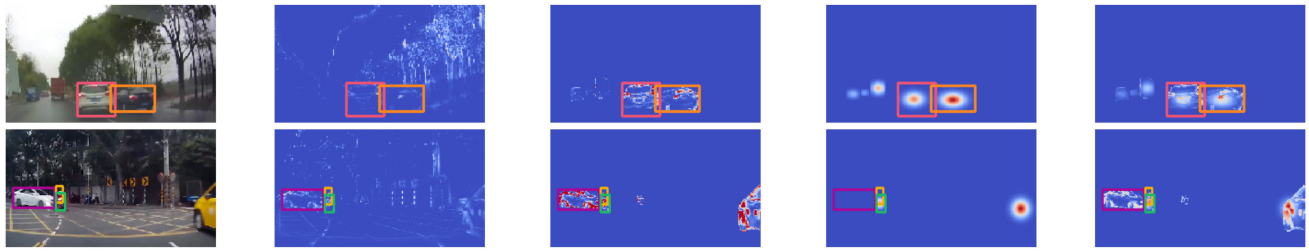
Authorized licensed use limited to: Sungkyunkwan University. Downloaded on June 21,2023 at 11:45:09 UTC from IEEE Xplore. Restrictions apply.



(a) Duration distribution (b) Anomaly category distribution

(c) Object categories (d) # anomalous objects (e) Ego-car involvement

Fig. 5. DoTA dataset statistics.



(a) GT image (b) AnoPred (c) AnoPred+Mask (d) FOL-STD (e) FOL-Ensemble

Fig. 6. Anomaly score maps computed by four methods. Ground truth anomalous regions are labeled by bounding boxes. Brighter color indicates higher score.

Next, we calculate the spatio-temporal true positive rate (STTPR),

$$\text{STTPR} = \frac{\sum_{t \in TP} \text{TARR}_t}{|P|}, \tag{6}$$

where TP represents all true positive predictions and P represents all ground truth positive frames. STTPR is a true positive rate where each true positive is weighted by its TARR. We then use STTPR and the false positive rate to plot an ROC curve (which we call Spatial-Temporal ROC or STROC) and calculate the area under the curve, which gives the STAUC. Note that STAUC $\leq$ AUC; the two are equal in the best case where $\text{TARR}_t = 1 \; \forall t$.

Object-centric VAD [14], [15], [22] computes per-object anomaly scores s_k instead of an anomaly score map ΔI. To generalize the STAUC metric to object-centric methods, we first create pseudo-anomaly score maps as illustrated in Fig. 6d. Each object has a 2D Gaussian distribution centered in its bounding box. The pixel score is then computed as the sum of the scores calculated from all boxes it occupies,

$$\Delta I_{pseudo}(i) = \sum_{\forall k, i \in B_k} s_k \, e^{-\frac{|i_x - x_k|^2}{2w_k} - \frac{|i_y - y_k|^2}{2h_k}}, \tag{7}$$

where i_x and i_y are coordinates of pixel i and $[x_k, y_k, w_k, h_k]$ are center location, width, and height of object bounding box B_k. For the Ensemble method, we take the average of ΔI and ΔI_{pseudo} as the anomaly score map in Fig. 6e. This map is used as ΔI in Eq. (5) to compute TARR and STAUC.

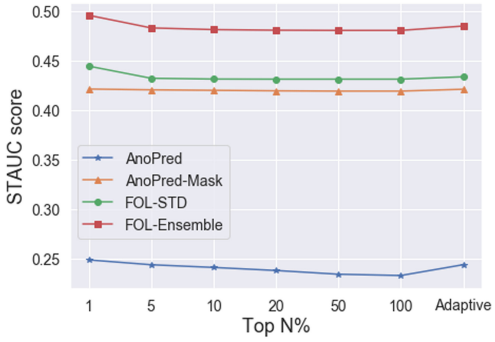
TARR is not robust to anomalous region size m_t. When $m_t \ll M$, TARR could be small even though all anomaly scores are high in m_t. We thus propose selecting the top $N\%$ of pixels with the largest anomaly scores as candidates, and compute TARR from these candidates instead of all pixels. An extremely small N such as 0.01 may result in a biased candidate set dominated by false or true detections such that $TARR = 0$ or 1. To address this issue, we compute an adaptive N for each frame based on the size of its annotated anomalous region,

$$N_{adaptive} = \frac{\text{number of pixels in anomalous region}}{\text{Total number of pixels}} \times 100. \tag{8}$$

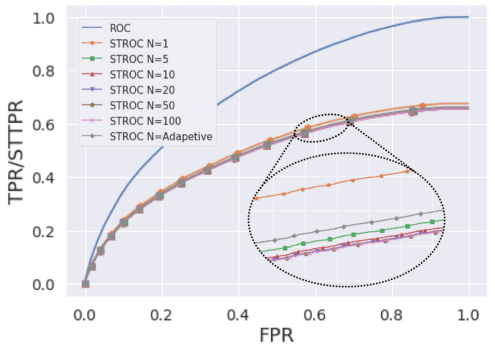
The average $N_{adaptive}$ of the test data in DoTA is 11.12 with a standard deviation 13.09. The minimum and maximum $N_{adaptive}$ values are 0.005 and 95.8, showing extreme cases where the anomalous object is very small (far away) or large (nearby).

A critical consideration for any new metric is its robustness to hyper parameters. We have tested STAUC with $N = [1, 5, 10, 20, 50, 100, N_{adaptive}]$ for different VAD methods. As shown in Fig. 7a, STAUC slightly decreases with increasing N but stabilizes when N is large, indicating that STAUC is relatively robust. Fig. 7b shows that STROC curves with

Authorized licensed use limited to: Sungkyunkwan University. Downloaded on June 21,2023 at 11:45:09 UTC from IEEE Xplore. Restrictions apply.



(a)



(b)

Fig. 7. (a) STAUC values of different methods using different top $N\%$; (b) ROC curve and STROC curves of the Ensemble method with different top $N\%$.

different N are close, especially when $N \geq 5$, and their upper bound is the traditional ROC. $N_{adaptive}$ is selected for our benchmarks based on each frame's annotation and its corresponding mid-range STAUC value.

5.3 Comparison With Spatio-Temporal Action Recognition Metrics

Two existing evaluation metrics related to our STAUC are the supervoxel-AUC [78], [79] and mean average precision (mAP) [4] used for spatio-temporal action recognition (STAR).

Supervoxel-AUC. Uses an overlap threshold between predicted and ground truth supervoxels to determine whether a supervoxel is considered positive or negative. The authors report multiple AUC values at different overlap thresholds (e.g., 10%, 20%, etc.). However, most VAD methods predict pixel-level anomaly score maps rather than supervoxels, therefore supervoxel-AUC cannot be directly applied to VAD evaluation. Our STAUC, on the other hand, uses TARR as a measure of how positive a frame is so that we do not need to evaluate STAUC under different thresholds, making STAUC more universal than supervoxel-AUC. Another advantage of STAUC is that it can evaluate different output and ground truth formats: STAUC can be applied to pixel-level, object-level and, essentially, supervoxel-level output since it computes how well anomaly scores locate within the ground truth region regardless of the region format.

Mean Average Precision (mAP). Has been used for evaluating object-level action recognition. mAP computes the IoU between detected and ground truth bounding boxes and uses a fixed IoU threshold (e.g., 0.5 in [4]) to categorize into positives and negatives. The AP is then computed for each action type and the mean value is reported. However, the precision-recall curve used to compute AP can perform differently for balanced versus unbalanced data distributions, which makes mAP less appropriate for VAD tasks where anomalous events are rare compared to normal events. Our STAUC can be considered as a combination of the mAP idea and the AUC metric so that it evaluates spatial accuracy and can be applied to imbalanced data.

Compared to supervoxel-AUC and mAP, our STAUC is specifically designed to extend VAD evaluation from temporal to spatio-temporal.

6 Experiments

We benchmarked VAD baselines and our methods on the new DoTA dataset. DoTA also provides categorical annotations to suit video action recognition (VAR) and online action detection tasks, thus we provide extra benchmarks for state-of-the-art methods for these two tasks using the DoTA dataset. We randomly partitioned DoTA into 3,275 training and 1,402 test videos and use these splits for all tasks. Unsupervised VAD models must be trained only with non-anomalous data, so we use the precursor frames from each video for training. VAR and online action detection models are fully-supervised and thus are trained using all training data.

6.1 Task 1: Video Anomaly Detection (VAD)

6.1.1 Benchmarked Methods

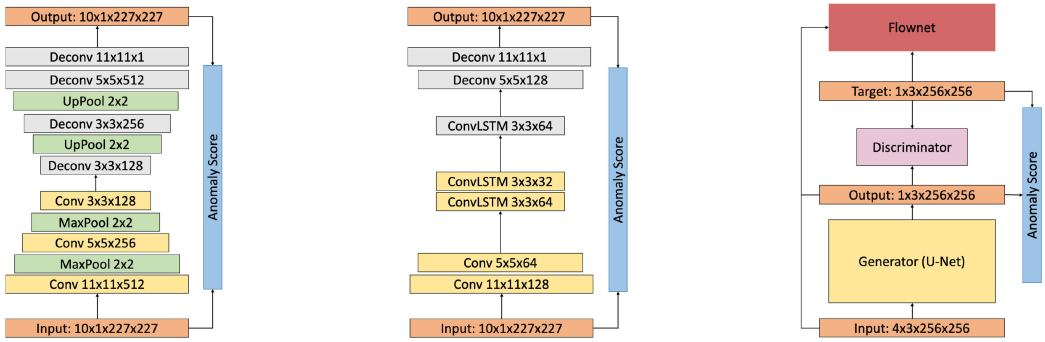
We benchmark three frame-level VAD method, ConvAE [12], ConvLSTMAE [17], and AnoPred [8] and their variants as baselines. Frame-level methods detect anomalies by either reconstructing past frames or predicting future frames, and computing the reconstruction or prediction error as the anomaly score.

ConvAE. [12] is a spatio-temporal autoencoder model which encodes temporally stacked images with 2D convolutional encoders, and decodes with deconvolutional layers to reconstruct the input (Fig. 8a). The per-pixel reconstruction error forms an anomaly score map ΔI, and mean squared error (MSE) is computed as a frame-level anomaly score. To further compare the effectiveness of image and motion features, we implemented *ConvAE(gray)* and *ConvAE(flow)* to reconstruct the grayscale image and dense optical flow, respectively. The input to ConvAE(flow) is a stacked historical flow map with size $20 \times 227 \times 227$ acquired from pre-trained FlowNet2 [80]. We trained both variants using AdaGrad with learning rate 0.01 and batch size 24.

ConvLSTMAE. [17] is similar to ConvAE but models spatial and temporal features separately. A 2D CNN encoder first captures spatial information from each frame, then a multi-layer ConvLSTM recurrently encodes temporal features. Another 2D CNN decoder then reconstructs input video clips (Fig. 8b). We implemented *ConvLSTMAE(gray)* and *ConvLSTMAE(flow)*. We trained both variants using AdaGrad with learning rate 0.01 and batch size 24.

AnoPred. [8] is a frame-level VAD method that takes four contiguous previous RGB frames as input and applies UNet to predict a future RGB frame (Fig. 8c). AnoPred boosts prediction accuracy with a multi-task loss incorporating image intensity, optical flow, gradient, and adversarial losses. AnoPred was proposed for surveillance cameras. However, traffic videos are much more dynamic, making future frame

Authorized licensed use limited to: Sungkyunkwan University. Downloaded on June 21,2023 at 11:45:09 UTC from IEEE Xplore. Restrictions apply.



(a) ConvAE [12] (b) ConvLSTMAE [17] (c) AnoPred [8]

Fig. 8. Network architectures of methods for video anomaly detection.

prediction difficult. Therefore we also benchmarked a variant of AnoPred to focus on video foregrounds. We used Mask-RCNN [81] pre-trained on Cityscapes [82] to acquire object instance masks for each frame, and apply instance masks to input and target images, resulting in an *AnoPred+Mask* method that only predicts foreground objects and ignores noisy backgrounds such as trees and billboards. In contrast to [12], [17], AnoPred uses Peak Signal to Noise Ratio as the anomaly score with better results. Both variants are trained based on the original paper.

Implementation Details. We used the published implementations of ConvAE, ConvLSTMAE, and AnoPred and modified the input layer size to suit grayscale or optical flow input. All these models were trained according to the original papers. Our FOL-based methods use hidden size 128 for all GRU modules and are trained using the RMSprop [83] optimizer with batch size 16, learning rate 0.0001, and no weight decay. All models are trained and evaluated on NVidia Titan XP GPUs. For evaluation, we ignore videos with unknown category or without objects, resulting in 1,305 test videos.

6.1.2 Results

Overall Results. The top four rows of Table 3 show performance of ConvAE and ConvLSTMAE with grayscale or optical flow inputs. Generally, using optical flow achieves better AUC, indicating that motion is an informative feature for this task. However, all baselines achieve low STAUC, meaning that they cannot localize anomalous regions well. AnoPred achieves 67.5 AUC but only 24.4 STAUC, while AnoPred with masked RGB input (AnoPred+mask) has 2.7 lower AUC but 17.7 higher STAUC. By applying instance masks, the model focuses on foreground objects to avoid computing high scores for the background, resulting in slightly lower AUC but much higher STAUC. This supports our hypothesis that higher AUC does not always imply a better VAD model, while STAUC better captures the ability to localize anomalous regions.

Table 3 also shows evaluations of four variants of our methods: FOL-IoU (prediction accuracy), FOL-Mask (prediction mask accuracy), FOL-STD (prediction consistency), and FOL-Ensemble, where FOL-Ensemble is an ensemble model of FOL-STD and AnoPred+Mask. FOL-Mask outperforms FOL-IoU as it is more robust to inaccurate object tracking. FOL-STD outperforms FOL-IoU and FOL-Mask by a large margin, which shows the effectiveness of our proposed consistency metric over the existing accuracy based metric. Its higher STAUC also shows that FOL-STD is more robust to scenarios where objects are not accurately detected or tracked. FOL-STD outperforms AnoPred on both metrics by specifically focusing on object motion and location, both of which are important indicators of traffic anomalies. An ablation study for FOL-STD is also shown in Table 3 to evaluate the effectiveness of different modules in the FOL network. Although FOL-STD with bounding box input serves as a reasonable baseline, adding ego motion and optical flow streams further boosts the AUC and STAUC values, indicating the effectiveness of our multi-stream FOL network. Finally, the FOL-Ensemble method achieves the best AUC and STAUC among all methods, indicating that combining frame-level appearance and motion features is a direction worth investigating in future VAD research.

Per-class Results. Table 4 shows results of AnoPred, AnoPred+Mask, FOL-STD, and FOL-Ensemble broken out according to the type of anomaly. We observe that STAUC (unlike AUC) can break out results by anomaly type. For example, Ensemble has comparable AUCs on the ego-involved OC (oncoming collision) and VP (vehicle-pedestrian) anomalies (73.4 vs 70.1) but significantly different STAUCs (56.6 vs 35.2), showing that anomalous region localization is harder for vehicle-pedestrian. Similar trends exist for the non-ego AH* (ahead collision), LA* (lateral collision), VP* (vehicle-pedestrian), and VO* (vehicle-obstacle) anomalies. Second, frame-level and object-centric methods compensate each other in VAD as shown by the Ensemble method's highest AUC and STAUC values in most columns. Third, localizing anomalous regions in non-ego anomalies (marked with *) is more difficult, perhaps because ego-involved anomalies have better dashcam visibility and larger anomalous regions. We also note that certain classes are especially challenging for various reasons; for example: pedestrians in vehicle-pedestrian (VP) videos become occluded or disappear quickly after an anomaly happens, the impacting vehicle in non-ego ahead (AH*) is often occluded by the vehicle it impacts, obstacles in non-ego vehicle-obstacle (VO*) such as bumpers or traffic cones are often occluded or not detected, and vehicles in non-ego lateral (LA*) usually move toward each other slowly until they collide and stop, making the anomaly relatively subtle.

Qualitative Results. Fig. 9a shows per-frame anomaly scores and TARRs of three methods on a video in which they all achieve high AUCs. AnoPred+Mask has low TARR along the video, indicating failure to localize anomalous regions. FOL-STD computes high anomaly scores but low TARR in

Authorized licensed use limited to: Sungkyunkwan University. Downloaded on June 21,2023 at 11:45:09 UTC from IEEE Xplore. Restrictions apply.

TABLE 3
Comparison of Video Anomaly Techniques on the DoTA Dataset, According to the AUC and STAUC Metrics

Method	Input	AUC ↑	STAUC ↑
ConvAE [12]	Gray	64.3	7.4
	Flow	66.3	7.9
ConvLSTMAE [17]	Gray	53.8	12.7
	Flow	62.5	12.2
AnoPred [8]	RGB	67.5	24.4
	Masked RGB	64.8	42.1
FOL-IoU	Box + Flow + Ego	61.2	34.6
FOL-Mask	Box + Flow + Ego	64.0	35.0
FOL-STD	Box	66.7	40.9
	Box + Ego	67.8	42.1
	Box + Flow	69.1	43.1
	Box + Flow + Ego	**69.7**	**43.7**
FOL-Ensemble	RGB + Box + Flow + Ego	**73.0**	**48.5**

the left example due to inaccurate trajectory prediction for the left car. In the right image, it finds an anomalous car but marks an unrelated car by mistake. Ensemble combines the benefits of both with scores for the 20-30th anomaly frames always higher than normal frames. It yields high TARR during the 10-20th anomaly frames as shown in the left score map. The right map shows a failure case combining the failure of AnoPred+Mask and FOL-STD. Although these methods achieve high AUC, their spatial localization is limited according to TARR. Fig. 9b shows an ego-involved ahead collision (AH). AnoPred+Mask incorrectly computes a high anomaly score in the early frames because the prediction of the left car is inaccurate. FOL-STD computes a low anomaly score for this frame, so the Ensemble method benefits. The right example shows that FOL-STD computes a high score correctly for the car ahead but incorrectly for the bus. Ensemble benefits from AnoPred+Mask so that it focuses more attention on the car instead of the bus.

6.2 Task 2: Video Action Recognition (VAR)

VAD detects the temporal range of an anomalous event but does not understand the anomaly type. The goal of Video Action Recognition (VAR) is to classify each video clip into one anomaly category. Taking advantage of the rich categorical annotation of the DoTA dataset, we benchmark seven VAR methods: C3D [46], I3D [47], R3D [48], MC3 [48], R(2+1)D [48], TSN [45], and SlowFast [20]. The previous training/test split is used. Unknown UK(*) anomalies are ignored, yielding 3216 training and 1369 test videos. We trained all models with SGD, learning rate 0.01 and batch size 16 on NVidia Titan XP GPUs. Models are initialized with pre-trained weights from Sports-1M [84] for C3D and Kinetics [85] for the other methods; 0.5 probability random horizontal flips create data augmentation. For evaluation, we randomly selected ten clips from each test video (as in [20]), except for TSN which uses 25 clips per video.

Table 5 shows the results. Although newer methods R(2+1)D and SlowFast achieve higher average accuracy, all suffer from low accuracy on DoTA, indicating that traffic anomaly classification is challenging. First, distant anomalies and occluded objects create difficulties; VO (vehicle-obstacle) and VO* are particularly hard to classify due to low visibility and diverse obstacle types (as seen in Section 6.1). AH* (ahead collision) and OC* (oncoming collision) are also difficult since oncoming vehicles are often occluded. Second, some anomalies are visually similar. For example, ST (start/stop/stationary) and ST* are rare and look similar to AH (ahead) and AH* or LA (lateral) and LA* (Fig. 4) — the only difference is whether the vehicle is starting, stopping, or stationary. Third, the anomaly category is often determined near the anomaly start time, while the later frames do not reveal this category clearly. We have observed 2-4% accuracy improvement when testing models only on the first half of each clip. Additional benchmarks are available in our supplementary material, which can be found on the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TPAMI.2022.3150763.

6.3 Task 3: Online Action Detection (OAD)

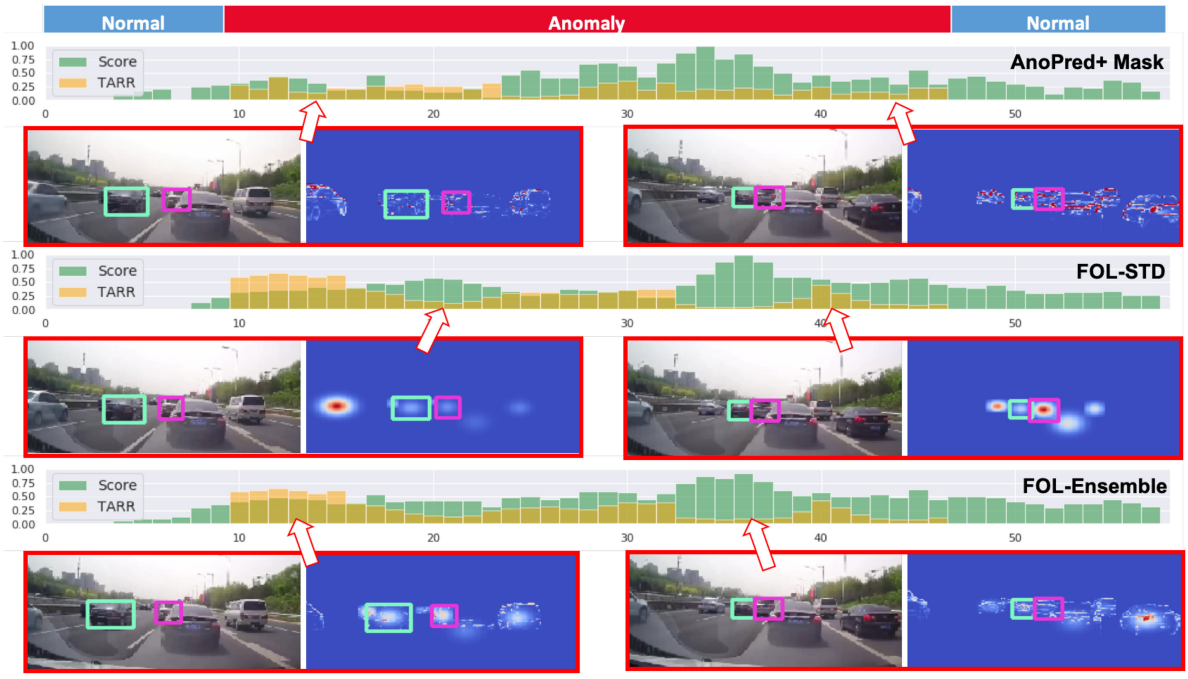
Finally we provide benchmarks for online video action detection on the DoTA dataset. OAD recognizes the anomaly type by only observing the current and past frames,

TABLE 4
Video Anomaly Detection Results for Each Type of Anomaly Class

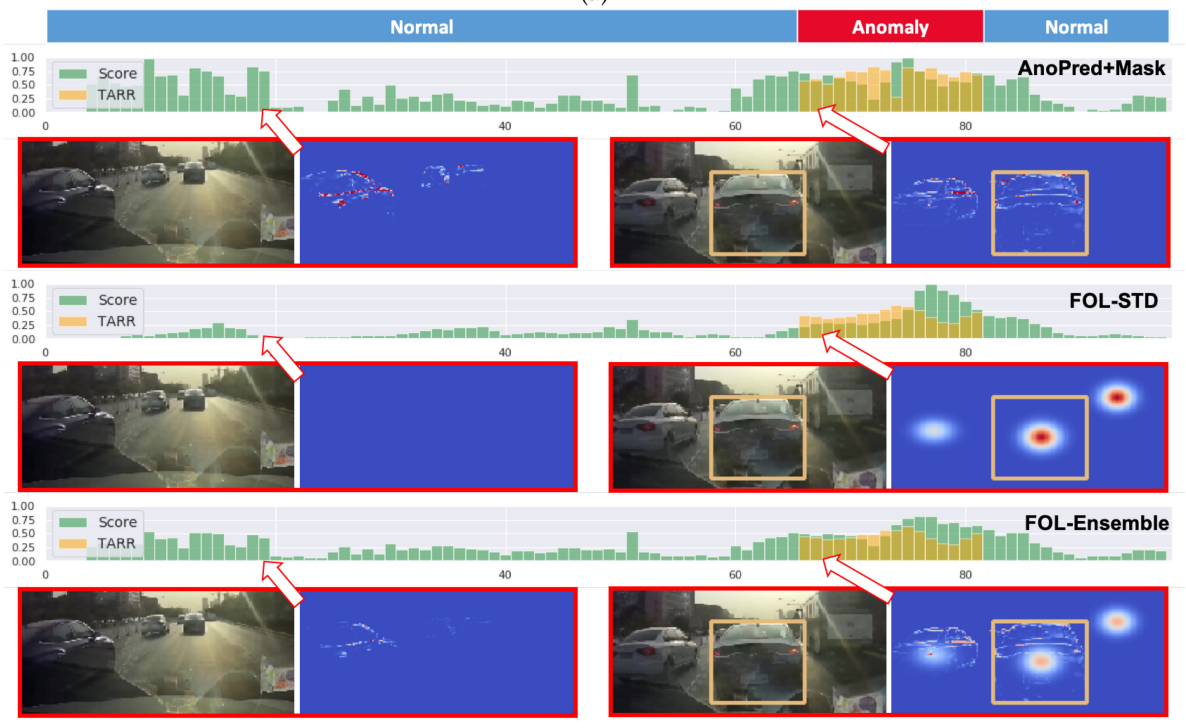
	Ego involved anomaly classes						Non-ego involved anomaly classes							
	ST	AH	LA	OC	TC	VP	ST*	AH*	LA*	OC*	TC*	VP*	VO*	OO*
	Individual Anomaly Class AUC:													
AnoPred [8]	69.9	73.6	**75.2**	69.7	73.5	66.3	70.9	62.6	60.1	65.6	65.4	64.9	64.2	57.8
AnoPred [8]+Mask	66.3	72.2	64.2	65.4	65.6	66.6	72.9	63.7	60.6	66.9	65.7	64.0	58.8	59.9
FOL-STD	67.3	77.4	71.1	68.6	69.2	65.1	75.1	66.2	66.8	74.1	72.0	69.7	63.8	69.2
FOL-Ensemble	**73.3**	**81.2**	74.0	**73.4**	**75.1**	**70.1**	**77.5**	**69.8**	**68.1**	**76.7**	**73.9**	**71.2**	**65.2**	**69.6**
	Individual Anomaly Class STAUC:													
AnoPred [8]	37.4	31.5	32.8	34.3	33.6	24.9	25.9	15.0	12.5	13.0	20.9	14.0	8.2	8.8
AnoPred [8]+Mask	51.8	51.9	45.1	50.3	47.5	**41.0**	45.3	31.1	33.8	42.5	40.3	25.3	22.9	33.8
FOL-STD	47.4	55.6	46.3	52.2	47.2	26.6	45.1	33.6	38.5	46.9	39.3	25.6	**29.0**	**44.4**
FOL-Ensemble	**54.4**	**60.3**	**53.8**	**56.5**	**54.9**	35.2	**52.4**	**36.4**	**40.8**	**51.9**	**44.7**	**28.6**	28.6	43.5

__ST__: collision with another vehicle that starts, stops, or is stationary; __AH__: ahead collision; __LA__: lateral collision; __OC__: oncoming collision; __TC__: turning or crossing collision; __VP__: vehicle-pedestrian collision; __VO__: vehicle-obstacle collision; __OO__: out-of-control; __UK__: unknown. Ego-involved and non-ego (*) anomalies shown separately. VO and OO are not shown because they do not contain anomalous traffic participants.

Authorized licensed use limited to: Sungkyunkwan University. Downloaded on June 21,2023 at 11:45:09 UTC from IEEE Xplore. Restrictions apply.



(a)



(b)

Fig. 9. Per-frame anomaly scores and TARRs of three methods for two different videos. The left and right columns show sample video frames and corresponding score maps, with an arrow indicating their temporal position within the video. Note that TARR only exists in positive frames.

making it suitable for autonomous driving applications. Since OAD does not have a full observation of the whole video sequence, it is more difficult than traditional VAR. We benchmark four OAD methods: *FC*, *LSTM*, *Encoder-decoder*, and *TRN*. The classifiers are designed to predict one out of the 16 anomaly categories. We use the same training configurations as in [21]. Table 6 shows the per-class average precision (AP) and the mean average prediction (mAP). We observe that although TRN, a state-of-the-art method, achieves the highest mAP, all methods suffer from low precision on DoTA. Similar to what we have observed in the VAD and VAR experiments, online action detection is also difficult for ST (start/stop/stationary), ST*, VP (vehicle-pedestrian), VP*, VO (vehicle-obstacle), and VO*. AH* (ahead) an OC* (oncoming) are also difficult due to the highly occluded front of a typical oncoming vehicle. We also observe that ego-involved anomalies are easier to recognize than non-ego anomalies due to their higher visibility.

Authorized licensed use limited to: Sungkyunkwan University. Downloaded on June 21,2023 at 11:45:09 UTC from IEEE Xplore. Restrictions apply.

TABLE 5
Video Action Recognition Per-Class and Mean Top-1 Accuracies on DoTA

Method	Backbone	Ego involved anomaly classes								Non-ego involved anomaly classes								AVG
		ST	AH	LA	OC	TC	VP	VO	OO	ST*	AH*	LA*	OC*	TC*	VP*	VO*	OO*	
TSN	ResNet50	18.2	67.2	**52.9**	**53.8**	**71.0**	0.0	0.0	61.6	0.0	14.7	25.3	6.7	48.1	9.5	0.0	**53.4**	30.2
C3D	VGG16	25.5	61.8	43.9	47.8	57.9	3.3	4.4	52.9	1.2	18.4	36.0	6.7	55.9	8.6	6.0	33.2	29.0
I3D	InceptionV1	10.0	62.4	45.8	45.8	62.2	2.8	6.9	66.6	2.4	**28.1**	24.5	4.7	60.3	9.5	5.0	37.6	29.7
R3D	ResNet18	0.0	56.5	49.6	49.8	66.6	4.4	6.2	47.7	1.8	17.6	32.2	1.0	48.3	**15.2**	6.5	48.0	28.2
MC3D	ResNet18	6.4	62.9	40.1	57.7	64.5	16.7	0.0	61.5	2.4	18.1	20.2	4.0	62.2	4.8	**6.5**	45.6	29.6
R(2+1)D	ResNet18	4.5	64.7	42.8	47.6	68.7	**25.6**	5.6	64.4	**9.4**	14.3	24.3	2.3	**64.7**	9.5	0.0	47.8	**31.0**
SlowFast	ResNet50	0.0	**70.0**	46.0	48.9	67.2	5.6	**13.1**	**68.3**	5.9	24.9	**37.2**	3.3	64.0	0.0	0.0	41.3	**31.0**

See Table 4 Caption for Class Abbreviations

TABLE 6
Online Video Action Detection Average Precisions on DoTA

Method	Ego involved anomaly classes								Non-ego involved anomaly classes								mAP
	ST	AH	LA	OC	TC	VP	VO	OO	ST*	AH*	LA*	OC*	TC*	VP*	VO*	OO*	
FC	**2.5**	13.9	10.6	6.2	16.3	0.8	**1.2**	21.0	0.6	2.9	3.0	0.6	8.0	**1.2**	0.7	7.6	9.9
LSTM	0.6	19.9	15.1	9.2	25.3	2.4	0.6	34.3	0.6	3.8	5.0	1.5	11.0	1.2	0.5	13.3	12.9
Encoder-Decoder	0.5	20.1	15.6	10.4	28.1	**2.9**	0.7	**39.9**	**0.8**	3.7	7.4	2.5	14.7	1.2	0.5	13.2	14.5
TRN	1.0	**22.8**	**20.6**	**15.5**	**30.0**	1.5	0.7	32.3	0.7	**4.0**	**10.2**	**2.9**	**17.0**	1.2	**0.7**	**13.8**	**15.3**

See Table 4 caption for class abbreviations.

7 Conclusion and Future Work

This paper proposed a novel FOL-based unsupervised video anomaly detection (VAD) method for driving videos. A prediction consistency metric was introduced for computing anomaly scores which is robust to inaccurate object detection and tracking in driving videos. We further introduce an ensemble method to combine object- and frame-level VAD methods to boost performance. We also introduced DoTA, a large-scale dataset containing temporal, spatial, and categorical annotations and benchmarked state-of-the-art VAD methods. We proposed a new spatial-temporal area under curve (STAUC) metric to better evaluate VAD performance. Experimental results show that our method achieves state-of-the-art results on DoTA in terms of both AUC and STAUC. Our DoTA dataset also enables research on video action recognition (VAR) and online action detection in driving scenarios; both of these problems are far from solved according to experimental results. Future work will include but not limited to spatio-temporal localization of anomalies in driving scenarios, early detection of traffic accidents, and validation and verification of autonomous driving systems.

Acknowledgments

The views and conclusions contained in this paper are those of the authors and should not be interpreted as representing the official policies, either expressly or implied, of the U.S. Government or any sponsor.

References

[1] F.-H. Chan, Y.-T. Chen, Y. Xiang, and M. Sun, "Anticipating accidents in dashcam videos," in *Proc. Asian Conf. Comput. Vis.*, 2016, pp. 136–153.

[2] R. Herzig *et al.*, "Spatio-temporal action graph networks," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshop*, 2019, pp. 2347–2356.

[3] J. Fang, D. Yan, J. Qiao, and J. Xue, "DADA: A large-scale benchmark and model for driver attention prediction in accidental scenarios," 2019, *arXiv:1912.12148*.

[4] C. Gu *et al.*, "AVA: A video dataset of spatio-temporally localized atomic visual actions," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 6047–6056.

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

[6] R. Hou, C. Chen, and M. Shah, "Tube convolutional neural network (T-CNN) for action detection in videos," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 5823–5832.

[7] J. Tang, J. Xia, X. Mu, B. Pang, and C. Lu, "Asynchronous interaction aggregation for action detection," 2020, *arXiv: 2004.07485*.

[8] W. Liu, W. Luo, D. Lian, and S. Gao, "Future frame prediction for anomaly detection—a new baseline," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 6536–6545.

[9] N. Kalra and S. M. Paddock, "Driving to safety: How many miles of driving would it take to demonstrate autonomous vehicle reliability?," in *Transportation Research Part A: Policy and Practice*, Santa Monica, CA, USA: RAND Corporation, 2016.

[10] R. Nayak, U. C. Pati, and S. K. Das, "A comprehensive review on deep learning-based methods for video anomaly detection," *Image Vis. Comput.*, vol. 106, 2020, Art. no. 104078.

[11] C. Lu, J. Shi, and J. Jia, "Abnormal event detection at 150 FPS in MATLAB," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2013, pp. 2720–2727.

[12] M. Hasan, J. Choi, J. Neumann, A. K. Roy-Chowdhury , and L. S. Davis, "Learning temporal regularity in video sequences," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 733–742.

[13] W. Luo, W. Liu, and S. Gao, "A revisit of sparse coding based anomaly detection in stacked RNN framework," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 341–349.

[14] R. Morais, V. Le, T. Tran, B. Saha, M. Mansour, and S. Venkatesh, "Learning regularity in skeleton trajectories for anomaly detection in videos," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 11988–11996.

[15] R. T. Ionescu, F. S. Khan, M.-I. Georgescu, and L. Shao, "Object-centric auto-encoders and dummy anomalies for abnormal event detection in video," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 7834–7843.

[16] J. R. Medel and A. Savakis, "Anomaly detection in video using predictive convolutional long short-term memory networks," 2016, *arXiv:1612.00390*.

[17] Y. S. Chong and Y. H. Tay, "Abnormal event detection in videos using spatiotemporal autoencoder," in *Proc. Int. Symp. Neural Netw.*, 2017, pp. 189–196.

Authorized licensed use limited to: Sungkyunkwan University. Downloaded on June 21,2023 at 11:45:09 UTC from IEEE Xplore. Restrictions apply.

[18] W. Luo, W. Liu, and S. Gao, "Remembering history with convolutional LSTM for anomaly detection," in *Proc. IEEE Int. Conf. Multimedia Expo*, 2017, pp. 439–444.

[19] J. Bakker, H. Jeppsson, L. Hannawald, F. Spitzhüttl, A. Longton, and E. Tomasch, "IGLAD-International harmonized in-depth accident data," in *Proc. 25th Int. Tech. Conf. Enhanced Saf. Veh.*, 2017, pp. 1–12.

[20] C. Feichtenhofer, H. Fan, J. Malik, and K. He, "Slowfast networks for video recognition," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 6201–6210.

[21] M. Xu, M. Gao, Y.-T. Chen, L. S. Davis, and D. Crandall, "Temporal recurrent networks for online action detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 5531–5540.

[22] Y. Yao, M. Xu, Y. Wang, D. Crandall, and E. M. Atkins, "Unsupervised traffic accident detection in first-person videos," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2019, pp. 273–280.

[23] W. Li, V. Mahadevan, and N. Vasconcelos, "Anomaly detection and localization in crowded scenes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 1, pp. 18–32, Jan. 2014.

[24] W. Sultani, C. Chen, and M. Shah, "Real-world anomaly detection in surveillance videos," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 6479–6488s.

[25] F. Yu, W. Xian, Y. Chen, F. Liu, M. Liao, V. Madhavan, and T. Darrell, "BdDD100K: A diverse driving video database with scalable annotation tooling," 2018, *arXiv: 1805.04687*.

[26] W. Bao, Q. Yu, and Y. Kong, "Uncertainty-based traffic accident anticipation with spatio-temporal relational learning," in *Proc. 28th ACM Int. Conf. Multimedia*, 2020, pp. 2682–2690.

[27] D. Gong *et al.*, "Memorizing normality to detect anomaly, Memory-augmented deep autoencoder for unsupervised anomaly detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 1705–1714.

[28] J. Wang and A. Cherian, "GODS: Generalized one-class discriminative subspaces for anomaly detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 8200–8210.

[29] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei , and S. Savarese, "Social LSTM: Human trajectory prediction in crowded spaces," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 961–971.

[30] A. Gupta, J. Johnson, L. Fei-Fei , S. Savarese, and A. Alahi, "Social GAN: Socially acceptable trajectories with generative adversarial networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 2255–2264.

[31] N. Lee, W. Choi, P. Vernaza, C. B. Choy, P. H. Torr, and M. Chandraker, "Desire: Distant future prediction in dynamic scenes with interacting agents," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 2165–2174.

[32] N. Deo and M. M. Trivedi, "Multi-modal trajectory prediction of surrounding vehicles with maneuver based LSTMs," in *Proc. IEEE Intell. Veh. Symp.*, 2018, pp. 1179–1184.

[33] B. Ivanovic, E. Schmerling, K. Leung, and M. Pavone, "Generative modeling of multimodal multi-human behavior," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2018, pp. 3088–3095.

[34] C. Choi, A. Patil, and S. Malla, "Drogon: A causal reasoning framework for future trajectory forecast," 2019, *arXiv: 1908.00024*.

[35] B. Ivanovic and M. Pavone, "The trajectron: Probabilistic multi-agent trajectory modeling with dynamic spatiotemporal graphs," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 2375–2384.

[36] T. Salzmann, B. Ivanovic, P. Chakravarty, and M. Pavone, "Trajectron++: Multi-agent generative trajectory forecasting with heterogeneous data for control," 2020, *arXiv: 2001.03093*.

[37] H. Zhao *et al.*, "TNT: Target-driven trajectory prediction," 2020, *arXiv: 2008.08294*.

[38] Y. Yao, E. Atkins, M. Johnson-Roberson , R. Vasudevan, and X. Du, "BiTraP: Bi-directional pedestrian trajectory prediction with multi-modal goal estimation," *IEEE Robot. Automat. Lett.*, vol 6, no. 4, pp. 1463–1470, Apr. 2021.

[39] C. Wang, Y. Wang, M. Xu, and D. Crandall, "Stepwise goal-driven networks for trajectory prediction," *IEEE Robot. Automat. Lett.*, vol. 7, no. 2, pp. 2716–2723, Apr. 2022.

[40] T. Yagi, K. Mangalam, R. Yonetani, and Y. Sato, "Future person localization in first-person videos," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 7593–7602.

[41] Y. Yao, M. Xu, C. Choi, D. Crandall, E. M. Atkins, and B. Dariush, "Egocentric vision-based future vehicle localization for intelligent driving assistance systems," in *Proc. Int. Conf. Robot. Automat.*, 2019, pp. 9711–9717.

[42] S. Malla, B. Dariush, and C. Choi, "Titan: Future forecast using action priors," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 11183–11193.

[43] 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 *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 6261–6270.

[44] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Proc. 27th Int. Conf. Neural Inf. Process. Syst.*, 2014, pp. 568–576.

[45] L. Wang *et al.*, "Temporal segment networks: Towards good practices for deep action recognition," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 20–36.

[46] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3D convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 4489–4497.

[47] J. Carreira and A. Zisserman, "Quo vadis, action recognition? A new model and the kinetics dataset," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 4724–4733.

[48] D. Tran, H. Wang, L. Torresani, J. Ray, Y. LeCun, and M. Paluri, "A closer look at spatiotemporal convolutions for action recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 6450–6459.

[49] R. De Geest , E. Gavves, A. Ghodrati, Z. Li, C. Snoek, and T. Tuytelaars, "Online action detection," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 269–284.

[50] J. Gao, Z. Yang, and R. Nevatia, "REE: Reinforced encoder-decoder networks for action anticipation," in *Proc. Brit. Mach. Vis. Conf.*, 2017.

[51] Z. Shou *et al.*, "Online detection of action start in untrimmed, streaming videos," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 534–551.

[52] M. Gao, M. Xu, L. S. Davis, R. Socher, and C. Xiong, "StartNet: Online detection of action start in untrimmed videos," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 5541–5550.

[53] H. Eun, J. Moon, J. Park, C. Jung, and C. Kim, "Learning to discriminate information for online action detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 806–815.

[54] M. Gao, Y. Zhou, R. Xu, R. Socher, and C. Xiong, "WOAD: Weakly supervised online action detection in untrimmed videos," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 1915–1923.

[55] H. Eun, J. Moon, J. Park, C. Jung, and C. Kim, "Temporal filtering networks for online action detection," *Pattern Recognit.*, vol. 111, 2021, Art. no. 107695.

[56] M. Xu *et al.*, "Long short-term transformer for online action detection," in *Proc. Conf. Neural Inf. Process. Syst.*, 2021.

[57] 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 *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 464–474.

[58] J. Zhao *et al.*, "Tuber: Tube-transformer for action detection," 2021, *arXiv:2104.00969*.

[59] H. Zhu, H. Wei, B. Li, X. Yuan, and N. Kehtarnavaz, "A review of video object detection: Datasets, metrics and methods," *Appl. Sci.*, vol. 10, 2020, Art. no. 7834.

[60] K. Kang, W. Ouyang, H. Li, and X. Wang, "Object detection from video tubelets with convolutional neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 817–825.

[61] X. Zhu, Y. Xiong, J. Dai, L. Yuan, and Y. Wei, "Deep feature flow for video recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 4141–4150.

[62] X. Zhu, Y. Wang, J. Dai, L. Yuan, and Y. Wei, "Flow-guided feature aggregation for video object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 408–417.

[63] X. Zhu, J. Dai, L. Yuan, and Y. Wei, "Towards high performance video object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 7210–7218.

[64] C. Feichtenhofer, A. Pinz, and A. Zisserman, "Detect to track and track to detect," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 3057–3065.

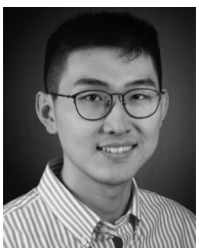
[65] S. Wang, Y. Zhou, J. Yan, and Z. Deng, "Fully motion-aware network for video object detection," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 557–573.

[66] F. Xiao and Y. J. Lee, "Video object detection with an aligned spatial-temporal memory," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 494–510.

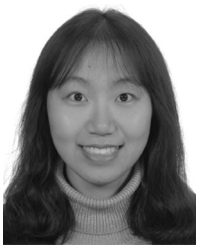
[67] M. Liu and M. Zhu, "Mobile video object detection with temporally-aware feature maps," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 5686–5695.

Authorized licensed use limited to: Sungkyunkwan University. Downloaded on June 21,2023 at 11:45:09 UTC from IEEE Xplore. Restrictions apply.

[68] M. Liu, M. Zhu, M. White, Y. Li, and D. Kalenichenko, "Looking fast and slow: Memory-guided mobile video object detection," 2019, *arXiv:1903.10172*.
[69] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
[70] Y. Chen, Y. Cao, H. Hu, and L. Wang, "Memory enhanced global-local aggregation for video object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 10337–10346.
[71] M. Han, Y. Wang, X. Chang, and Y. Qiao, "Mining inter-video proposal relations for video object detection," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 431–446.
[72] A. Bhattacharyya, M. Fritz, and B. Schiele, "Long-term on-board prediction of people in traffic scenes under uncertainty," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 4194–4202.
[73] N. Wojke, A. Bewley, and D. Paulus, "Simple online and realtime tracking with a deep association metric," in *Proc. IEEE Int. Conf. Image Process.*, 2017, pp. 3645–3649.
[74] W. Li, V. Mahadevan, and N. Vasconcelos, "Anomaly detection and localization in crowded scenes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 1, pp. 18–32, 2013.
[75] T. Suzuki, H. Kataoka, Y. Aoki, and Y. Satoh, "Anticipating traffic accidents with adaptive loss and large-scale incident db," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 3521–3529.
[76] B. Ramachandra, M. Jones, and R. R. Vatsavai, "A survey of single-scene video anomaly detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, Nov. 25, 2020, doi: 10.1109/TPAMI.2020.3040591.
[77] P. Bergmann, M. Fauser, D. Sattlegger, and C. Steger, "MVtec AD–a comprehensive real-world dataset for unsupervised anomaly detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 9584–9592.
[78] L. Wang, Y. Qiao, and X. Tang, "Video action detection with relational dynamic-poselets," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 565–580.
[79] K. Soomro, H. Idrees, and M. Shah, "Action localization in videos through context walk," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 3280–3288.
[80] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox, "Flownet 2.0: Evolution of optical flow estimation with deep networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 1647–1655.
[81] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2980–2988.
[82] M. Cordts *et al*."The Cityscapes dataset for semantic urban scene understanding," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 3213–3223.
[83] G. Hinton, N. Srivastava, and K. Swersky, "Neural networks for machine learning, lecture 6a: Overview of mini–batch gradient descent," *Cited On*, vol. 14, no. 8, p. 2, 2012.
[84] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, "Large-scale video classification with convolutional neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 1725–1732.
[85] W. Kay *et al.*, "The kinetics human action video dataset," 2017, *arXiv: 1705.06950*.



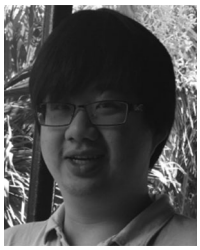
Yu Yao (Member, IEEE) received the BEng degree in aerospace engineering from the Beijing Institute of Technology in 2015 and the MS degree in robotics from the University of Michigan in 2017. He is currently working toward the PhD degree with the University of Michigan Robotics Institute. His research interests include trajectory prediction, anomaly detection, action recognition/prediction, scene understanding, and their applications to autonomous vehicles and intelligent transportation systems.



Xizi Wang (Member, IEEE) received the BS degree in information security and the MS degree in electronic and communications engineering from Shanghai Jiao Tong University in 2016 and 2019, respectively. She is currently working toward the PhD degree with Indiana University. Her research interests mainly include egocentric vision analysis and action recognition.

Mingze Xu (Member, IEEE) received the BEng degree in software engineering from Jilin University in 2012, and the MS and PhD degrees in computer science from Indiana University in 2014 and 2020, respectively. His research interests include computer vision and deep learning, especially on action and activity recognition, first-person (egocentric) vision, image and video segmentation, and embodied visual recognition.

Zelin Pu received the BEng degree in mechanical engineering from the University of Michigan at Ann Arbor in 2019. He is currently growing a startup company KIRAIN, aiming to provide a more intelligent future for the pharmaceutical industry. His research interests include optical character recognition and drug synthesis route prediction.



Yuchen Wang (Member, IEEE) received the BE degree in software engineering from Tongji University in 2011, and the MS degree in computer science in 2017 from Indiana University, Bloomington, where he is currently working toward the PhD degree with the School of Informatics and Computing. His research interests include computer vision and robotics.

Ella Atkins (Senior Member, IEEE) received the BS and MS degrees in aeronautics and astronautics from MIT, and the MS and PhD degrees in computer science and engineering from the University of Michigan. She is currently a professor of aerospace engineering and robotics with the University of Michigan, where she directs the Autonomous Aerospace Systems Lab. She is the editor-in-chief of the *AIAA Journal of Aerospace Information Systems* (JAIS) and a fellow of the American Institute of Aeronautics and Astronautics. She has served on multiple National Academy study panels, as a member of Aeronautics and Space Engineering Board, and pursues research in aerospace system contingency management, autonomy, and safety.

David J. Crandall (Member, IEEE) received the BS and MS degrees in computer science and engineering from Pennsylvania State University in 2001, and the MS and PhD degrees in computer science from Cornell University in 2007 and 2008, respectively. He is currently the Luddy Professor of computer science with Indiana University. His research interests include computer vision, machine learning, and data mining. He was the recipient of the National Science Foundation CAREER Award, two Google Faculty Research Awards, IU Trustees Teaching Award, Grant Thornton Fellowship, Luddy named professorship, and numerous best paper awards and nominations. He is an associate editor for IEEE TPAMI and IEEE TMM.

▷ **For more information on this or any other computing topic, please visit our Digital Library at** www.computer.org/csdl.

Authorized licensed use limited to: Sungkyunkwan University. Downloaded on June 21,2023 at 11:45:09 UTC from IEEE Xplore. Restrictions apply.