Abstract—Identification of high-risk driving situations is generally approached through collision risk estimation or accident pattern recognition. In this work, we approach the problem from the perspective of subjective risk. We operationalize subjective risk assessment by predicting driver behavior changes and identifying the cause of changes. To this end, we introduce a new task called driver-centric risk object identification (DROID), which uses egocentric video to identify object(s) influencing a driver’s behavior, given only the driver’s response as the supervision signal. We formulate the task as a cause-effect problem and present a novel two-stage DROID framework, taking inspiration from models of situation awareness and causal inference. A subset of data constructed from the Honda Research Institute Driving Dataset (HDD) is used to evaluate DROID. We demonstrate state-of-the-art DROID performance, even compared with strong baseline models using this dataset. Additionally, we conduct extensive ablative studies to justify our design choices. Moreover, we demonstrate the applicability of DROID for risk assessment.

Index Terms—Causal inference, egocentric driver behavior modeling, risk object identification, situation awareness.

I. INTRODUCTION

MORE than 1.3 million people die in road accidents worldwide every year, or approximately 3 700 people per day [1]. Road traffic accidents are among the leading causes of non-natural death around the world. The majority of these accidents are due to driver errors, such as exercising poor awareness [2]. To reduce the number of accidents, developing intelligent driving systems such as advanced driver assist systems (ADAS) that identify high-risk situations is in urgent need.

This problem of risk assessment has been studied extensively in the literature [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15], [16], [17], [18], [19], [20], [21], [22], and has been approached by modeling risk as objective or subjective. In this work, we focus on subjective risk [23], i.e., the driver’s own perceived risk, which is an output of the driver’s cognitive process. For example, in Fig. 1, the driver reacts to the crossing pedestrian (i.e., the driver slows down) when passing through the intersection. The driver then reacts to the construction cone (i.e., deviates into the oncoming lane). In these daily tasks, drivers constantly observe traffic situations and plan accordingly to avoid potential hazards. Understanding driver behavior, i.e., when and why a driver reacts to a situation, is critical to the development of intelligent driving systems. We propose to operationalize subjective risk assessment as the prediction of changes in driver behavior and identification of the cause of such changes. We then discuss the computational framework corresponding to this operationalization.

A natural question arises: What changes drivers’ behavior? We propose a new task called driver-centric risk object identification (DROID), which uses egocentric video collected from front-facing cameras to identify object(s) influencing a driver’s behavior, given only the driver’s response as the supervision signal. Note that risk object identification from egocentric videos is crucial for safety systems such as ADAS, where front-facing cameras are the primary device. The proposed setting is more

Fig. 1. Human drivers perceive scenes, assess risks, make a plan, and take actions in different traffic situations. Risk assessment, identifying hazards and risk factors that have the potential to cause harm, is indispensable for driving safety. In this work, we focus on the prediction of driver behavior changes and identification of the cause of changes. We cast the problem as a cause-effect problem. We propose a novel computational framework that identifies traffic participants (cause, such as the crossing pedestrian shown) making drivers react (effect).
Fig. 2. A conceptual diagram of the proposed two-stage DROID framework. We first predict driver response in a given situation. To identify object(s) influencing driver behavior, we intervene in the input observation by removing a traffic participant at a time (i.e., simulating a situation without the traffic participant), and predict the corresponding driver response. For instance, removing the crossing pedestrian changes driver response (effect). The effects of removing other traffic participants remain the same. We conclude that the crossing pedestrian is the risk object (cause).

challenging than existing tasks because the latter utilize either human annotations of object importance [24], [25], [26], [27], risk regions [17], [20], [28], or human gaze patterns [29], [30], [31], [32], [33] as supervision signals for identification. In contrast, the proposed task must identify driving-related risk or important objects from the driver’s response.

To address this challenging task, we formulate the problem as cause-effect [34], and propose a novel two-stage framework. The core concept of the framework is depicted in Fig. 2. In the first stage, a driver model learns to predict driver response in a given situation. To distinguish whether the driver’s behavior is influenced, we cast the problem as a binary classification, predicting the response of drivers to be Stop or Go. Future works could further refine the granularity of driver response – for example, by including driver actions, such as stepping lightly or heavily on the brake pedal [35]. To accurately predict the response of a driver, a driver behavior model should capture complicated spatio-temporal interactions between a driver and traffic elements (e.g., vehicles, pedestrian, and lane markings). We propose a novel driver behavior model motivated by the model of situation awareness (SA) [36]. Specifically, the proposed model predicts driver responses by encapsulating the driver’s goal (i.e., driver intention), perception (i.e., elements of the environment), comprehension (i.e., interactions between driver and Thing objects and interactions between driver and Stuff objects in 3D), and projection (i.e., intention-aware interaction forecasting). Thing and Stuff objects are defined in Section V-A.

In the second stage of the framework, given a Stop response (i.e., driver behavior is influenced by certain objects), we intervene in input video by removing a tracklet at a time and inpainting the removed area in each frame to simulate a scenario without the presence of the tracklet. The same driver behavior model is applied to predict the effect of the removal. The process iterates through all tracklets and records the corresponding effects on driver response. Note that we assume that the cause of driver response change is either vehicles or pedestrians. The tracklet whose removal causes a maximum response change is the risk object.

Our work differs from existing works threefold:

1) Operationalization of Subjective Risk Assessment: We propose to operationalize subjective risk assessment by predicting driver behavior changes and identifying the cause of such changes.

2) Task Formulation: A new task called DROID is introduced, which aims to identify object(s) influencing driver’s behavior from egocentric videos, given only the driver’s response as the supervision signal.

3) Methodology: A causal inference-based framework is proposed to identify risk objects.

In this work, we make the following substantial extensions to our early results [37]:

1) We re-design the driver behavior model substantially to predict driver response by modeling driver decision processes via the model of SA [36].

2) We systematically benchmark three different tasks on the constructed dataset, i.e., driver response prediction, driver intention prediction, and DROID.

3) We conduct thorough ablative studies to justify the architectural designs.

II. RELATED WORK

Vision-Based Risk Assessment: Numerous attempts have been made to build better systems that can robustly assess high-risk situations to reduce traffic fatalities. Existing objective risk assessment algorithms aim to estimate collision risks by computing of time-to-X (e.g., time-to-collision) [3], [5], [38]; predicting traffic participants’ future trajectories [39], [40], [41], [42], [43], [44], [45], [46], [47] for collision checking [4], [6], [7], [11], [12], [13]; or detecting and anticipating traffic accidents [16], [17], [18], [19], [20], [21]. Reliable state estimation from images of traffic participants and environments are the prerequisite for vision-based collision risk estimation-based methods. Significant advances in state estimation algorithms such as object detection and tracking [48], [49], [50], [51], [52], [53], depth estimation [54], [55], [56], road topology modeling [57], [58], [59], and trajectory prediction [60], [61], [62] are observed in the literature. The authors of [63] provide a comprehensive review of vision-based algorithms for traffic scene understanding.

In this work, instead of estimating collision risks, we focus on subjective risk assessment [23], an output of the driver’s cognitive process. Existing works [22], [64], [65], [66] formulate subjective risk assessment as risk-level classification, which requires human annotators to label the risk level of a scenario (e.g., lane change). This work looks into a different formulation for subjective risk assessment. Specifically, we utilize driver behavior changes as direct outputs of driver’s cognitive process. The formulation mitigates the subjectivity found in the
existing strategy. Furthermore, we also identify the cause of such changes, which are not considered in [22], [64], [65], [66] and are essential for planning and decision making. To this end, we propose a new task called DROID, which aims to identify object(s) influencing driver’s behavior from egocentric videos, given only the driver’s response as the supervision signal. We propose an end-to-end trainable framework that leverages the SA model [36] to encapsulate state estimation, situation understanding, and future prediction for driver response prediction. Furthermore, we incorporate causal inference to improve the performance of prediction and identify risk objects. We empirically demonstrate the effectiveness of our proposed method for DROID. We hope that our findings will pave the way for a tight integration of causal reasoning and vision-based risk assessment, a largely under-explored yet critical path towards reliable intelligent driving systems.

**Vision-Based Driver Behavior Modeling:** Driver behavior modeling has been studied extensively in the intelligent vehicle community. Driver behavior is inherently multimodal and difficult to predict, but effective modeling of driver behavior is indispensable to enabling safe and robust intelligent driving systems (e.g., advanced driver assist systems). Detailed reviews of existing driving behavior modeling can be found [67], [68], [69]. While significant advances have been shown, vision-based driver behavior models that use egocentric video to predict driver’s intention and response have not been widely explored. We review vision-based end-to-end driving models and interaction modeling from egocentric view prediction that are relevant to the proposed driver behavior model in the following.

End-to-end driving models have drawn a considerable attention in the vision community [70], [71], [72], [73], [74], [75], [76], [77], [78]. Recent driving models [75], [77] have shown the effectiveness of incorporating semantics and depth as intermediate representations for driving. The proposed driver behavior model considers both cues as well. Specifically, we utilize semantics to differentiate different traffic participants (i.e., *Thing* and *Stuff*, defined in Section V-A). In addition, depth is used to calculate spatial distances between ego vehicle and traffic participants for relational modeling using graph convolution networks (GCNs) [79]. The conditional imitation learning framework [72], [80], [81] is introduced to condition imitation learning on a high-level command input. Specifically, the framework enables a trained driving policy that can respond to a navigational command. Motivated by this, our proposed driver behavior model conditions driver response prediction on driver intention. We empirically show that the proposed designs are effective for driver response prediction. Moreover, we demonstrate the trained driver behavior model is valuable for risk object identification. Note that the proposed driver behavior model is complementary to end-to-end driving models. Novel architectural designs of end-to-end driving modeling may benefit vision-based driver behavior modeling and vice versa.

Interaction modeling from an egocentric view has been explored in pedestrian behavior prediction [45], [61]; driver attention prediction [21], [33]; driver intention prediction [82], [83], [84]; important object detection [27]; and human-human or human-object interaction [85], [86], [87]. Our work is closely related to [27], [84], [86], [87] because these works explicitly model interactions between ego (e.g., camera wearer [87]) and interactors (e.g., objects in kitchen [88]). The explicit modeling between ego and interactors is important for understanding the behavior of the camera wearer or ego vehicles [84], [86]. In the Brain4Car project [84], Jain et al. propose to combine spatio-temporal graphs and Long Short-term Memory (LSTM) to capture the relationship between the driver and inside/outside context for driver intention prediction. In [86], [87], the authors exploit LSTM [89] to model the relation between ego and interactors. Instead of using LSTM, Zhang et al. [27] leverage GCNs [79] to model interactions between ego vehicle and traffic participants. In this work, we use GCNs to model interactions. This work differs from those previously described in that we introduce two types of interactions (i.e., interactions between ego vehicle and *Thing* objects, and interactions between driver and *Stuff* objects in 3D) for driver response prediction and risk object identification. Empirically, we demonstrate the importance of the proposed architecture for the targeted applications.

**Causality in Computer Vision:** Computer vision research has proliferated over the past decades due to the advance in deep learning algorithms. However, current deep learning models may suffer from spurious correlation [90] as a result of ignoring causality in data, although in fact, humans perceive the causality of the physical world. To address the issue, recent studies [91], [92], [93], [94], [95] explicitly consider the concept of causality in deep learning architectural designs.

To our best knowledge, we are among the first to apply causal inference to egocentric images captured in driving scenes. Kim et al. [96] propose a causality test as a means to identify input regions influencing the output their driving model. We also employ causal inference similar to the causality test of Kim et al. [96]. However, the purpose of causal inference in this work is to identify risk objects that cause drivers to change behaviors. Moreover, we design a simple but effective data augmentation strategy using causal intervention. This leads to a more robust driving model.

Haan et al. [97] propose to incorporate functional causal models [34] into imitation learning to address the issue of “causal misidentification,” a phenomenon in which accessing to more information leads to unsatisfactory generalization performance in the presence of distributional shift in imitation learning. Samsami et al. [98] extend [97] to address inertia and collisions found in imitation learning-based autonomous driving policy. A causal approach that mitigates the impact of distributional shifts in motion forecasting is studied in a recent work [99].

Our work is complementary to [97], [98], [99]. Specifically, the focus of these models [97], [98], [99] is to improve the robustness under distribution shifts via causal inference. In this work, causal inference is used to synthesize counterfactual scenarios to improve the performance of driver behavior prediction and identify risk objects.

### III. Dataset

To study DROID, a dataset with diverse reactive scenarios (i.e., drivers react to potential hazards while navigating to their...
goals) is indispensable. For instance, when human drivers intend to turn left at an unprotected intersection, they react (e.g., slowing down or stopping) to certain traffic participants to avoid dangerous situations. Existing datasets [16], [17], [18], [19] are used to study traffic accidents. However, these datasets cannot be utilized to study DROID, which aims to discover the causal relationship between risk object and driver behavior change, because they only have accident data. Thus, we leverage the Honda Research Institute Driving Dataset (HDD) [100] to construct a subset of data for DROID.

Fig. 3 illustrates how we construct the subset from the HDD dataset, hereafter referred to as HDD Subset (HDDS). The Goal-oriented layer defined in the HDD dataset denotes tactical driver behavior such as right turn, left turn, or lane change. As shown in Fig. 3, each frame is labeled with either a goal-oriented or background action. To obtain the Intention of an n-frame clip (the parameter n is 20 in our implementation), we use the last frame’s label of the Goal-oriented layer as the Intention label. While performing a tactical behavior, drivers might have to stop or deviate due to traffic participants or obstacles. We extend the Stimulus-driven actions, i.e., Stop and Deviate, defined in the HDD dataset, as the Response label. Note that both Stop and Deviate are merged into Stop, as depicted in Fig. 3. The rest of the frames are labeled as Go. The HDD dataset also annotates a Cause layer to explain the reason for Stop and Deviate actions. We obtain the Test2 set of the HDDS by selecting frames from the four Cause scenarios, i.e., Congestion, Crossing Pedestrian, Crossing Vehicle, and Parked Vehicle. Moreover, in the Test2 set, we use annotated bounding boxes of risk objects (i.e., object[s] influencing driver’s behavior) from HDD for DROID benchmarks. In [100], given an interactive scenario (i.e., Cause layer is labeled as Crossing Vehicle), Ramanishka et al. work with a third-party annotation company to have two human annotators label the closest object with the same category (vehicle) labeled in the Cause layer. Then, there is a third person who checks the consistency between the two annotators. For our DROID benchmarks, we focus on scenarios in which drivers react to vehicles or pedestrians.

HDDS has 184 890 frames for training driver response and intention predictors. Two test sets are constructed for driver response prediction and DROID, respectively. The Test1 split has 63 314 frames for both driver response and intention benchmarks. The Test2 has 630 frames (i.e., 630 different risk objects) covering four reactive scenarios, i.e., Congestion, Crossing Pedestrian, Crossing Vehicle, and Parked Vehicle for DROID benchmarks. Detailed statistics are shown in Table I. Note that in daily driving, drivers react to diverse traffic participants, e.g., bicyclists, motorcyclists, jaywalkers, traffic signs, construction areas, and so on. A new dataset that covers these interaction types would be invaluable for benchmarking DROID. We leave it for future work.

IV. PROBLEM FORMULATION

Given a reactive scenario with T RGB images \( I := \{I_1, I_2, \ldots, I_T\} \), the goal is to identify the object influencing driver response in the last frame (DROID task).

We formulate this task as a cause-effect problem [34]. Specifically, a two-stage framework is proposed to identify the cause (i.e., the risk object) of an effect (i.e., driver behavior change) via the Situation Awareness-based Driver Behavior Model and Causal Reasoning for DROID. We discuss the methodology in the following.
TABLE I
STATISTICS (ANNOTATED FRAMES) OF HDDS

| Split | BG | IP | LT | RT | LLC | LRC | L.LB | RLB | CP | RP | MG | UT | STP | G |
|-------|----|----|----|----|-----|-----|------|-----|----|----|----|----|-----|----|
| Train | 737949 | 48933 | 21819 | 19824 | 4815 | 4386 | 1833 | 717 | 2364 | 588 | 1182 | 2001 | 184890 | 661521 |
| Test2 | 236622 | 17777 | 7017 | 6195 | 1098 | 1212 | 435 | 324 | 432 | 123 | 327 | 432 | 63314 | 208675 |

Intention (BG) background, (IP) intersection passing, (LT) left turn, (RT) right turn, (LLC) left lane change, (RCL) right lane change, (LLB) left lane branch, (RLB) right lane branch, (CP) crosswalk passing, (RP) railroad passing, (MG) merge, (UT) u-turn.

Response: (STP) stop, (G) go.

V. DRIVER BEHAVIOR MODEL

An overview of the proposed driver behavior model is depicted in Fig. 4. To predict the response of a driver, a driver behavior model should capture complicated spatio-temporal interactions between a driver and traffic participants. We propose a novel driver behavior model motivated by the model of situation awareness [36] (left). Given a video clip, 3D convolutions (I3D [101]), object detection, semantic segmentation, and depth estimation are applied to obtain states of traffic participants in a traffic environment at the Perception stage (Section V-A). At the Comprehension stage, an Ego-Thing Graph and an Ego-Stuff Graph are constructed to model spatiotemporal interactions between a driver and traffic participants (Section V-B). In this work, we categorize traffic participants into two types, i.e., Thing and Stuff. The details are discussed in Section V-B. The final stage, Projection (Section V-C), forecasts future interactions between driver and traffic participants for driver response prediction. Frame-wise interactions obtained from Ego-Thing Graph and Ego-Stuff Graph are fused and fed into an encoder LSTM to form interaction representation. Intention representation obtained from the I3D head and interaction representation are sent to a decoder Temporal Recurrent Network, or TRN (the architecture is shown in Fig. 5) to predict driver response.

A. Perception

Perception plays an essential role in the SA model [36]. This component incorporates the status, attributes, and dynamics of relevant traffic participants of a traffic environment. Specifically, given T RGB images, we apply object detection [48] and semantic segmentation [102] to obtain Thing and Stuff objects, respectively. Previous work [103], [104], [105] has different definitions of Thing and Stuff based on the spatial extent or shape of objects. In this work, we distinguish Stuff objects from Thing objects by evaluating whether states of an object can be influenced by other objects. If yes, we categorize the object as a Thing object. For instance, cars are Thing objects since they stop or yield to obstacles. A traffic light turns red or green by itself, so is categorized as a Stuff object. In addition to detection and segmentation, we perform object tracking using Deep SORT [51] and depth estimation [106].

B. Comprehension

We interpret Comprehension as the spatio-temporal interactions between the driver and Thing objects, and interactions between the driver and Stuff objects in the 3D world. Note that a thorough modeling of Comprehension is beyond the scope of this work. Specifically, we construct two graphs, i.e., Ego-Thing Graphs and Ego-Stuff Graphs, modeled with GCNs [79]. The
details of each graph are discussed below. Note that we choose the interaction modeling proposed in [107] because the model explicitly model interactions among drivers, traffic participants, and road scene infrastructure from egocentric images. In this work, we extend the modeling for driver response prediction.

1) Ego-Thing Graph: The Ego-Thing Graph is designed to model interactions between a driver and Thing objects.

Graph Definition: We denote a sequence of frame-wise Ego-Thing Graphs as $G^{ET}_t = \{G^{ET}_t\}_{t = 1, \ldots, T}$, where $T$ is the number of frames, and $G^{ET}_t \in \mathbb{R}^{(K+1) \times (K+1)}$ is the Ego-Thing affinity matrix at frame $t$ encoding pairwise interactions among Thing objects and Ego. Specifically, $G^{ET}_t(i, j)$ denotes the influence of object $j$ on object $i$. A node $i$ at time $t$ is represented by two types of features $(x^t_i, p^t_i)$, where $x^t_i$ represents the appearance feature, and $p^t_i \in \mathbb{R}^{1 \times 3}$ is the 3D location of the $i$th object in respect to Ego in a local frame.

Node Feature Extraction: Thing objects are car, person, bicycle, motorcycle, bus, train, and truck. Given bounding boxes obtained from object detection [48], we keep $K$ top-scoring detected boxes. The parameter $K$ is set to 20 empirically, as most frames in HDDS have no more than 20 objects. There are $K + 1$ objects, where index $i = 1, 2, \ldots, K$ corresponds to Thing objects, and index $K + 1$ corresponds to Ego. The appearance feature for $i$th object is denoted as $x^t_i \in \mathbb{R}^{1 \times D}, i = 1, 2, \ldots, K, K + 1$. RoIAlign [48] and max pooling are applied to obtain the appearance features of Thing objects. The appearance of Ego is obtained by the same procedure as Thing objects, but with a frame-size bounding box.

Relational Modeling: We consider both appearance features and distance constraints, motivated by the same spatial dimension of an intermediate I3D [101] feature map $X$. Note that the downsampled mask is denoted as $M^t_i$. A Stuff object feature is obtained as follows:

$$x^t_i = \sum_{w=1}^{W} \sum_{h=1}^{H} x^t_{(w,h)} \cdot M^t_{i_{(w,h)}},$$

where $x^t_{(w,h)} \in \mathbb{R}^{1 \times D}$ is a $D$-dimension feature at location $(w, h)$ for time $t$, and $M^t_{i_{(w,h)}}$ is a binary scalar indicating whether object $i$ exists at location $(w, h)$.

Relational Modeling: We neglect interactions among Stuff objects since by definition, Stuff objects are presumed to not have interactions with each other. Thus, we set $f_a$ as in (3) to 0 for every pair of Stuff objects. To model spatial constraint, we unproject every pixel within a downsampled binary mask $M^t_i$ to the 3D space, and calculate the relative distance between the corresponding 3D coordinates and the 3D coordinate of Ego. We choose the minimum distance within a downsampled mask. The distance threshold $\mu$ in Ego-Stuff Graphs is empirically set to be 0.6.

3) Interaction Modeling as Message Passing: In Sections V-B1 and V-B2, two relational models are discussed. To predict driver response, we need interaction modeling that captures influences of multiple traffic participants on a driver. We formulate interactions as message passing in GCN that takes a graph as input, passes information through edges, and outputs updated nodes’ features. The message passing process in GCN is written as:

$$X^{l+1} = GX^lW^l + X^l,$$

where $G$ is the affinity matrix discussed in Sections V-B1 and V-B2. The matrix $X^l \in \mathbb{R}^{(K+1) \times D}$ is the appearance feature matrix for the $l$th layer. $W^l \in \mathbb{R}^{D \times D}$ is a trainable weight matrix. We also build a residual connection by adding $X^l$. Layer Normalization [109] and ReLU are applied before $X^{l+1}$.
Driver intention is indispensable for planning the next action [72], estimating the importance of road users [113], and assessing risk [114]. Similarly, in our task, driver response (i.e., Go and Stop) is determined not only by interactions among traffic participants but also driver intention (e.g., Left Turn or Right Turn). For instance, a vehicle turning right at an intersection will not stop for pedestrians walking on the left sidewalk. Hence, we treat features extracted from the 3D head as the intention representation, since 3D features capture the historical motion dynamics and imply intention information. The representation is used to initialize the hidden state of the first decoder LSTM cell. Note that the design differs from [110], which initializes the hidden state, $h_0$, with zeros.

VI. CAUSAL REASONING

The previous section introduces the proposed driver behavior model. In this section, we discuss how we utilize intervention, a powerful tool for causal inference, as a means for data augmentation to improve the performance of the driver behavior model (Section VI-A) and apply causal inference to identify the risk object (Section VI-B).

A. Driver Behavior Model Training With Data Augmentation Via Intervention

The performance of learning-based driver behavior models depends in a large part on the amount of training data under different traffic configurations [116]. We propose a novel data augmentation strategy via intervention [34]. Intervention is a means to differentiate among the different causal structures that are compatible with an observation [117]. The different causal structures between two events A and B are either A causes B, or B causes A, or they do not influence each other but they have a common cause. We assume that non-causal objects do not influence the behavior of the driver. Thus, we can generate a new data point using the concept of intervention, i.e., removing non-causal objects. For instance, in a Go scenario, a driver enters an intersection while pedestrians walk on the sidewalk in an opposite direction. It is reasonable to assume that driver behavior is the same if a pedestrian is not present. By removing the pedestrian, we can generate a new data point for training non-causal objects. However, exhaustive risk object labeling is costly, and that is not the focus of this work.

We cannot remove causal objects and assume the corresponding driver response to be Go even if causal objects are identified. This is because traffic situations are inherently complicated. The corresponding driver response is unclear when the causal objects are removed. For instance, a driver is in a congestion situation (i.e., driver stops for the frontal vehicle), and the traffic light of the driver’s lane is red. In this situation, the frontal vehicle is labeled as the risk object (cause). However, driver response remains the same if the frontal vehicle were not present because...
Algorithm 1: Driver Behavior Model Training.

\( T \): Number of frames  
\( N \): Number of Thing objects in a given tracklet list  
\( A_r \): Ground truth driver response (either Go or Stop)  

**Input:** A sequence of RGB frames \( I := \{I_1, I_2, \ldots, I_T\} \)  

**Output:** Predicted driver response \( a_r \) and intention \( a_i \).  
Notice that \( a_r \) consists of confidence scores of Go or Stop. \( a_r := \{r^{go}, r^{stop}\} \).

\[
\begin{align*}
1: & \quad O := \text{DetectionAndTracking}(I) \\
& \quad := \{O_1, O_2, \ldots, O_N\} // List of Thing object tracklets \\
2: & \quad S := \text{SemanticSegmentation}(I) \\
& \quad := \{S_1, S_2, \ldots, S_T\} // List of Stuff objects \\
3: & \quad // Data Augmentation via Intervention (Section VI-A) \\
4: & \quad \text{if } A_r \text{ is Go and } N > 1 \text{ then} \\
5: & \quad \quad // Randomly remove a tracklet \\
& \quad \quad k := \text{RandomSelect}(N) \\
6: & \quad \quad \text{else} \\
7: & \quad \quad k \text{ is empty} \\
8: & \quad \quad \text{end if} \\
9: & \quad // Mask out Thing object \( k \) on each mask frame \\
& \quad \quad M := \text{MaskGenerator}(I, O_k) \\
10: & \quad // Remove a Thing object \( k \) from the tracklet list \\
& \quad \quad O' = O - \{O_k\} \\
11: & \quad a_r, a_i := \text{DrivingModelTraining}(I, M, O', S) \\
& \quad // Discussed in Section VI-A \\
12: & \quad \text{return } a_r, a_i
\end{align*}
\]

of the red light. Generating Stop scenarios is non-trivial, and we leave it for future works.

To train driver behavior models with the proposed data augmentation strategy, a model should be able to “intervene,” i.e., remove a non-causal object from images. We realize the strategy by replacing standard convolutional layers in I3D with partial convolutional layers [115, 118]. Note that a partial convolutional layer is initially introduced for image inpainting. We utilize partial convolutions to simulate a scenario without the presence of an object. A 3D partial convolutional layer takes two inputs, i.e., a sequence of RGB frames and a one-channel binary mask for each frame. The pixel values of a mask are 1 by default. While training the driver behavior model with data augmentation, we set the pixels within the selected object to be 0. In addition, the node of the selected object in a graph is disconnected from the rest of the objects. Details can be found in Fig. 6.

The proposed training process is outlined in Algorithm 1. Given training samples in a Go scenario, we randomly select an object \( k \) to intervene, i.e., simulating a situation without the presence of the object. Specifically, given a tracklet \( o_k \), a one-channel binary mask \( M_t \) at time \( t \) is defined as

\[
M_t(i,j) = \begin{cases} 
0, & \text{if } (i,j) \text{ in region } o^t_k \\
1, & \text{otherwise}
\end{cases}
\]  \hspace{1cm} (6)

Algorithm 2: Causal Inference for DROID.

\( T \): Number of frames  
\( N \): Number of objects  

**Input:** A sequence of RGB frames \( I := \{I_1, I_2, \ldots, I_T\} \) where the ego vehicle stops  

**Output:** Risk object ID

\[
\begin{align*}
1: & \quad O := \text{DetectionAndTracking}(I) \\
& \quad := \{O_1, O_2, \ldots, O_N\} // List of Thing object tracklets \\
2: & \quad S := \text{SemanticSegmentation}(I) \\
& \quad := \{S_1, S_2, \ldots, S_T\} // List of Stuff objects \\
3: & \quad \text{for } O_k \in O \text{ do} \\
4: & \quad \quad // Mask out Thing object \( k \) on each frame \\
& \quad \quad M := \text{MaskGenerator}(I, O_k) \\
5: & \quad \quad // Remove the Thing object \( k \) from the tracklet list \\
& \quad \quad O' = O - \{O_k\} \\
6: & \quad \quad // Predict driver response and intention \\
& \quad \quad \text{without the object } k, \text{ where } a_r := \{r^{go}_k, r^{stop}_k\} \\
7: & \quad \quad a_r, a_i := \text{DrivingModel}(I, M, O', S) \\
8: & \quad \text{return } \arg \max_k (r^{go}_k)
\end{align*}
\]

where \( o^t_k \) is the bounding box of the \( k \)th object at time \( t \), and \((i, j)\) is a pixel coordinate within the box. Note that \( k \)th object is discarded from the tracklet list while training the driver behavior model.

B. Causal Inference for DROID

Given a Stop scenario, we aim to identify the corresponding risk object. We deploy the same intervention process discussed in Section VI-A to identify the risk object. Specifically, the masks of a tracklet and the corresponding video frames are processed by the same driver behavior model. The model outputs the confidence score of Go and Stop without the presence of the tracklet. After iterating through all tracklets, we select the object whose tracklet elimination yields the highest Go confidence score to be the risk object. This is because the object causes the most driver behavior change. Algorithm 2 describes the overall causal inference process.

VII. EXPERIMENTS

A. Implementation Details

We implement our framework in TensorFlow. All experiments are performed on a server with four NVIDIA TITAN-XP cards. The input to the framework is a 20-frame clip with a resolution of 224 \( \times \) 224 at 3 fps, approximately 6.67 s. The framework outputs the predictions of driver intention and response of the very next frame. For the data preprocess, we use [102] trained on Mapillary Vistas research set [120] to perform semantic segmentation. We apply object detection to every frame via a
Fig. 6. We simulate a situation using partial convolutional layers [115]. Note that a partial convolutional layer is initially introduced for image inpainting. We utilize partial convolutions to simulate a scenario without the presence of an object. The left-hand side of the figure depicts when an intervention is disabled. To simulate a situation without an object (e.g., the car in the green box), we set the pixels of the binary mask within the green box to 0. In addition, the Ego-Thing Graph is constructed without considering this car as a node.

Fig. 7. DROID results obtained by Causation and Correlation. Note that Causation is the causal inference-based approach proposed in the paper. Instead of using causal inference, Correlation determines the risk object by selecting the object sending the highest attention weight to Ego in the Ego-Thing Graph. The top row shows an egocentric view where green boxes indicate our Causation results, blue boxes are Correlation results, and ground truth boxes are in red. A bird’s-eye-view representation is presented in the bottom row, providing information including scene layout and intentions of traffic participants.
TABLE II
RESULTS OF OUR DRIVER RESPONSE PREDICTION (ROWS 5-8) COMPARED WITH BASELINE PREDICTIONS (ROWS 1-4)

| Model                                      | Response       | Intention      |
|--------------------------------------------|----------------|----------------|
|                                            | Perplexity     | Macro Accuracy | Micro Accuracy | Overall mAP | Overall mAP |
| 1. I3D + LSTM                              | 1.00           | 64.37          | 77.95          | 71.07       | /            |
| 2. I3D + LSTM + Multi-head                 | 0.93           | 68.27          | 79.04          | 70.12       | 36.41        |
| 3. Pixel-level Attention [96]              | 0.89           | 76.15          | 80.21          | 78.57       | /            |
| 4. Object-level Attention [119]            | 0.84           | 78.81          | 83.19          | 79.02       | /            |
| 5. GCN                                     | 0.83           | 77.57          | 82.64          | 80.33       | /            |
| 6. GCN + Multi-head                       | 0.72           | 76.30          | 85.68          | 84.46       | 36.31        |
| 7. GCN + TRN Head                         | 0.69           | 79.32          | 86.17          | 83.44       | 36.80        |
| 8. GCN + TRN Head + Data Augmentation     | 0.37           | 87.63          | 92.56          | 95.44       | 36.73        |

Perplexity (lower is better), macro and micro-average accuracies, and overall mAP are used as metrics for driver response prediction. The unit is % for all metrics except perplexity. The best and second best performances are shown in bold and underlined, respectively. We also report the performance of driver intention prediction using the overall mean average precision (mAP) as the metric.

Mask R-CNN model [48] trained on Cityscape dataset [121]. Based on the detection results, Deep SORT [51] is utilized to associate detected objects into tracks. New objects are identified when the detection cannot be associated with an existing track.

We adopt Inception-v3 [122] pre-trained on ImageNet [123] as the backbone, following [101] to inflate 2D convolution into 3D convolution, and fine-tune it (i.e., I3D) on the Kinetics action recognition dataset [124]. The intermediate feature used in RoIAlign and MaskAlign is the Mixed_3c layer, where the number of feature channels is 512. The intention feature is generated from a $1 \times 1 \times 1$ convolution on the Mixed_5c layer’s feature, and the channel number of the feature is 512. The downsampling of binary mask $M^{i}$ is 28 $\times$ 28. The decoder length is set to 3, and the TRN model is identical to the original implementation [110] except for the hidden state. The number of hidden cells we use is 1024 while it is 2 $\times$ 1000 in [110]. Also, instead of initializing hidden states with zeros, we input the intention representation to fulfill our intention-aware design.

The model is trained in a two-stage training scheme with a batch size of 32. First, we finetune the network with HDDS for 50 000 iterations. At this stage, GCN and TRN are not involved, and we use the driver intention as the training target. At the second stage, we employ the augmentation strategy mentioned in Section VI-A. The weights trained from the first stage are loaded to the completed driver behavior model with GCN and TRN. We finetune the network for another 20 000 iterations. We use the Adam optimizer [125] with the default parameters ($\beta_{1} = 0.9$, $\beta_{2} = 0.999$ and $\epsilon = 1 \times 10^{-8}$) for both stages. The learning rate is set to be 0.001 and 0.0002 for the first and second stages, respectively.

B. Driver Behavior Model Performance

1) Evaluation Setup: The performance of the driver behavior model is evaluated as a discrete feasible action prediction, in accordance with [116], [119], [126]. The two discrete actions, Go and Stop, are evaluated. We follow the train/test1 split in HDDS, where 846 411 and 271 989 samples are used for training and testing, respectively. Four evaluation metrics are utilized. First, we report perplexity as in [116], [119], [126]. Perplexity calculates the negative log-likelihood of predicted probability of Response given ground truth (lower is better). Second, we report the macro-averaged accuracy. Note that, in a multi-class classification setup, the micro-averaged accuracy is preferable if the label distribution is imbalanced. In our task, the Go to Stop ratio is approximately 4:1. Therefore, we also report the micro-averaged accuracy as the third metric. Response prediction can be treated as an online action detection task [110], [127]. We use per-frame mean average precision (mAP) as the fourth evaluation metric.

2) Evaluation: Table II summarizes the results of the driver behavior models. We compare the following baselines. To compare different models, we keep their backbone network (i.e., Inception-v3) the same.

I3D+LSTM: We extract visual features from the Mixed_5c layer of I3D and sequentially input the features at each time step to a two-layer LSTM [110] for temporal modeling.

Pixel-Level Attention: The pixel-level attention module is proposed by [96] to improve model’s interpretability and performance.

Object-Level Attention: In [119], the authors propose an object-centric attention mechanism to augment end-to-end policy learning. Both pixel- and object-level attention modules are incorporated into I3D+LSTM.

The following summarizes our proposals.

GCN: This method is based on our previous work [107]. The key difference between GCN and three baselines is the input feature to the LSTM module. Specifically, the feature is processed via GCNs and contains interaction between traffic participants and driver.

Multi-Head: We add an additional head for driver intention prediction to I3D+LSTM and GCN. A standard cross-entropy loss is used for driver intention prediction. Note that both the interaction and intention features share the same features from the Mixed_5c layer of I3D.

TRN Head: To forecast future interactions, we incorporate TRN [110]. We initialize TRN with intention representation (as shown in Fig. 5).

Intervention: The concept of intervention is utilized to augment training data to improve the performance of driver response prediction discussed in Section VI-A.
We show that GCN outperforms baselines, demonstrating the importance of interaction modeling. By incorporating Multi-head, i.e., intention modeling, both extensions reduce the perplexity by 0.07 and 0.11, respectively. With TRN Head, we observe that perplexity is reduced by 0.03. Finally, we demonstrate that Intervention significantly improves the performance of the driver response prediction (0.32 decrease in perplexity). More detailed discussion can be found in Section VII-B3.

While promising improvements are observed for driver response prediction, the trend does not hold for driver intention prediction, as shown in Table II. Our hypothesis is that this is because driver response prediction utilizes TRN with the interaction and intention representations. Note that the interaction representation is obtained via extracting features from the Mixed_3c layer of I3D, processing the features via GCN, and encoding those processed features by LSTM. In contrast, driver intention prediction only uses the intention representation obtained from the I3D head after the Mixed_5c layer. Therefore, the intention representation is not trained effectively due to its architectural design. In the future work, we plan to incorporate road structures [57], [58], [59] or explicitly model possible goals [128] to improve the performance of driver intention prediction.

3) Ablation Study: We conduct ablation studies to understand the contributions of the proposed architecture designs. The studies are summarized in Table III.

| Model                     | Perplexity |
|---------------------------|------------|
| Intention Modeling        |            |
| Without intention modeling| 0.83       |
| Multi-head                | 0.72       |
| TRN Head                  | 0.69       |
| Different Graph            |            |
| Ego-Stuff Graph           | 0.74       |
| Ego-Thing Graph           | 0.80       |
| Ego-Thing Graph + Ego-Stuff Graph | 0.69 |
| Spatial Modeling          |            |
| Appearance Relation       | 0.73       |
| Appearance + Spatial Relation | 0.69      |
| Data Augmentation         |            |
| Without Augmentation      | 0.69       |
| With Augmentation         | 0.37       |

A lower perplexity indicates better model performance. The best-performing models within each category (row labels) are shown in bold.

| Model                     | Perplexity |
|---------------------------|------------|
| Intention Modeling        | 0.83       |
| Multi-head                | 0.72       |
| TRN Head                  | 0.69       |
| Different Graph            | 0.74       |
| Ego-Stuff Graph           | 0.80       |
| Ego-Thing Graph           | 0.69       |
| Spatial Modeling          | 0.73       |
| Appearance Relation       | 0.69       |
| Data Augmentation         | 0.69       |
| With Augmentation         | 0.37       |

Variations of Different Graphs: The experiments aim to prove the importance of modeling interactions with both Thing and Stuff for driver response prediction. From Table III, we observe that Ego-Stuff and Ego-Thing Graphs capture different aspects of interactions with ego-vehicles. When both Ego-Stuff and Ego-Thing interactions are considered jointly, the model achieves the best perplexity results. The results empirically show the hypothesis that an explicit modeling of drivers, traffic participants, and road infrastructure is crucial for driver response prediction.

Importance of Spatial Relation: We study the importance of the spatial relation function (3) to the Response prediction. We conduct two experiments, i.e., 1) using only the appearance relations, and 2) appending 3D spatial relation as an additional constraint. As we live in a 3D world, an interaction model for driver response prediction should take 3D spatial relations into account. As shown in Table III, we confirm our hypothesis that the response prediction performance is superior when the proposed 3D spatial constraint is utilized.

Data Augmentation via Intervention: We study the impact of data augmentation by comparing the performance of two models trained with and without the data augmentation strategy. In particular, augmentation plays a vital role in learning-based solutions for various computer vision tasks. In this work, we utilize the concept of intervention to synthesize new training data. Note that we also use intervention for risk object identification. The last section in Table III showcases the significance of using augmented data, which cuts the perplexity by nearly half. The data augmentation strategy adds variations to the training set that improve the robustness of the driver behavior model.

C. Driver-Centric Risk Object Identification

1) Evaluation Setup: We evaluate DROID in the four reactive scenarios: Congestion; Crossing Pedestrian; Crossing Vehicle; and Parked Vehicle. We use accuracy (number of correct predictions over the number of samples) as the evaluation metric. A correct prediction is one that has an Intersection over Union (IoU) score between a selected box and a ground truth box that is larger than a predefined threshold. Similar to [129], [130], accuracies at IoU thresholds of 0.5 and 0.75 are reported. In addition, mean accuracy (mACC) is calculated by using IoU thresholds ranging from 0.5 to 0.95 (in increments of 0.05).

2) Evaluation: We compare the performance of DROID with the following baselines. The results are shown in Table IV.

Random Selection: Random selection randomly picks an object as the risk object from all the detections for a given frame. Note that the method does not process any visual information except by means of object detection. The method is used to contextualize the challenge of this task.

Driver Attention Prediction: Uses a pre-trained model [30] trained on the (Berkeley DeepDrive) BDD-A dataset to predict the driver’s gaze attention maps at each frame. We compute an average attention weight of every detected object region based on a predicted attention map. The risk object is the object with the advantage of future prediction. We empirically demonstrate the effectiveness of the TRN Head design.

| Model                     | Perplexity |
|---------------------------|------------|
| Intention Modeling        | 0.83       |
| Multi-head                | 0.72       |
| TRN Head                  | 0.69       |
| Different Graph            | 0.74       |
| Ego-Stuff Graph           | 0.80       |
| Ego-Thing Graph           | 0.69       |
| Spatial Modeling          | 0.73       |
| Appearance Relation       | 0.69       |
| Data Augmentation         | 0.69       |
| With Augmentation         | 0.37       |
the highest attention weight, indicating the driver’s gaze attends to this region. The model is trained with human gaze signals that are unavailable in HDD. The performance of this method is slightly better than Random Selection, as reported in the second row of Table IV. We observe that predicted attention maps tend to focus at a vanishing point. Note that this issue has been raised in [131], highlighting the problem as one of the challenges of imitating human gaze behavior.

Object-Level Attention Selector: The object-level attention driving model [119] is reformulated for DROID. The risk object is the object with the highest object-attention score.

Pixel-Level Attention: Kim et al. [96] propose a causality test to search for regions that influence the network’s output behavior. Note that region proposals are formed based on sampling predicted pixel-level attention maps. To identify a risk object, we replace the region proposal strategy used in [96] with object detection, and utilize the inferred pixel-level attention map to filter out detections with low attention values. In the experiments, we set the threshold at 0.002. The modification ensures a fair comparison, as region proposals obtained from [96] are not guaranteed to be an object entity. Note that the code of region proposal generation detailed in [96] is not publicly available.

We report favorable DROID performance over existing baselines [30], [96], [119] in Table IV. The results indicate the effectiveness of the proposed driver behavior model and causal inference for the task. In the next section, we perform ablation studies to examine the contributions of each part of our model. Notice that our evaluation protocol differs from [37]. In [37], the authors train four different models and test four scenarios independently, whereas a single driver behavior model is trained in this work.

Ablation Study: Three variations are studied to analyze their impacts on the performance of DROID: (1) architecture of the driver behavior model, (2) intention modeling and (3) training strategy. The results are summarized in Table V.

Architecture: The completed framework (GCN + TRN Head, reported in the last row of Table V) boosts the mAccs of GCN by 6.2%, 0.5%, 3.0% and 24.6% in four different scenarios, respectively. This architecture ranks first in three scenarios (Crossing Vehicle, Parked Vehicle, and Congestion). In most scenarios, using GCN architecture performs better than the pure I3D + LSTM model. These results show a similar trend in the performance of driver response prediction in Table II (Row 5 v.s. Row 1, and Row 6 v.s. Row 2). The results are aligned with the design of the causal inference-based framework, i.e., the identification of risk object(s) relies on the prediction of driver response when different objects are removed. A better driver response prediction improves the performance of risk object identification.

Intention Modeling: Multi-head and TRN head-based intention modelings improve the accuracy of identifying risk objects. This observation also aligns with common senses: the risk object varies depending on different navigation goals, i.e., intentions. In Section VII-B2, the TRN head-based approach achieves a better performance for the driver response prediction task than Multi-head. However, we do not observe the same phenomenon in the DROID task. It could be because the two tasks are not designed in a unified manner. Thus, the value of TRN-based modeling cannot be observed with DROID.

Training With Data Augmentation: We observe significant improvement in all scenarios with the proposed data augmentation strategy except for Crossing Pedestrian. The results indicate the effectiveness of the proposed training strategy. For Crossing Pedestrian, our conjecture is that vehicles are likely to be chosen as risk objects because of the natural imbalanced distribution in the training data. Note that the ratio of detected vehicles to pedestrians is approximately 17:1. Our model learns how to identify risk objects under traffic configurations (especially different vehicle configurations) so that the model performs favorably for scenarios that involve interacting with vehicles. In contrast, scenarios that involve interacting with pedestrians are less emphasized. To solve this problem, a possible solution is to perform a category-aware intervention so that a balanced distribution can be obtained.

In summary, with the proposed components, i.e., TRN Head, intention modeling, and training with data augmentation, we demonstrate state-of-the-art DROID performance. We observe a similarly improved DROID performance in driver response prediction, discussed in Section VII-B.

4) Correlation versus Causation: We study the importance of causal modeling for this task. Instead of using causal inference (called Causation) to identify the risk object, the risk object is the object sending the highest attention weight to Ego in Ego-Thing Graph. We call this method Correlation. In Table V, the second to the last row shows the results of Correlation. Our Causation approach significantly outperforms Correlation in all reactive scenarios. We empirically demonstrate the need of causal modeling for this task.

In Fig. 7, ground truth risk objects are enclosed in red bounding boxes, our Causation results are shown in green, and the Correlation predictions are shown in blue boxes. In addition, we provide a bird’s-eye-view pictorial illustration of scenes in the second row. Note that it depicts scene layouts, driver intention, and traffic participants’ intentions, with identified risk objects in green boxes. In Fig. 7(b), three crossing pedestrians with different intentions are depicted. Our Causation approach correctly identifies the left-hand side pedestrian as the risk object while the driver intends to turn left. While Correlation predicts the same result, our method is more explainable because the

| Model                        | Crossing Vehicle | Crossing Pedestrian | Parked Vehicle | Congestion |
|------------------------------|------------------|---------------------|----------------|------------|
| Random Selection             | 15.1             | 7.1                 | 6.4            | 5.5        |
| Driver’s Attention Prediction | 16.8             | 8.9                 | 10.0           | 21.3       |
| Object-level Attention       | 32.6             | 9.5                 | 22.6           | 40.7       |
| Pixel-level Attention        | 28.0             | 8.1                 | 15.6           | 35.0       |
| GCN (ours)                   | 27.5             | 13.4                | 26.0           | 51.1       |
| GCN + TRN Head (ours)        | 72.0             | 13.3                | 22.3           | 32.1       |
| GCN + TRN Head + Data Augmentation (ours) | 32.5 | 12.9 | 28.4 | 37.8 |

The methods with * are re-implemented by us to ensure the same backbone is used for fair comparisons. mAcc stands for mean accuracy, and the unit is %. The best and second best performances are shown in bold and underlined, respectively.
TABLE V
ABSTRACTION STUDY OF THE PROPOSED FRAMEWORK

| Driver Behavior Model | Data Augmentation | Causal Inference | Crossing Vehicle $\text{Acc}_0.5$ | Crossing Vehicle $\text{mAcc}$ | Crossing Pedestrian $\text{Acc}_0.5$ | Crossing Pedestrian $\text{mAcc}$ | Parked Vehicle $\text{Acc}_0.5$ | Parked Vehicle $\text{mAcc}$ | Congestion $\text{Acc}_0.5$ | Congestion $\text{mAcc}$ |
|-----------------------|-------------------|------------------|-------------------------------|-----------------|------------------------------|-------------------|-----------------------------|-----------------|------------------------|------------------|
| 3D + LSTM             | ☒                 | ✓                | 29.9                         | 29.9           | 26.3                         | 15.5              | 14.3                       | 12.4            | 33.1                    | 28.7            |
| GCN (ours)            | ☒                 | ✓                | 31.8                         | 31.5           | 27.5                         | 16.2              | 15.5                       | 13.6            | 32.4                    | 29.4            |
| GCN + Multi-head (ours) | ☒             | ✓                | 31.8                         | 31.8           | 28.0                         | 17.9              | 17.9                       | 14.6            | 32.4                    | 29.4            |
| GCN + TRN Head (ours) | ☒                 | ✓                | 33.1                         | 33.1           | 29.0                         | 16.7              | 16.7                       | 13.2            | 33.8                    | 30.2            |
| GCN + TRN Head (ours) | ✓                 | ✓                | 28.3                         | 28.0           | 25.0                         | 13.1              | 11.9                       | 9.6             | 22.1                    | 21.3            |
| GCN + TRN Head (ours) | ✓                 | ✓                | 37.0                         | 37.0           | 32.5                         | 15.5              | 15.5                       | 12.9            | 35.3                    | 31.6            |

The unit is %. The best and second best performances are shown in bold and underlined, respectively. Mean accuracy ($\text{mAcc}$) and accuracies at IoU thresholds of 0.5 ($\text{Acc}_{0.5}$) and 0.75 ($\text{Acc}_{0.75}$) are reported.

---

D. Application: Risk Assessment

Our framework is able to perform risk assessment when multiple risk objects exist. We visualize objects’ risk scores in Fig. 8 under different reactive scenarios. All detected objects are encased in bounding boxes with different colors. Their risk scores are shown in a bar chart with color. The risk score of an object is equivalent to the predicted confidence score of $\text{Go}$ after the object is removed. A higher confidence score of $\text{Go}$ means that the object has a higher chance of influencing driver behavior. We use a black horizontal line to indicate the predicted confidence score of $\text{Go}$ when we do not intervene in the input. In all these cases, confidence scores are smaller than 0.5, representing correct driver response prediction by the proposed model. Favorable risk assessment results are demonstrated. In particular, in Fig. 8(b), when the two pedestrians could potentially be risk objects, our framework assigns high risk scores to both and the pedestrian closer to the vehicle is rated with a higher risk score.

---

Due to iterative causal inference, our framework requires the same amount of computational time ($\sim 0.15$ s) for each iteration with one NVIDIA TITAN-XP card. If there are more than 10 objects in a scene, the computation time of risk would be more than 1.5 s. Therefore, a single-shot design for risk assessment that does not require iterative causal inference is in need to realize future real-time applications of DROID.

---

VIII. CONCLUSION

In this paper, we focus on subjective risk assessment and operationalize the assessment by predicting driver behavior changes and identifying the cause of changes. A new task called DROID, which uses egocentric video collected from front-facing cameras to identify object(s) influencing a driver’s behavior, given only the driver’s response as the supervision signal, is introduced. We formulate the task as a cause-effect problem. A novel two-stage framework inspired by the model of SA and causal inference is present. We also construct a dataset for DROID to evaluate the proposed system. Extensive quantitative and qualitative evaluations are conducted. Favorable performance compared with strong baselines is demonstrated. Future work can leverage road topology explicitly to improve driver intention prediction.
Additionally, single shot risk assessment for DROID would be interesting to explore for practical applications.

ACKNOWLEDGMENTS
A part of the work was done when C. Li was an intern and Y.-T. Chen was a research scientist at Honda Research Institute USA, San Jose, CA, USA.

REFERENCES
[1] World Health Organization, “Global status report on road safety 2018: Summary,” 2018.
[2] National Highway Traffic Safety Administration, “Human factors evaluation of level 2 and level 3 automated driving concepts,” 2014. [Online]. Available: https://www.nhtsa.gov/
[3] M. Minderhoud and P. H. Bovy, “Extended time-to-collision measures for road traffic safety assessment,” Accident Anal. Prevention, vol. 33, no. 1, pp. 89–97, 2001.
[4] M. Althoff, O. Stürsberg, and M. Buss, “Stochastic reachable sets of interacting traffic participants,” in Proc. IEEE Intell. Veh. Symp., 2008, pp. 1086–1092.
[5] S. Worrall, D. Orchansky, F. Masson, and E. Nebot, “Improving vehicle safety using context based detection of risk,” in Proc. IEEE Int. Conf. Intell. Transp. Syst., 2010, pp. 379–385.
[6] M. Althoff and A. Mergel, “Comparison of Markov chain abstraction and Monte Carlo simulation for the safety assessment of autonomous cars,” IEEE Trans. Intell. Transp. Syst., vol. 12, no. 4, pp. 1257–1247, Dec. 2011.
[7] D. Greene et al., “An efficient computational architecture for a collision early-warning system for vehicles, pedestrians, and bicyclists,” IEEE Trans. Intell. Transp. Syst., vol. 12, no. 4, pp. 942–953, Dec. 2011.
[8] S. Lefèvre, C. Laugier, and J. Ibañez-Guzmán, “Evaluating risk at road intersections by detecting conflicting intentions,” in Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst., 2012, pp. 4841–4846.
[9] A. Lawitzky, D. Althoff, C. F. Passenberg, G. Tänzmeister, D. Wollherr, and M. Buss, “Interactive scene prediction for automotive applications,” in Proc. IEEE Intell. Veh. Symp., 2013, pp. 1028–1033.
[10] S. Lefèvre, D. Vasquez, and C. Laugier, “A survey on motion prediction and risk assessment for intelligent vehicles,” ROBOMECH J., vol. 1, 2014, Art. no. 1.
[11] J. Hariyono, A. Shabbaz, L. Kurnianggoro, and K.-H. Jo, “Estimation of collision risk for improving driver’s safety,” in Proc. 42nd Annu. Conf. IEEE Ind. Electron. Soc., 2016, pp. 901–906.
[12] M. Roth, F. Flohr, and D. M. Gavrila, “Driver and pedestrian awareness-based collision risk analysis,” in Proc. IEEE Intell. Veh. Symp., 2016, pp. 454–459.
[13] A. Paigwar, E. Baranov, A. Renzaglia, C. Laugier, and A. Legay, “Probabilistic collision risk estimation for autonomous driving: Validation via statistical model checking,” in Proc. IEEE Intell. Veh. Symp., 2020, pp. 737–743.
[14] A. Chinea and M. Parent, “Risk assessment algorithms based on recursive neural networks,” in Proc. Int. Joint Conf. Neural Netw., 2007, pp. 1434–1440.
[15] F. D. Salim, S. W. Loke, A. Rakotomainarainy, B. Srinivasan, and S. Krishnaswamy, “Collision pattern modeling and real-time collision detection at road intersections,” in Proc. IEEE Intell. Transp. Syst. Conf., 2007, pp. 161–166.
[16] 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.
[17] T. Suzuki, H. Kataoka, Y. Aoki, and Y. Sato, “Anticipating traffic accidents with adaptive loss and large-scale incident DB,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2018, pp. 3521–3529.
[18] Y. Yao, M. Xu, Y. Wang, D. J. 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.
[19] T. You and B. Han, “Traffic accident benchmark for causality recognition,” in Proc. Eur. Conf. Comput. Vis., 2020, pp. 540–556.
[20] R. Herzog et al., “Spatio-temporal action graph networks,” in Proc. IEEE Int. Conf. Comput. Vis. Workshop, 2019, pp. 2347–2356.
[21] J. Fang, D. J. Xue, and H. Yu, “DADA: Driver attention prediction in driving accident scenarios,” IEEE Trans. Intell. Transp. Syst., vol. 23, no. 6, pp. 4959–4971, Jun. 2022.
[22] H. Cui et al., “DiversityGAN: Diversity-aware vehicle motion prediction via latent semantic sampling,” IEEE Robot. Automat. Lett., vol. 5, no. 4, pp. 5089–5096, Oct. 2020.
[23] R. Cui et al., “Deep kinematic models for kinematically feasible vehicle trajectory predictions,” IEEE Trans. Intell. Transp. Syst., vol. 20, no. 4, pp. 1330–1338, Jul. 2019.
[24] J. Y. Ke, Z. Fang, T. Mao, Z. Wang, and S. Deng, “Joint prediction for kinematic trajectories in vehicle-pedestrian mixed scenes,” in Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst., 2020, pp. 273–280.
[25] K.-H. Zeng, S.-H. Chou, F.-H. Chan, J. C. Niebles, and M. Sun, “Agent-centric risk assessment: Accident anticipation and risky region localization,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2017, pp. 1330–1338.
[26] S. Allotto, A. Palazzi, F. Solera, S. Calderara, and R. Cucchiara, “A critical review,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2021, pp. 1756–1764.
[27] S. Bae, E. Pakdamanian, I. Kim, L. Feng, V. Ordonez, and L. Barnes, “MEDIRL: Predicting the visual attention of drivers via maximum entropy deep inverse reinforcement learning,” in Proc. IEEE/CVF Int. Conf. Comput. Vis., 2021, pp. 13158–13168.
[28] J. Pearl, Causality, New York, NY, USA: Cambridge Univ. Press, 2009.
[29] M. Kokubun, H. Konishi, K. Higuchi, T. Kurahashi, Y. Umemura, and H. Nishi, “Quantitative assessment of driver’s risk perception using a simulator,” Int. J. Veh. Saf., vol. 1, no. 1–3, pp. 5–21, 2005.
[30] M. R. Endsley et al., “Theoretical underpinnings of situation awareness: A critical review,” Situation Awareness Anal. Meas., vol. 1, no. 1–1, pp. 3–21, 2000.
[31] M. Li, S. H. Chan, and Y.-T. Chen, “Who makes drivers stop? Towards driver-centric risk assessment: Risk object identification via causal inference,” in Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst., 2020, pp. 10711–10718.
[32] J. C. Hayward, “Near-miss determination through use of a scale of danger,” Pennsylvania State University University Park, 1972.
[33] H. Bi, Z. Fang, T. Mao, Z. Wang, and Z. Deng, “Joint prediction for kinematic trajectories in vehicle-pedestrian mixed scenes,” in Proc. IEEE Int. Conf. Comput. Vis., 2019, pp. 10382–10391.
[34] T. Salzmann, B. Ivanovic, P. Chakravarty, and M. Pavone, “Trajectron: Dynamically-feasible trajectory forecasting with heterogeneous data,” in Proc. Eur. Conf. Comput. Vis., 2020, pp. 683–700.
[35] H. Cui et al., “Deep kinematic models for kinematically feasible vehicle trajectory predictions,” in Proc. IEEE Int. Conf. Robot. Automat., 2020, pp. 10563–10569.
[36] A. Kawasaki and A. Seki, “Multimodal trajectory predictions for urban environments using geometric relationships between a vehicle and lanes,” in Proc. IEEE Int. Conf. Robot. Automat., 2020, pp. 9203–9209.
[37] R. Cheung et al., “DiversityGAN: Diversity-aware vehicle motion prediction via latent semantic sampling,” IEEE Robot. Automat. Lett., vol. 5, no. 4, pp. 5089–5096, Oct. 2020.
[38] K. Fang et al., “It is not the journey but the destination: Endpoint conditioned trajectory prediction,” in Proc. Eur. Conf. Comput. Vis., 2020, pp. 759–776.
[39] B. Liu et al., “Spatiotemporal relationship reasoning for pedestrian intent prediction,” IEEE Robot. Automat. Lett., vol. 5, no. 2, pp. 3485–3492, Apr. 2020.
[97] P. de Haan, D. Jayaraman, and S. Levine, “Causal confusion in imitation learning,” in Proc. IEEE Conf. Neural Inf. Process. Syst., 2019, pp. 11698–11709.

[98] M. R. Samimi, M. Bahari, S. Salehkaleybar, and A. Alahi, “Causal representation learning for autonomous driving,” 2021, arXiv:2112.03908.

[99] Y. Liu, R. Cadei, J. Schweizer, S. Bahmani, and A. Alahi, “Towards robust and adaptive motion forecasting: A causal representation perspective,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2022, pp. 17060–17071.

[100] V. Ramanishka, Y.-T. Chen, T. Misu, and K. Saenko, “Toward driving scene understanding: A dataset for learning driver behavior and causal reasoning,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2018, pp. 7699–7707.

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

[102] S. Rota Bulan, I. Porzi, and P. Kontschieder, “In-place activated BatchNorm for memory-optimized training of DNNs,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2018, pp. 5639–5647.

[103] D. A. Forsyth et al., “Finding pictures of objects in large collections of images,” in Proc. Object Representation Comput. Vis. II: ECCV ’96 Int. Workshop, Cambridge, UK, Apr. 13–14, 1996, pp. 355–360.

[104] G. Heitz and D. Koller, “Learning spatial context: Using stuff to find things,” in Proc. Eur. Conf. Comput. Vis., 2008, pp. 30–43.

[105] H. Caesar, J. Uijlings, and V. Ferrari, “COCO-stuff: Thing and stuff classes in context,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2018, pp. 1209–1218.

[106] R. Ranftl, K. Lasinger, D. Hafner, K. Schindler, and V. Kolm, “Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 44, no. 3, pp. 1623–1637, Mar. 2022.

[107] C. Li, Y. Meng, S. H. Chan, and Y.-T. Chen, “Learning 3D-aware egocentric spatial-temporal interaction via graph convolutional networks,” in Proc. IEEE Int. Conf. Robot. Automat., 2020, pp. 8418–8424.

[108] J. Wu, L. Wang, L. Wang, J. Guo, and G. Wu, “Learning actor relation graphs for group activity recognition,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2019, pp. 9956–9966.

[109] J. I. Ba, J. R. Kiros, and G. E. Hinton, “Layer normalization,” 2016, arXiv:1607.06450.

[110] A. Rahimpour, S. Martin, A. Tawari, and H. Qi, “Context aware road-user safety,” Ph.D. dissertation, INRIA, 2013.

[111] A. Tawari, P. Mallela, and S. Martin, “Learning to attend to salient targets in driving videos using fully convolutional RNN,” in Proc. IEEE Intell. Transp. Syst. Conf., 2018, pp. 3225–3232.

[112] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “ImageNet: A large-scale hierarchical image database,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2009, pp. 248–255.

[113] D. Wang, C. Devin, Q.-Z. Cai, P. Krähenbühl, and T. Darrell, “Monocular plan view networks for autonomous driving,” in Proc. IEEE Int. Conf. Intell. Robots Syst., 2019, pp. 2876–2883.

[114] S. Lefèvre, C. Laugier, and J. Ibañez-Guzmán, “Intention-aware risk & paths to long term human trajectory forecasting,” in Proc. IEEE Int. Conf. Comput. Vis., 2021, pp. 15213–15222.

[115] P. de Haan, D. Jayaraman, and S. Levine, “Causal confusion in imitation learning,” in Proc. IEEE Conf. Neural Inf. Process. Syst., 2019, pp. 11698–11709.

[116] M. R. Samimi, M. Bahari, S. Salehkaleybar, and A. Alahi, “Causal representation learning for autonomous driving,” 2021, arXiv:2112.03908.

[117] Y. Hagmayer, S. A. Sloman, D. A. Lagnado, and M. R. Waldmann, “Causality and the construction of mental representations,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2022, pp. 15213–15222.

[118] D. Wang, C. Devin, Q.-Z. Cai, P. Krähenbühl, and T. Darrell, “Monocular plan view networks for autonomous driving,” in Proc. IEEE Int. Conf. Intell. Robots Syst., 2019, pp. 2876–2883.

[119] R. De Geest, Z. Gavves, A. Ghodrati, L. Li, C. Snoek, and T. Tuytelaars, “Online action detection,” in Proc. Comput. Vis.–ECCV 2016: 14th Eur. Conf., Amstderdam, The Netherlands, Oct. 11-14, 2016, pp. 269–284.

[120] M. CaIaMala, Y. An, H. Girase, and J. Malik, “From goals, waypoints & paths to long term human trajectory forecasting,” in Proc. IEEE Int. Conf. Comput. Vis., 2021, pp. 15213–15222.

[121] T.-Y. Lin et al., “Microsoft COCO: Common objects in context,” in Proc. Comput. Vis.–ECCV 2014: 13th Eur. Conf., Zurich, Switzerland, Sep. 6-12, 2014, pp. 740–755.

[122] D. Wang, C. Devin, Q.-Z. Cai, P. Krähenbühl, and T. Darrell, “Monocular plan view networks for autonomous driving,” in Proc. IEEE Int. Conf. Intell. Robots Syst., 2019, pp. 2876–2883.

[123] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “ImageNet: A large-scale hierarchical image database,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2009, pp. 248–255.

[124] D. Wang, C. Devin, Q.-Z. Cai, P. Krähenbühl, and T. Darrell, “Monocular plan view networks for autonomous driving,” in Proc. IEEE Int. Conf. Intell. Robots Syst., 2019, pp. 2876–2883.

[125] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” 2014, arXiv:1412.6980.

[126] D. Wang, C. Devin, Q.-Z. Cai, P. Krähenbühl, and T. Darrell, “Monocular plan view networks for autonomous driving,” in Proc. IEEE Int. Conf. Intell. Robots Syst., 2019, pp. 2876–2883.

[127] R. De Geest, Z. Gavves, A. Ghodrati, L. Li, C. Snoek, and T. Tuytelaars, “Online action detection,” in Proc. Comput. Vis.–ECCV 2016: 14th Eur. Conf., Amsterd...