2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 25-29, 2020, Las Vegas, NV, USA (Virtual)

Who Make Drivers Stop? Towards Driver-centric Risk Assessment: Risk Object Identification via Causal Inference

Chengxi Li[1] Stanley H. Chan[1] Yi-Ting Chen[2]

***Abstract*— A significant amount of people die in road accidents due to driver errors. To reduce fatalities, developing intelligent driving systems assisting drivers to identify potential risks is in an urgent need. Risky situations are generally defined based on collision prediction in the existing works. However, collision is only a source of potential risks, and a more generic definition is required. In this work, we propose a novel driver-centric definition of risk, i.e., objects influencing drivers' behavior are risky. A new task called risk object identification is introduced. We formulate the task as the cause-effect problem and present a novel two-stage risk object identification framework based on causal inference with the proposed object-level manipulable driving model. We demonstrate favorable performance on risk object identification compared with strong baselines on the Honda Research Institute Driving Dataset (HDD). Our framework achieves a substantial average performance boost over a strong baseline by 7.5%.**

I. INTRODUCTION

More than 1.3 million people die in road accidents worldwide every year, approximately 3,700 people per day [34]. Car accident deaths are the 1^{st} leading causes of death, excluding health illness. A massive number of car accident fatalities are due to driver errors such as the lack of awareness [1]. To reduce the fatality rate, developing intelligent driving systems assisting drivers to identify potential risks is in an urgent need. The identification of potential risks has been studied extensively in the risk assessment literature [20]. At the core of risk assessment is the definition of risk. In the context of intelligent vehicles, the risk is generally defined based on collision prediction. While the definition is widely used, road collision is only a source of potential hazards in driving [20].

In this paper, we propose a driver-centric definition of risk, i.e., *objects influencing drivers' behavior are risky*. Imagine driving in a scenario shown in Fig. 1 that we are planning to pass the intersection while yielding to a crossing pedestrian. Without the presence of the pedestrian, we would have passed the intersection without stopping. In other words, if we did not slow down or stop, a dangerous situation would occur. During our daily driving, we frequently interact with different traffic participants under diverse configurations. These reactive scenarios are commonly encountered than collisions. Thus, we believe that the proposed definition captures various learning signals for assessing risk.

C. Li[1] and S. H. Chan[1] are with Department of Electrical and Computer Engineering, Purdue University, West Lafayette, IN, USA. Y.-T. Chen[2] is with Honda Research Institute USA, San Jose, CA, USA.

The work was done when C. Li was an intern at Honda Research Institute, USA.

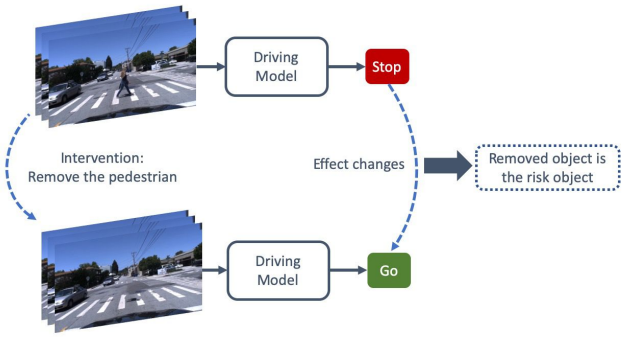


Fig. 1: A two-stage risk object identification framework. Intervening the input by removing the pedestrian changes the driver's behavior (effect) from 'Stop' to 'Go', indicating the removed object is the risk object (cause) for 'Stop'.

A natural question arises: *who makes drivers stop?* We propose a new task called *risk object identification*, which aims to identify the object influencing drivers' behavior. The proposed task can be approached via two existing methodologies: (1) supervised learning algorithms that learn to localize risky regions [39], recognize important objects [9], and imitate drivers' gaze behavior [3], and (2) salient regions/objects identification via self-attention mechanisms in end-to-end networks [16], [32].

In the first category, risky regions and important objects are identified by formulating as a two-class object detection problem. Specifically, object-object interaction [39], [9], object-environment interaction [39], object motion trajectory [39], [9] are designed to enable models to identify risky regions or important objects. Alternatively, objects influencing drivers' behavior can be obtained via predicting pixel-level driver's attention learned by imitating human gaze behavior [3], [35], [31]. While promising performances have been shown, labeling risky regions or important objects for training two-class object detectors requires extensive human-labeled annotations. Human gaze behavior is intrinsically noisy, and fixations may not directly associate with objects influencing drivers' behavior.

For the second strategy, objects influencing behaviors of drivers can be formulated as selecting regions/objects with high activations in visual attention heat maps learned from end-to-end driving models [16], [32]. Kim and Canny [16] proposed a two-stage framework. In the first stage, a visual attention mechanism is designed to highlight image regions that influence the end-to-end driving model's output. A

causal filtering step as the second stage is applied to determine the regions influencing the network's behavior. Wang et al. [32] incorporated an object-level attention mechanism into end-to-end driving models to increase robustness and interpretability. Both attention mechanisms are trained to optimize task-dependent objective functions. However, there is no guarantee that networks will attend the regions/objects that influence drivers' behavior. It is worth noting that the issue 'causal misidentification' is discussed in training end-to-end driving models [8].

To address the issues mentioned above, we formulate the definition as the cause-effect problem [26] and propose a novel risk object identification framework. The core concept is depicted in Fig. 1. First, an object-level manipulable driving model is learned to predict drivers' behavior. In this work, we simplify the possible driver behaviors to be 'Go' or 'Stop.' Note that modeling of more fine-grained drivers' behavior is our future work. Second, given a 'Stop' prediction (i.e., driver behavior is influenced by objects), we intervene 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 trained driving model is used to predict the corresponding driver behavior. Note that we assume that the cause of driver behavioral change is vehicles or pedestrians in this work. The object causes the maximum effect change is the one that influences drivers' behavior. We benchmark the proposed framework on the Honda Research Institute Driving Dataset (HDD) [27]. Experimental results show that the proposed framework achieves favorable risk object identification performance compared with strong baselines, both quantitatively and qualitatively. Furthermore, extensive ablation studies are conducted to justify our design choices.

The contributions of this work are summarized as follows. First, we propose a novel driver-centric definition of risk, i.e., objects influencing drivers' behavior are risky, and a new task called risk object identification is introduced. Second, we formulate the task as the cause-effect problem and propose a novel two-stage risk object identification framework based on causal inference with the proposed object-level manipulable driving model. Third, we demonstrate favorable performance on risk object identification in comparison with strong baselines on the HDD dataset [27].

II. Related Work

A. Risk Assessment

Living agents can assess risk for decision making. Lefèvre et al. [20] surveyed existing methods for motion prediction and risk assessment in the context of intelligent vehicles. A popular risk assessment methodology is to predict all possible colliding future trajectories. While many works follow the direction, Lefèvre et al. [19] defined the computation of risk as to the probability that expectation and intention do not match. The paradigm is very closed to the proposed definition of risk, i.e., objects influencing drivers' behavior, the underlying risk object identification process is different. In [19], a risk object is identified by computing the 'hazard probability' based on the same computation of risk. We recognize the risk object by simulating the causal effect by removing an object using end-to-end driving models.

B. Causal Confusion in End-to-end Driving Models

Recent successes [5], [36] demonstrated that a driving policy can be learned in a supervised manner from human demonstration [6], [32], [17]. Additionally, recent driving datasets [27], [38] with high-quality drivers' demonstration, enable training driving models under different real-world scenarios. However, the issue of causal confusion in training end-to-end driving models is raised in [7], [11], [8]. Haan et al. [8] proposed incorporating the concept of functional causal models [26] into imitation learning to address the issue of 'causal misidentification.' In [11], they overcame the causal misidentification issue by adding noises to inputs.

Our work is complementary to [11], [8]. Specifically, the focus of [11], [8] is to improve the robustness of driving models. Instead, our proposed driving model is to enable intervention for risk object identification. We believe the two lines of work should be studied jointly to obtain robust driving models with explicit reasoning mechanisms.

III. Method

We formulate the risk object identification problem as the cause-effect problem [26]. Specifically, we leverage and realize the concept of causal inference in a two-stage framework to identify the cause (i.e., the object) of an effect (i.e., driver behavioral change) via the proposed object-level manipulable driving model. We discuss the methodology of the proposed framework in the following.

A. Object-level Manipulable Driving Model

To realize causal inference for risk object identification, we propose an object-level manipulable driving model with the two properties. First, the driving model should be able to predict the corresponding driver behavior in an intervened scenario, i.e., an object is removed from historical observations. Second, the driving model needs to predict driver behavior while interacting with traffic participants. To realize the properties, we use *partial convolutional layers* [23] instead of standard convolutional layers. 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 partial convolutional layer takes two inputs, i.e., an RGB frame and corresponding one-channel binary mask. The value of pixels within a mask is set to be 1 by default. While training the driving model with an intervention (Section III-B) and perform causal inference for risk object identification (Section III-C), we set the pixels within as elected object to be 0.

To obtain representations of objects, Mask R-CNN [12], and Deep SORT [33] is applied to detect and track every object. RoIAlign [12] is employed to extract the corresponding object representation. The ego's representation is extracted via average pooling the features extracted from the partial convolution networks. Here we use 'ego' to denote

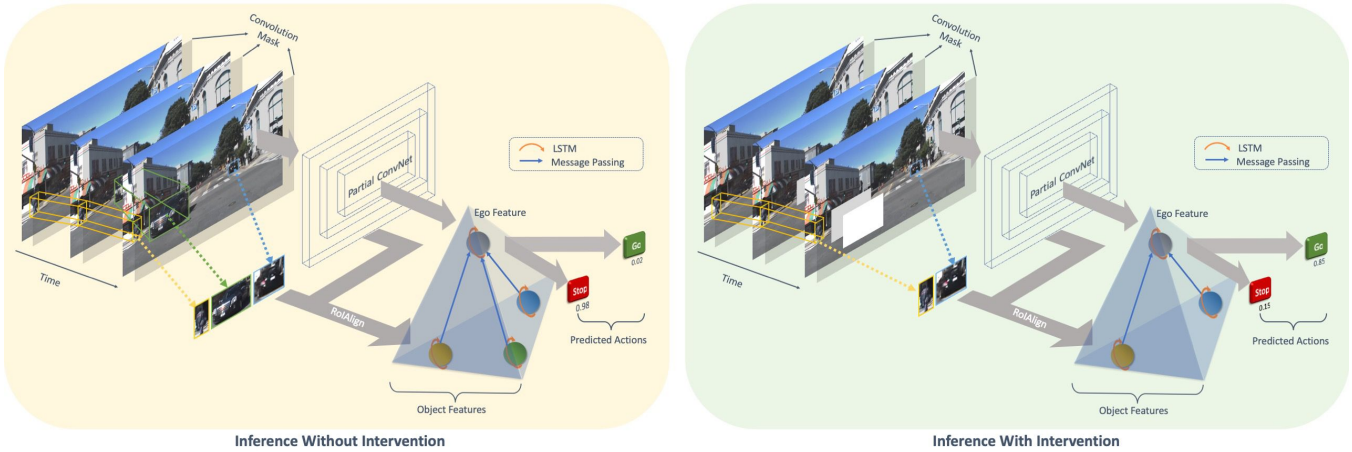


Fig. 2: An overview of our framework. The right and left figures show the inference process with and without intervention, respectively. Both employ the same driving model to output the predicted driver behavior. The inputs to the driving model include a sequence of RGB frames, a sequence of binary masks, and object tracklets. Partial convolution and average pooling are employed to obtain the ego features, while object features are extracted by RoIAlign. Each feature is modeled temporally and then propagates information to form a scene representation for final prediction. On the right, the input is intervened at an object level by masking the selected object on the convolution mask and removing it from the tracklets. For example, we remove the car in the green box, and the driving model returns a high confidence score of 'Go.'

the representation of a driver. We use a long short-term memory(LSTM) module [13] to model the ego and objects' temporal dynamics.

Motivated by [32] and [21], modeling interaction is essential for driver behavior modeling. We model interactions between the ego and objects via the following message passing mechanism,

$$g = h_e \oplus \frac{1}{N}(\sum_{i=1}^{N} h_i) \tag{1}$$

where g is defined as the aggregated representation. The ego's representation h_e is obtained after temporal modeling and $h_o = \{h_i, h_2, \cdots, h_N\}$ are representations of N objects. $\oplus$ indicates a concatenation operation. To manipulate the representation at the object level, we set the pixel value of the binary mask to 0 at the selected object's location. The mask influences representations extracted from the partial convolution networks and disconnect the message of the selected object from the rest of the objects and the ego. This representation g is passed through fully connected layers to obtain the final classification of driver behavior. In this work, we categorize driver behavior to be 'Go' and 'Stop'. An overview of the proposed driving model architecture is shown in Fig. 2.

B. Training with Intervention

We cluster and label the training samples into two categories — (1) 'Go': the ego vehicle moves without stopping/yielding/deviating and (2)'Stop': the ego vehicle stops, yields, or deviates for objects. We use the behavioral change between these two states to reason the risk object that influences driver behavior. It is worth noting that this is the only supervision signal. The performance of driving models can be improved by training with samples with different traffic configurations [36]. Due to the limited real-world human driver demonstrations, we design a training strategy that utilizes the concept of intervention from causal inference [26] to improve the robustness of the driving model. Specifically, we create new configurations based on a simple yet effective notion, i.e., removing non-causal objects does not affect driver behavior. For instance, in the 'Go' scenario, the ego vehicle goes straight and passes an intersection while pedestrians on the sidewalk. It is reasonable to assume if a pedestrian was not present, the behavior of the ego vehicle is the same.

This strategy is only applicable to the first category. In the second category, we need to know the causal object to remove non-casual objects. Intensive labeling of risk objects' locations is required. Note that this contradicts the spirit of the proposed weakly supervised method. Moreover, even if the annotations of causal objects are given, we cannot intervene on the causal object and simply assume the corresponding driver behavior to 'Go' because traffic situations are inherently complicated, making the intervened diver behavior unclear. For instance, imagine that under a congestion circumstance where the ego vehicle stops for the front vehicle at an intersection, while the traffic light shows red. In such a case, the front vehicle is labeled as the risk object (cause). However, removing the front vehicle does not necessarily change the ego vehicle's behavior because of the red light.

Algorithm 1 provides the pseudo-code of the proposed training process. For training samples in the first category, we randomly select one object k to intervene. Based on detection and tracking, a one-channel binary mask is generated and set

Algorithm 1: Training Driving Model with Intervention

T: Number of frames
N: Number of objects in the given tracklet list
h_e: Hidden states of ego in LSTM module
h_o: Hidden states of objects in LSTM Module $h_o := \{h_1, h_2, \cdots, h_N\}$
A: Ground truth driver behavior (either 'GO' or 'STOP')
Input: A sequence of RGB frames $I := \{I_1, I_2, \cdots, I_T\}$
Output: Confidence score of driver behaviors s^{go}, s^{stop}

$O := \texttt{DetectionAndTracking}(I)$
$:= \{O_1, O_2, \cdots, O_N\}$ // List of tracklets
if A is 'GO' and $N > 1$ **then**
  // Randomly choose one object to remove
  $k := \texttt{RandomSelect}(N)$
else
  k is empty
end if
// Mask out the region of object k on each mask frame
$M := \texttt{MaskGenerator}(I, O_k)$
// Remove the object k from the tracklet list
$O' = O - \{O_k\}$
$s^{go}, s^{stop} := \texttt{DrivingModel}(I, M, O')$ //Defined as below
return s^{go}, s^{stop}

function DRIVINGMODEL(I, M, O)
  for $t \in \{1, 2, \cdots, T\}$ **do**
    $e_t := \texttt{EgoFeature}(I_t, M_t)$
    $h_e := \texttt{LSTM}(e_t, h_e)$
    for $O_i \in O$ **do**
      $f_i^t := \texttt{RoIAlign}(o_i^t)$ // Object Features (o_i^t is i-th object's bounding box at time t)
      $h_i := \texttt{LSTM}(f_i^t, h_i)$
    end for
  end for
  $g := \texttt{MessagePassing}(h_e, h_o)$
  $s^{go}, s^{stop} := \texttt{ActionClassifier}(g)$
  return s^{go}, s^{stop}
end function

the pixels' value within the k-th object's region to be 0.

$$M_t(i,j) = \begin{cases} 0, & \text{if } (i,j) \text{ in region } o_k^t \\ 1, & \text{otherwise} \end{cases} \quad (2)$$

where M_t denotes the generated mask at time t, o_k^t is the bounding box of the k-th object at time t, and (i, j) is the pixel coordinate. This mask and the corresponding RGB frame will be the inputs to the partial convolutional networks. Notice that the k-th object is discarded from the tracklet list before feeding into the driving model.

C. Causal Inference for Risk Object Identification

The concept of causal inference is utilized for risk object identification. Specifically, given video frames and tracklets, the masks of a tracklet and the corresponding video frames are processed by the trained driving model. The driving model predicts the confidence score of 'Go' and 'Stop.' We select the object with the highest confidence score of 'Go', indicating this object causes the most substantial driver behavioral change as the risk object. Algorithm 2 describes the process.

Algorithm 2: Inference for Risk Object Identification

T: Number of frames
N: Number of objects
Input: A sequence of RGB frames $I := \{I_1, I_2, \cdots, I_T\}$ where the ego car stops
Output: Risk object ID

$O := \texttt{DetectionAndTracking}(I)$
$:= \{O_1, O_2, \cdots, O_N\}$// List of tracklets
for $O_k \in O$ **do**
  // Mask out the region of object k on each frame
  $M := \texttt{MaskGenerator}(I, O_k)$
  // Remove the object k from the tracklet list
  $O' = O - \{O_k\}$
  // Predict the action of ego car without object k
  $s_k^{go}, s_k^{stop} := \texttt{DrivingModel}(I, M, O')$
end for
return $\arg\max_k(s_k^{go})$

	# Training Frames	# Validation Frames	# Test Frames
Crossing Vehicle	18,696	5,652	311
Crossing Pedestrian	20,784	3,999	84
Parked Vehicle	11,484	3,537	136
Congestion	21,132	5,607	99

TABLE I: Statistics of train/val/test samples in HDD [27] used in our experiments.

IV. EXPERIMENTS

A. Dataset

We evaluate the proposed framework on the HDD dataset [27], a multisensory 104-hour naturalistic driving dataset providing a 4-layer representation of tactical driver behavior. The **Stimulus-driven action** layer includes behaviors such as 'stop' and 'deviate' and the **Cause** layer denotes the reasons for behavioral changes such as an oncoming vehicle. For example, while going straight, an oncoming vehicle causes the driver to stop. In total, the dataset has 6 **Cause** scenarios, i.e., *Stopping for Congestion*, *Stopping for Crossing Vehicle*, *Deviating for Parked Vehicle*, *Stopping for Pedestrian*, *Stopping for Sign*, and *Stopping for Red Light*. The first four scenarios are selected to evaluate the proposed risk object identification framework. We utilize the frame-level driver behavior label (i.e., 'Go' and 'Stop') to train our driving model.

The dataset consists of 137 sessions, and each session represents a navigation task performed by a driver. In [27],

Driving Model	Mask	Training with Intervention	Crossing Vehicle			Crossing Pedestrian			Parked Vehicle			Congestion		
			$Acc_{0.5}$	$Acc_{0.75}$	$mAcc$	$Acc_{0.5}$	$Acc_{0.75}$	$mAcc$	$Acc_{0.5}$	$Acc_{0.75}$	$mAcc$	$Acc_{0.5}$	$Acc_{0.75}$	$mAcc$
Vanilla CNN	RGB	✗	36.0	35.7	31.4	20.2	16.7	14.9	36.8	32.4	29.7	87.9	83.9	76.8
Partial CNN	RGB	✗	38.6	37.6	33.5	22.6	19.0	16.2	36.0	32.4	29.0	81.8	81.8	73.7
	RGB	✓	41.2	40.5	36.2	19.0	16.7	13.5	39.0	36.8	32.4	**94.9**	**91.0**	**82.6**
	Convolution	✗	38.6	37.6	33.6	22.6	17.9	16.2	36.8	33.1	29.5	88.9	84.8	78.0
	Convolution	✓	44.4	43.1	38.5	25.0	22.6	19.3	34.6	33.1	28.8	88.0	84.8	77.3
Partial CNN + Object	Convolution	✗	39.9	38.9	34.4	27.4	22.6	18.9	31.6	27.9	24.7	91.9	87.9	79.7
	Convolution	✓	**49.2**	**48.6**	**43.0**	**35.7**	**32.1**	**27.0**	**47.1**	**44.9**	**39.8**	92.9	88.9	81.0

TABLE II: Ablation studies. Results of risk object identification in four scenarios on the HDD. The unit is %. The best and second performances are shown in bold and underlined, respectively.

the authors split the dataset into training and testing sets according to the vehicle's geolocation. We use the same train/test data split as in [27] (i.e., 100 sessions for training and 37 sessions for testing). The dataset provides annotations of risk objects for a tiny portion of the dataset, making it infeasible to train a robust supervised learning-based two-class object detection model as in [39], [9]. The statistics of train/val/test samples are presented in TABLE I.

We use accuracy, i.e., the number of correct predictions over the number of ground truth samples as the evaluation metric. An accurate prediction is that an Intersection over Union (IoU) score is greater than a threshold. Like [22], [40], accuracy at IoU thresholds of 0.5 and 0.75 are reported, as well as mean accuracy $mACC$, which is the average accuracy at 10 IoU thresholds evenly distributed from 0.5to 0.95.

B. Implementation Details

We implement our framework in PyTorch [2] and perform all experiments on a system with Nvidia Quadro RTX 6000 graphics cards. The framework takes a sequence of frames with a resolution of 299×299 at 3 fps, and T is set to 3 in all the experiments, approximately 1s. The corresponding input mask maintains the same size as the input image. The convolutional backbone is a InceptionResnet-V2 [30], pre-trained on ImageNet [28] and is modified with partial convolution layers [23], [24]. A Detectron model [10] trained on MSCOCO [22] is used to generate object bounding boxes. RoIAlign extracts an object representation with a size of $20 \times 8 \times 8$ from the Conv2d_7b layer, which is then flattened into a 1280-dimensional vector.

We follow the same initialization strategy as in [37], i.e., the hidden state units is set to 512, and dropout keep probability [29] is set to 0.5 at hidden state connections LSTM. The aggregated feature g concatenated from ego features and object features is a 1-D vector with 1024 channels. Similar to [16], the output sizes of 3 fully-connected layers before the final binary classifier are 100, 50, and 10, respectively. The network is trained end-to-end for 10 epochs with batch size set to 16. We use Adam [18] optimizer with default parameters, learning rate 0.0005, and weight decay 0.0005.

C. Ablation Studies

We evaluate three aspects of our framework in TABLE II.

Architecture of the Driving Model. Our proposed driving model uses features from CNN features and object features. For CNN features, we test two backbone features, i.e., vanilla convolution and partial convolution.

Intervention Mask. The input to Partial CNN includes an extra mask, offering two options to intervene an image. We either input an RGB image with selected region masked out or feed in a binary mask with the selected region set to 0. We denote the two ways of intervention as 'RGB mask' and 'Convolution mask' in TABLE II.

Training with Intervention. To understand the intervention's effect on training driving models, i.e., using intervention to generate more traffic configurations, we explore two experimental settings — training with and without intervention. Note that we always use a convolution mask to remove selected objects while using the training with the intervention strategy for Partial CNN. In the Partial CNN + Object model, we additionally remove the selected object features during message passing.

Our final framework (last row in TABLE II) boosts the $mACC$ by 11.6%, 13.5%, 11%, and 7.3%, respectively, compared with the lowest accuracies. It ranks first in three scenarios (Crossing Vehicle, Crossing Pedestrian and Parked Vehicle) and second in Congestion case. Training with intervention always leads to an increase in accuracy when object-level information is modeled. However, it does not necessarily help the performance when the driving model is modeled using only partial CNN.

Regarding the intervened mask types, we observe that in Crossing Vehicle and Crossing Pedestrian scenarios, intervening with a convolution mask achieves higher accuracy than an RGB mask. However, in the other two scenarios, it is the opposite. We conjecture that, when the ego vehicle deviates for parked cars or stops for congestion, the target risk object is salient and closed to the ego vehicle. Under such a circumstance, inputting a masked RGB frame could be sufficient for the driving model to predict correct driver behavior. Therefore, the hallucination effect of partial convolutional layers is negligible.

D. Quantitative Evaluation

There is no existing work being developed for risk object identification. Therefore, we re-implement three approaches [16], [32], [35], and follow their design philosophy

Method	*mAcc*			
	Crossing Vehicle	Crossing Pedestrian	Parked Vehicle	Congestion
Random Selection	15.1	7.1	6.4	5.5
Driver's Attention Prediction * [35]	16.8	8.9	10.0	21.3
Object-level Attention Selector * [32]	36.5	21.2	20.1	8.9
Pixel-level Attention + Causality Test * [16]	<u>41.9</u>	<u>21.5</u>	<u>34.6</u>	<u>62.7</u>
Ours	**43.0**	**27.0**	**39.8**	**81.0**

TABLE III: Comparison with baseline methods. The methods with * are our re-implementation. The unit is %. The best and second performances are shown in bold and underlined, respectively.

to select important/risk objects. The comparison with our method is shown in TABLE III.

Random Selection and **Driver Attention Prediction**. The results of these two methods are not directly comparable to ours, and we show the results to provide an essential measure of the difficulty of this task. We first propose a naive baseline, randomly selecting one object as the risk object from all the detections for a given frame. This method does not process any visual information. In TABLE III row 2, we use a pre-trained model [35] to predict the driver's gaze attention maps at each frame. We compute the average attention weight of every detected object region. The risk object is one with the highest attention weight, indicating the driver's gaze attends this region. The model is trained with the human gaze as supervision, which is unavailable in the HDD dataset. Thus, we use the model pre-trained on the BDD-A dataset [35]. The performance of this method is slightly better than **Random Selection**. By visualizing the predicted attention map, we discover that the heated spots tend to cluster around the vanishing point. Note that the issue has been raised in [4]. The reference highlights that this is one of the challenges of imitating human gaze behavior.

Object-level Attention Selector. Wang et al. [32] designed an object-centric driving model by learning object-level attention weights, which can be further used as an object selector for identifying risk objects. Motivated by their design, we modify the message passing in our driving model to be object-level attention and re-train our model. We evaluate the accuracy in four scenarios based on the selected object with the highest attention weight.

Pixel-level Attention + Causality Test. Kim et al. [16] proposed a causality test to search for regions that influence the network's output. They utilized the pixel-level attention map learned from an end-to-end driving model to sample particles conditioned on the attention value over an input image. The sampled particles are clustered to produce a convex hull further to form region proposals. Each convex hull is masked out on an RGB image, and the image is sent to the trained model to perform a causality test, iteratively. The region, which leads to the maximum decrease of prediction performance, will be the risk object. For a fair comparison, we replace the region proposals with object detections and utilize the pixel-level attention to filter out detections with low attention values. In the experiments, we set the threshold at 0.002.

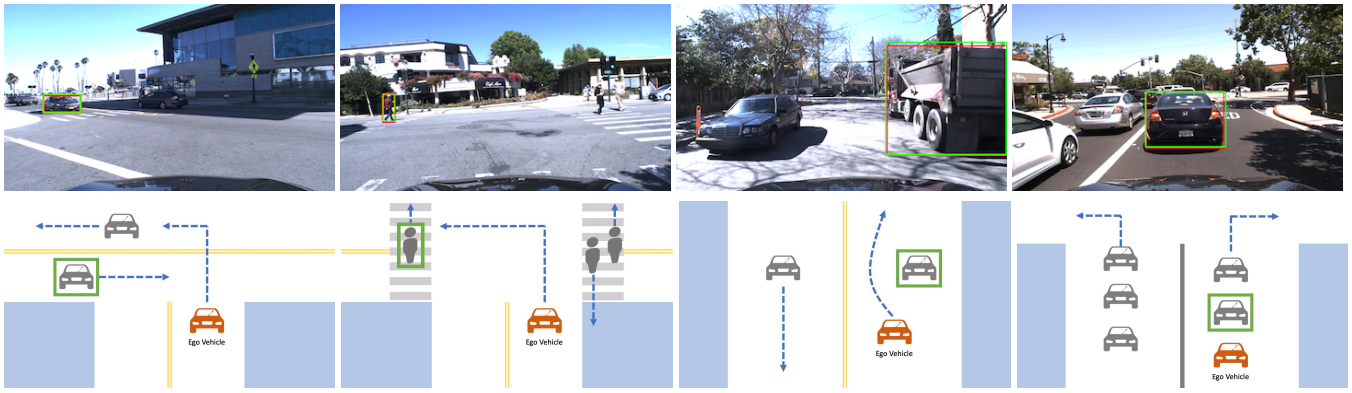
The reason for this modification is that, compared with our detections generated from the state-of-the-art object detection algorithm, the region proposals obtained from pixel-level attention are not guaranteed to be an object, which results in an extremely low IoU and accuracy. Additionally, the code of generating region proposals is not publicly available. This method's performance is the closest to our results since the causality test is similar to our inference with intervention Our performance is better because our driving model is manipulated at the object-level and is trained with the intervention strategy for robustness.

E. Visualization

We visualize the qualitative results of our method in the four scenarios. In Fig. 3, ground truth risk objects are enclosed in red bounding boxes, and our predicted results are colored in green. To better visualize the interactions between traffic participants, we provide a birds-eye-view (BEV) pictorial illustration in the second row. The BEV figures depict the scene layout, the intention of the ego vehicle and other traffic participants', and the identified risk object in the green box. In Fig. 3 (b), three pedestrians are presented in the scene, crossing the road towards different destinations. Our approach correctly tags the left-hand side pedestrian to be the risk object. The ego vehicle intends to take a left turn, indicating that our driving model can implicitly anticipate the ego vehicle's intention based on historical observations.

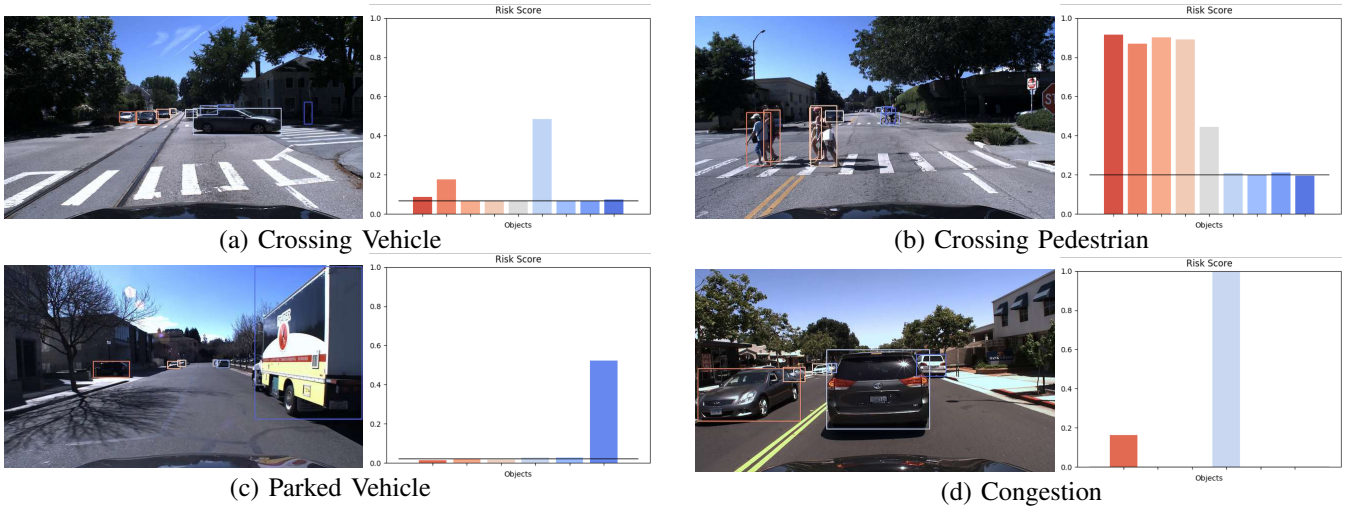
In addition to risk object identification, our framework can also assess the risk of every object in the scene. We visualize the corresponding results in Fig. 4. All detected objects are localized using colored bounding boxes, and the related risk scores are in a bar chart with related colors. The risk score of an object is defined as the predicted confidence score of 'Go' when the object is masked using the proposed risk object identification framework. A higher score of 'Go' represents a higher possibility that the object influences the ego vehicle behavior. We use a black horizontal line to indicate the predicted confidence score of 'Go' without any interventions on the input. If the score of 'Go' is less than 0.5, the driver behavior is classified as 'Stop.' Our framework generates satisfactory risk assessment results qualitatively.

In Fig. 4 (b), when multiple risk objects (a group of people) exist, our framework assigns high-risk scores to all pedestrians. The result seems correct at first glance. However, even if one of the pedestrians was not present, the ego vehicle should have stopped. We conjecture that the partial convolutional operation not only hallucinates the removed area but also affects the surrounding regions due to the growing receptive field as networks go deeper. As pedestrians are adjacent, removing one person by partial convolutions may dilute the surrounding ones and return high-risk scores. To verify the conjecture, we manually inpainting the image by masking every pedestrian iteratively and feed the inpainted image to the same driving model without applying partial convolutions. The corresponding results are shown in Fig. 5. A lower risk score of each pedestrian is observed that



(a) Crossing Vehicle (b) Crossing Pedestrian (c) Parked Vehicle (d) Congestion

Fig. 3: Risk object identification results on sample scenarios selected from the HDD dataset. The top row shows an egocentric view where green boxes are the predicted risk objects, and ground truth ones are in red. A birds-eye-view representation is presented in the bottom row, providing information including scene layout, intentions of traffic participants, and the ego vehicle. The objects in green frames are the risk objects detected by our method.



(a) Crossing Vehicle (b) Crossing Pedestrian

(c) Parked Vehicle (d) Congestion

Fig. 4: Risk assessment results on sample scenarios selected from the HDD dataset. On the left, all detected objects are localized in colored bounding boxes. The risk score of each object is depicted in a bar chart on the right. The color of each bar is one-to-one matched to the bounding box. A black horizontal line is used to indicate the predicted 'Go' score without any interventions.

aligns with our intuition. It also proves our conjecture that partial convolutional operations influence the behavior of the proposed driving model while more studies are needed.

F. Failure Cases

While our model shows the possibility to identify the intention of the ego vehicle (Fig. 3 (b)), there are situations that our driving model selects an incorrect risk object due to wrong intention prediction. In Fig. 6 (a), the ego vehicle plans to take a right turn and stops for the car in the red box. However, our framework selects the white pickup truck as the risk object because of the incorrect prediction of the ego vehicle due to ambiguous and historical cues. Additionally, in Fig. 6 (b), our driving model cannot distinguish which car will move first at a 4-way stop intersection and where it is going, resulting in a wrong selection. Hence, we believe explicitly modeling the intention of drivers, and other participants' in the driving model will render better inference results.

V. CONCLUSIONS

In this paper, we propose a novel driver-centric definition of risk, i.e., objects influencing drivers' behavior are risky. A new task called risk object identification is introduced. We formulate the task as the cause-effect problem and propose a novel two-stage risk object identification framework based on causal inference with the proposed object-level manipulable driving model. Favorable performance on risk object identification in comparison with strong base-lines is demonstrated on the HDD dataset. Extensive quantitative and qualitative evaluations are conducted. For future works, as highlighted in IV-F, explicit intention modeling of driver and traffic participants' will be beneficial. Additionally, a more sophisticated driver behavior modeling that considers

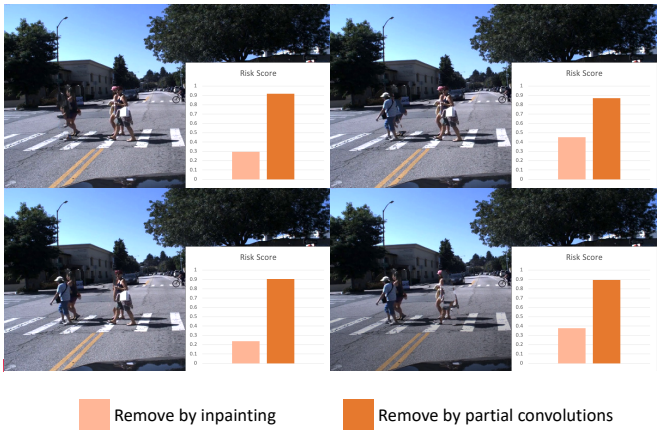


Fig. 5: An example of computed risk scores by using inpainted images and partial convolutions (our method).

(a) (b)

Fig. 6: Examples of failure cases. Our prediction is in green and ground truth is in red.

acceleration, brake, and steering are essential for reasoning the causal and effect. Furthermore, it will be valuable for practical applications to formulate the framework into a single-stage framework, as presented in [15], [25], [14], [8].

VI. Acknowledgement

The work is sponsored by Honda Research Institute USA.

References

[1] NHTSA. https://www.nhtsa.gov/research/. 1
[2] PyTorch. https://pytorch.org/. 5
[3] S. Alletto, A. Palazzi, F. Solera, S. Calderara, and R. Cucchiara. DR(eye)VE: A Dataset for Attention-based Tasks with Applications to Autonomous and Assisted Driving. In *CVPRW*, 2016. 1
[4] S. M. Ashish Tawari, Praneeta Mallela. Learning to Attend to Salient Targets in Driving Videos using Fully Convolutional RNN. In *ITSC*, 2018. 6
[5] M. Bojarski, D. D. Testa, D. Dworakowski, B. Firner, B. Flepp, P. Goyal, L. D. Jackel, M. Monfort, U. Muller, J. Zhang, X. Zhang, J. Zhao, and K. Zieba. End to End Learning for Self-Driving Cars. In *arXiv preprint arXiv:1604.07316*, 2016. 2
[6] F. Codevilla, M. Müller, A. Löpez, V. Koltun, and A. Dosovitskiy. End-to-end Driving via Conditional Imitation Learning. In *ICRA*, 2018. 2
[7] F. Codevilla, E. Santana, A. López, and A. Gaidon. Exploring the Limitations of Behavior Cloning for Autonomous Driving. In *arXiv preprint arXiv:1904.08980*, 2019. 2
[8] P. de Haan, D. Jayaraman, and S. Levine. Causal Confusion in Imitation Learning. In *NeurIPS*, 2019. 2, 8
[9] M. Gao, A. Tawari, and S. Martin. Goal-oriented Object Importance Estimation in On-road Driving Videos. In *ICRA*, 2019. 1, 5
[10] R. Girshick, I. Radosavovic, G. Gkioxari, P. Dollár, and K. He. Detectron. https://github.com/facebookresearch/detectron, 2018. 5
[11] J. Hawke, R. Shen, C. Gurau, S. Sharma, D. Reda, N. Nikolov, P. Mazur, S. Micklethwaite, N. Griffiths, A. Shah, and A. Kendall. Urban Driving with Conditional Imitation Learning. In *arXiv preprint arXiv:1912.00177*, 2019. 2
[12] K. He, G. Gkioxari, P. Dollar, and R. Girshickn. Mask R-CNN. In *CVPR*, 2017. 2
[13] S. Hochreiter and J. Schmidhuber. Long Short-term Memory. *Neural computation*, 1997. 3
[14] T. Jayram, T. Kornuta, V. Albouy, E. Sevgen, and A. Ozcan. Learning Multi-Step Spatio-Temporal Reasoning with Selective Attention Memory Network. In *NeurIPS*, 2019. 8
[15] N. R. Ke, O. Bilaniuk, A. Goyal, S. Bauer, H. Larochelle, C. Pal, and Y. Bengio. Learning Neural Causal Models from Unknown Interventions. In *arXiv preprint arXiv:1910.01075*, 2019. 8
[16] J. Kim and J. Canny. Interpretable Learning for Self-driving Cars by Visualizing Causal Attention. In *ICCV*, 2017. 1, 5, 6
[17] J. Kim, T. Misu, Y.-T. Chen, A. Tawari, and J. Canny. Grounding Human-to-vehicle Advice for Self-driving Vehicles. In *CVPR*, 2019. 2
[18] D. P. Kingma and J. Ba. Adam: A Method for Stochastic Optimization. In *arXiv preprint arXiv:1412.6980*, 2014. 5
[19] S. Lefèvre, C. Laugier, and J. Ibañez-Guzmàn. Evaluating Risk at Road Intersections by Detecting Conflicting Intentions. In *IROS*, 2012. 2
[20] S. Lefèvre, D. Vasquez, and C. Laugier. A Survey on Motion Prediction and Risk Assessment for Intelligent Vehicles. *ROBOMECH Journal*, 1:1, 2014. 1, 2
[21] C. Li, Y. Meng, S. H. Chan, and Y.-T. Chen. Learning 3D-aware Egocentric Spatial-Temporal Interaction via Graph Convolutional Networks. In *ICRA*, 2020. 3
[22] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft COCO: Common Objects in Context. In *ECCV*, 2014. 5
[23] G. Liu, a. K. J. S. Fitsum A. Reda, T.-C. Wang, A. Tao, and B. Catanzaro. Image Inpainting for Irregular Holes using Partial Convolutions. In *ECCV*, 2018. 2, 5
[24] G. Liu, K. J. Shih, T.-C. Wang, F. A. Reda, K. Sapra, Z. Yu, A. Tao, and B. Catanzaro. Partial Convolution based Padding. *arXiv preprint arXiv:1811.11718*, 2018. 5
[25] S. Nair, Y. Zhu, S. Savarese, and L. Fei-Fei. Causal Induction from Visual Observations for Goal Directed Tasks. In *NeurIPS 2019 Workshop on Causal Machine Learning*, 2019. 8
[26] J. Pearl. Causality. *Cambridge University Press*, 2009. 2, 3
[27] V. Ramanishka, Y.-T. Chen, T. Misu, and K. Saenko. Toward Driving Scene Understanding: A Dataset for Learning Driver Behavior and Causal Reasoning. In *CVPR*, 2018. 2, 4, 5
[28] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet LargeScale Visual Recognition Challenge. In *IJCV*, 2015. 5
[29] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *JMLR*, 2014. 5
[30] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alem. Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning. In *AAAI*, 2017. 5
[31] A. Tawari, P. Mallela, and S. Martin. Learning to Attend to Salient Targets in Driving Videos using Fully Convolutional RNN. In *ITSC*, 2018. 1
[32] D. Wang, C. Devin, Q.-Z. Cai, F. Yu, and T. Darrell. Deep Object Centric Policies for Autonomous Driving. In *ICRA*, 2019. 1, 2, 3, 5, 6
[33] N. Wojke, A. Bewley, and D. Paulus. Simple Online and Realtime Tracking with a Deep Association Metric. In *ICIP*, 2017. 2
[34] World Health Organization. Global status report on road safety 2018: Summary, 2018. 1
[35] Y. Xia, D. Zhang, J. Kim, and D. W. Ken Nakayama, Karl Zipser. Predicting Driver Attention in Critical Situations. In *ACCV*, 2018. 1, 5, 6
[36] H. Xu, Y. Gao, F. Yu, and T. Darrell. End-to-end Learning of Driving Models from Large-scale Video Datasets. In *CVPR*, 2017. 2, 3
[37] M. Xu, M. Gao, Y.-T. Chen, L. Davis, and D. Crandall. Temporal Recurrent Networks for Online Action Detection. In *ICCV*, 2019. 5
[38] F. Yu, W. Xian, Y. Chen, F. Liu, M. Liao, V. Madhavan, and T. Darrell. BDD100K: A Diverse Driving Video Database with Scalable Annotation Tooling. In *arXiv preprint arXiv:1805.04687*, 2018. 2
[39] 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 *CVPR*, 2017. 1, 5
[40] Z. Zhang, C. Yu, and D. Crandall. A Self Validation Network for Object-Level Human Attention Estimation. In *NeurIPS*, 2019. 5