Modeling Preemptive Behaviors for Uncommon Hazardous Situations From Demonstrations

Anonymous Author(s)
Affiliation
Address
email

Abstract

This paper presents a learning from demonstration approach to programming safe, autonomous behaviors for uncommon driving scenarios. Simulation is used to re-create a targeted driving situation, one containing a road-side hazard creating significant occlusion in an urban neighborhood, and collect optimal driving behaviors from 24 users. Paper employs a key-frame based approach combined with an algorithm to linearly combine models in order to extend the behavior to novel variations of the target situation. This approach is theoretically agnostic to the kind of LfD framework used for modeling data and our results suggest it generalizes well to variations containing additional number of hazards occurring in sequence. The linear combination algorithm is informed by analysis of driving data, which also suggests that decision making algorithms need to consider a trade-off between road-rules and immediate rewards to tackle some complex cases.

1 Introduction

There have been significant improvements in the field of autonomous driving [5, 7, 15, 13, 14, 12], however we do not currently see such vehicles on our roads. The technical reason is that Auto Vehicles (AV) are expected to demonstrate safety records superior to humans. Driving on urban roads in common and predictable situations can be considered a solved problem, however the real challenge of AV is handling unexpected situations while maintaining safety[11]. In this work, we focus on one such situation: when there is a risk of a previously unobserved pedestrian or object suddenly appearing from an occluded area (see figure 1). Rather than waiting for the hazard to emerge, AVs can potentially take preemptive actions to reduce risk of an accident once a hazard is sensed, for instance by steering farther away or reducing speed[10]. This paper studies how such behaviors could be learned from human demonstrations.

End-to-end models [5, 7] are data-driven and generally suffer from disproportionate quantities of various corner-cases in datasets, being incomplete in terms of safety guarantees. Another viable approach is to recreate the corner cases in simulation and use approaches like reinforcement learning (RL), inverse RL or learning from demonstration to model the behavior. RL however either requires an accurate model of the environment[15] or large amount of exploration[9] before figuring out how to avoid fatal scenarios, inverse RL on the other hand assumes similar underlying rewards for all demonstrations thus requiring large amount of perfect data to converge[16].

We use Learning from Demonstrations (LfD)[2, 3, 1] in this work because it inherently captures the human knowledge of reacting to obvious as well as emerging hazardous situations without placing any limiting assumptions on the behavior. We conduct our experiments in simulation, however our framework is also easily applicable to a real vehicle, if the data is sourced from such demonstrations.

This is a hypothesis driven exploratory study. We present our results on how the independent features of hazard affect behavior of drivers and also show algorithm generated constraints from

Submitted to 31st Conference on Neural Information Processing Systems (NIPS 2017). Do not distribute.

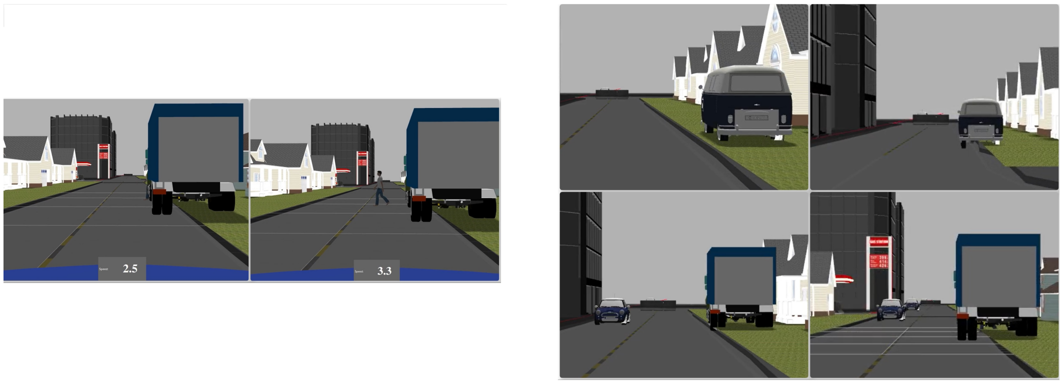

Figure 1: Depiction of Hazard and Variation of Size and Position in Simulation

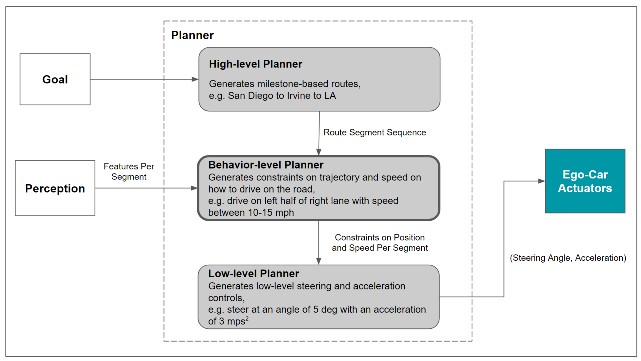

Figure 2: Schematic Diagram of System Architecture
This figure shows the three-tiered planning system as well as the data-flow between different modules relevant to this paper. The mid-level planner is highlighted to emphasize algorithm placement.

our trained models which learns variance of behavior over the recorded behaviors. We have also added a generalization algorithm to the constraint generator, agnostic of LfD model used, which linearly combines models depending upon the environment to generate constraints for out-of-training-distribution (OOTD) cases. Our results suggest that this approach captures enough details to generalize the solution for generating constraints given novel scenarios with multiple hazards in occurring in sequence with some or no overlap.

2 Approach

We started this work with a hypothesis that the size and closeness of the hazard have a significant correlation with the evasive behavior that it elicits in a human driver. By evasive behavior we mean, categorically, the extent of deviation from “normal" driving trajectory, how early on this behavior is triggered (D_{thresh}) and the change in speed. To test this hypothesis, we implemented a driving simulator using Gazebo [1] as can be seen in figure 1.

2.1 System Architecture

The system follows a funneled plan generation paradigm. The top layer provides a milestone-based plan using road networks to travel from start to goal position, which we will call the *route*. The middle layer generates speed, acceleration, trajectory, etc. constraints for the road-segments included in this route. The bottom-most layer consists of the actual low-level controllers of the autonomous car and generates optimized trajectories using the constraints from mid-level planner. Figure 2 is a schematic illustration of this architecture and how it interacts with other relevant modules.

[1]http://gazebosim.org

Our component sits in the middle layer and generates trajectory and speed constraints for the current segment of the route for the particular case of hazard occlusion. It depends on the perception module to provide it with the required state information (as described in section 3.1). Middle layer consists of many such modules which generate constraints and then the *constraint aggregator* combines the ones it deems relevant based on its scene understanding.

2.2 Study and Data Collection

Our driver population consisted of 24 people from Honda Research Institute, consisting of 3 female and 21 male drivers. More than 50% of the population had a driving experience of 10 years or plus and around 90% of the population drove an average of 0 to 2 hours per week. The users were provided with one pilot run to acclimate themselves with the sensitivity of the wheel and feel of the simulator before having them drive the controlled test cases. Our recorded feature set consisted of global measures namely: Location of the car, location of the hazard, heading of the car, speed of the car, size of the hazard, nature of the road: bidirectional or unidirectional, and road lane limits. Users were also asked to fill out a survey to note their driving experience statistics and general demographic information.

We used a Logitech Driving Force G29 Racing wheel and paddle setup, to interface with the gazebo environment. The interface for the study was completely based on Gazebo for visualization with ROS handling the back-end processing and communication. We wanted the drivers to have an idea that there is a non-zero probability of human beings appearing on the street, so that they drive with a safety-primed perspective to account for the possibility of pedestrians behind the hazard. This was done indirectly by lining pathways with moving pedestrians. During the pilot run, we also added actual pedestrians crossing the streets from the pathway and also from behind vans to directly prime them for such a possibility. For actual data collection cases a pedestrian was programmed to walk out from behind the hazard with a probability of 30%.

For the controlled cases we manipulated the following independent categorical variables: (a) Size of the vehicle: Moderate (Van) or Large (truck), (b) Closeness of vehicle to the driving lane: Close ($\sim$ 10% of the vehicle parked on the road) or Far (vehicle clearly parked on the curb), (c) Direction of Traffic: No traffic (unidirectional road) or Opposing traffic (bidirectional road). This gives us 8 unique scenarios which the user was required to drive through in the study. Figure 1 shows how the first two aspects were varied in the Gazebo world. Our dependent variables were: the choice of sub-lane on the road and the speed of the car.

In order to find the significance of effect that size and position of hazard had on driver's behavior we used paired t-test and Wilcoxon's test on the non-collision runs for each user. For the data measures to be of equal cardinality irrelevant of speed and sampling frequency, we binned speed and *sub-lane* values over every 0.5 meters and averaged them. Sub-lanes are lanes were further categorized into 0.2 m wide strips. We only consider the data after D_{thresh} has been crossed by the ego-car. We did this separately for each dependent variable. We only compared observations under similar traffic conditions.

3 Training and Generating Constraints

We adopted Key-frame based Learning from Demonstration (KLfD) from Akgun et al. [1] for modeling our training trajectories. It works by identifying key-frames in trajectories by repeatedly splining them based on base-set of knots, comparing interpolated trajectory with the original and adding points of maximum error to the set until this comparison error is less than some threshold. The base knot set consists of the start and end-points of the user demonstrated trajectory and the final set is then termed as *key-frames*. These key-frames are then further time aligned using Dynamic Time Warping [4] and clustered to give mean key-frames which can be used to extrapolate the behavior trajectory at run-time.

We used key-frames as an indirect measure of how far the ego-agent has progressed in its behavior. The more distance ego-car has traveled with respect to the hazard, the further it is in its behavior execution. We were able to use such a simplistic metric only because we are only considering variations of given target case. However, if an oracle exists which can provide our system with the closest key-frame to current state, this method will work irrelevant of level of data abstraction.

3.1 Training

Before starting key-frame extraction, the recorded features are first transformed to an intermediate, hazard-centric representation (see sub-figure (a), (b) of figure 3). We are following ROS conventions here, which means forward of ego-car (trajectory axis) is positive y and left of ego-car (sub-lane axis) is positive x. After transformation the feature pile consists of: Sub-lane ego-car is on, Lateral and perpendicular distance of ego-car from the hazard, V_x, V_y of the ego-car, Size of hazard, and Traffic condition.

Once transformed into this feature-set, we use only the non-accidental "good" (with points within road limits) trajectories and we train one model per unique scenario resulting in 8 different models. We follow the same steps as KLfD approach, except we use cubic splines (trade-off between smoothness and ability to linearly combine many knots) and save the mean as well as the variance of clustered key-frames since we are primarily interested in the safety boundaries. This ordered tuple of $\left(\mu_{K_x}, \sigma_{K_x}\right)$ is saved as the scenario model.

3.2 Constraint Generation

The final framework takes the following features as input: Current sub-lane of the car, position of the hazard, current heading of the car, current speed of the car, size of hazard, and the type of road. The output of the framework is max and min limits on the sub-lanes as well as vehicle speed for a time horizon of 5 seconds. For evaluation purposes, we are not classifying the cases but only passing randomized start values to each case model.

The system first calculates distance horizon by using current speed along with time horizon of 5 seconds in ego-centric coordinate system. This distance is used to create a grid in hazard-centric coordinate system, where key-frame variances are splined to create envelopes with respect to the hazard's location. Finally these envelopes are converted back to ego-car's coordinate system and passed to constraint aggregator.

3.2.1 Only One Hazard

This is the simplest case, where we load the right model and use cubic splines to extrapolate the envelope based on one standard deviation in both directions, accounting for $\sim 70\%$ of the variation.

3.2.2 Multiple Hazards

We treat each hazard as a new isolated one once we hit the D_{thresh} for it. If the hazards are at enough distance from each other such that the latter's D_{thresh} and formers active behavior time do not collide, then the agent just treats them as several one-hazard case and applies treatment from previous section. However, if hazards are close enough that the active behavior time and distance threshold overlap then the agents treats this as two hazards together. The algorithm in this case, uses an adapted D_{thresh} for next hazard. This provides us with a mixed key-frame set consisting of previous hazard's key-frames lying before adapted D_{thresh} and the next hazards key-frames lying after D_{thresh}. We use piecewise cubic splines to smoothly combine these knots from different key-frame sets. Cubic splines are heavily favored in field like computer graphics [8, 6] because they are twice differentiable and produce smoother curves as compared to higher degree splines, which is desirable in trajectory generation.

For hazard without any overlap we use end of the first hazard's bounding box as the adapted D_{thresh} and for hazards with overlap in bounding boxes we use the average of their centers as the adapted D_{thresh} (refer to section 4 for why we follow this methodology for combination).

4 Results and Discussion

The collected data suggests drivers follow a common pattern for the studied case. At some distance D_{thresh} the drivers start veering away from the sub-lane which is closer to the hazard, they all trace a path which maximizes their distance from the hazard and continue onwards or come back to the original lane depending upon traffic conditions. We found that for all the users in our population there was a statistically significant effect of size and position of hazard on driving behavior. Both our

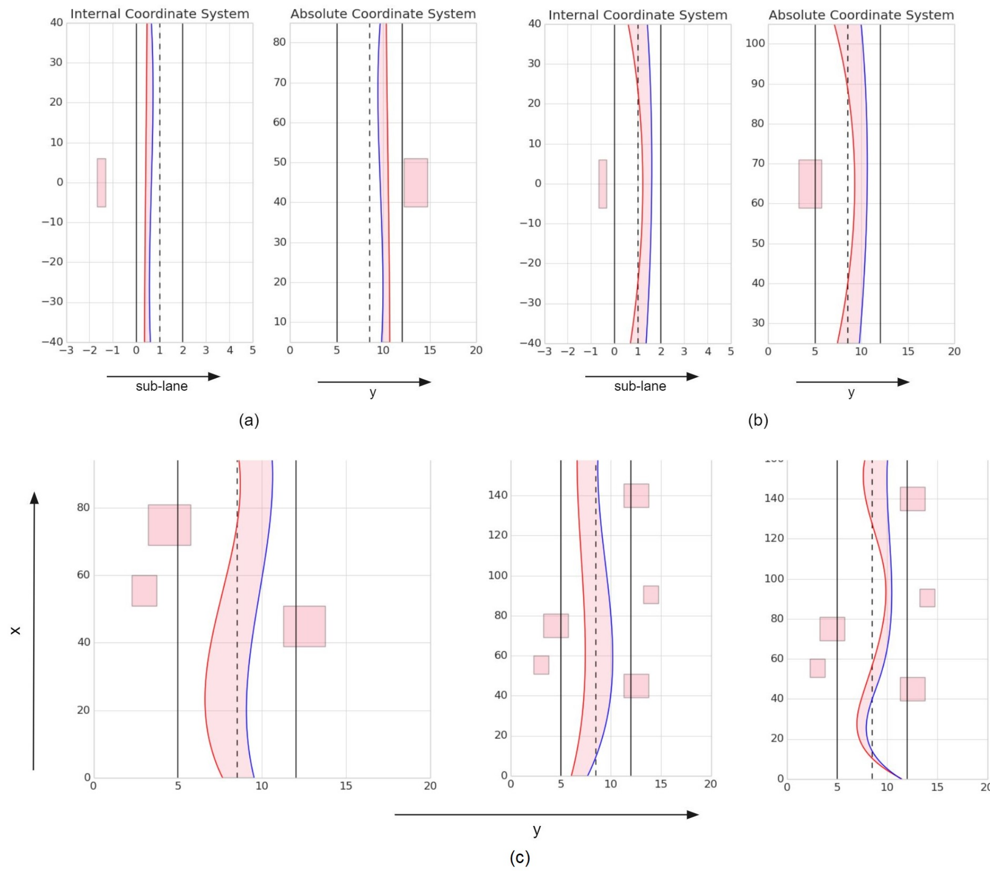

Figure 3: Generated Constraints for In-Distribution and OOTD Cases
(a) Constraints for large hazard in far position under bidirectional condition, (b) Constraints for large hazard in near position under unidirectional condition, (c) Constraints for OOTD cases arranged in order of complexity from left to right. They feature additional number of hazards and the last one also incorporates the bidirectional constraint of the road.

tests resulted in a p-value of less than 0.05. Interestingly, except one user everyone else collided with the occluded pedestrian at least once.

Figure 3 shows the generated constraints for cases from training set as well as OOTD cases. Sub-figures (a) and (b) show the internal hazard-centric coordinate system along with the resultant constraints in ego-car's coordinate system to illustrate the transformation. Sub-figure (c) shows generated constraints for cases with multiple hazards in sequence. The rightmost figure in sub-figure (c) is the hardest case that our algorithm can handle. Table 1 shows the distance of ego-car from

Table 1: Table showing D_{thresh} and point of Maximum Curvature for various cases

Traffic	Independent Variable	D_{thresh} in m	Point of Maximum Curvature (distance from hazard in m)
Uni	Small	36.5	-5.56
	Large	37.01	-5.01
	Near	36.18	-0.36
	Far	40	-13.17
Bi	Small	15.4	-10.92
	Large	14.93	0.01
	Near	16.15	-2.72
	Far	12.97	-8.06

hazard at the maximum point of curvature of the preemptive trajectory as well as the calculated D_{thresh} across the independent variables.

First, we present a qualitative evaluation of the auto-generated constraints. Our criteria includes two factors: 1. Following rules of the road, 2. Being safety optimal in novel situations. For the former criteria, we would like to point out how the algorithm performs on bidirectional roads versus unidirectional roads. The constraints, completely data-driven, are stricter for bidirectional scenario and the outer edge is more conservative as it tends to stick within its own lane allowance.

Now for the second, and more important factor, we would argue that the capability of the algorithm to handle multiple hazards in sequence attests to this. If one analyzes the figures a case can be made that the generated constraints ensure the trajectory is collision-free and realistic for a car to follow. To emphasize another interesting observation, one can see in the figures that the two extremes of the envelope are not actually symmetrical. While one is a tighter bound with less steering movement, the other is more curvy and weighted more towards "evading" the hazards. Such contrasting boundaries represent different stereotypes of driver profiles. The goal-oriented ones who take minimum deviations in evasive actions and the safety-oriented ones with extra steps of actions.

Next, we would like to briefly address the shortcoming of our algorithm here, namely its inability to handle hazards with higher degrees of overlap. We believe this is because humans inherently treat single hazard and multi-hazard situations differently and the single hazard demonstrations fail to capture this. The single hazard demonstrations tell us how far the ego-car should stay from the hazards, but when you add multiple such hazards this constraint can turn the ego-car into a sitting duck, just like a mobile robot surrounded by multitudes in a crowd. Such situations require an inherent concept of aggression and "goal-orientedness" on the agent's part, which is very different than what our demonstrations show.

Moving to the second part of our study, our statistical analysis suggests a validation of our initial hypothesis. It is also interesting to note here that as per Berndt and Clifford [4] such studies in simulation with results consistent across variables and users, are especially suited for evaluating hazard perception of the users and by an extension learning from the good measures. Moreover, hazard perception ability in humans has been found to be directly linked to risk of accident by the driver [10].

A striking observation that can be gleaned from table 1 is the variation in point of maximum curvature for positional cases under unidirectional road condition and for size under bidirectional condition. Under the bidirectional condition, the smaller hazard still allows the driver to have a largely unobstructed view of the road, see figure 1. In accordance with our hypothesis of preemptive behavior, the user here steers away well ahead in advance and is able to maintain a forward trajectory without colliding with either the hazard or the oncoming traffic while maintaining view of the road itself. However the larger hazard ends up blocking most of the user's access and view of the lane. In order to abide the road rules and prevent collision, the user here traces a larger curve trajectory but only when right next to the hazard. This is because this larger trajectory requires creeping into the next lane first before heading back on one's own.

This is an interesting case study for modeling corner-cases in AVs since our observations from real human users suggest that sometimes for the safety of ego as well as other agents in the environment, the rules of the road need to be made fluid. There needs to be a model which can inform the trade-off between such preemptive behaviors versus strict adherence to traffic rules depending upon scene-understanding. Under the current bottleneck of imperfect perception, we advocate the use of scientific exploration and analysis of corner-cases in urban driving scenarios to push the needle a little further in terms of increased autonomy with safety guarantees.

By the way of this paper, this is also our attempt in advocating for the use of LfD or imitation based techniques to encompass the complex decision process of a human driver. AVs are specially well suited for this problem, since recording "good", safe trajectories on the platform itself is much easier as compared to traditional robotics arms or even mobile platform operated via game controllers. This means the embodiment mapping [2] for the data is effectively the identity, thereby reducing the scope of our efforts to solve only the record mapping problem, i.e. states observed by machine to be similar to what the human observes.

References

[1] Baris Akgun, Maya Cakmak, Karl Jiang, and Andrea L Thomaz. Keyframe-based learning from demonstration. *International Journal of Social Robotics*, 4(4):343–355, 2012.

[2] Brenna D Argall, Sonia Chernova, Manuela Veloso, and Brett Browning. A survey of robot learning from demonstration. *Robotics and autonomous systems*, 57(5):469–483, 2009.

[3] Christopher G Atkeson and Stefan Schaal. Robot learning from demonstration. 97:12–20, 1997.

[4] Donald J Berndt and James Clifford. Using dynamic time warping to find patterns in time series. 10(16):359–370, 1994.

[5] Mariusz Bojarski, Davide Del Testa, Daniel Dworakowski, Bernhard Firner, Beat Flepp, Prasoon Goyal, Lawrence D Jackel, Mathew Monfort, Urs Muller, Jiakai Zhang, et al. End to end learning for self-driving cars. *arXiv preprint arXiv:1604.07316*, 2016.

[6] KW Brodlie and Sohail Butt. Preserving convexity using piecewise cubic interpolation. *Computers & Graphics*, 15(1):15–23, 1991.

[7] Chenyi Chen, Ari Seff, Alain Kornhauser, and Jianxiong Xiao. Deepdriving: Learning affordance for direct perception in autonomous driving. pages 2722–2730, 2015.

[8] Frederick N Fritsch and Ralph E Carlson. Monotone piecewise cubic interpolation. *SIAM Journal on Numerical Analysis*, 17(2):238–246, 1980.

[9] David Isele, Akansel Cosgun, Kaushik Subramanian, and Kikuo Fujimura. Navigating intersections with autonomous vehicles using deep reinforcement learning. *arXiv preprint arXiv:1705.01196*, 2017.

[10] Frank P McKenna and Mark S Horswill. Hazard perception and its relevance for driver licensing. *IATSS research*, 23(HS-042 879), 1999.

[11] Stefan Mitsch, Khalil Ghorbal, and André Platzer. On provably safe obstacle avoidance for autonomous robotic ground vehicles. 2013.

[12] Stéphane Ross, Geoffrey J Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. pages 627–635, 2011.

[13] Dorsa Sadigh, Shankar Sastry, Sanjit A Seshia, and Anca D Dragan. Planning for autonomous cars that leverage effects on human actions. 2016.

[14] Junqing Wei, Jarrod M Snider, Junsung Kim, John M Dolan, Raj Rajkumar, and Bakhtiar Litkouhi. Towards a viable autonomous driving research platform. pages 763–770, 2013.

[15] Tianhao Zhang, Gregory Kahn, Sergey Levine, and Pieter Abbeel. Learning deep control policies for autonomous aerial vehicles with mpc-guided policy search. pages 528–535, 2016.

[16] Brian D Ziebart, Andrew L Maas, J Andrew Bagnell, and Anind K Dey. Maximum entropy inverse reinforcement learning. 8:1433–1438, 2008.