Modeling Eye-Gaze Behavior of Electric Wheelchair Drivers via Inverse Reinforcement Learning

Yamato Maekawa[1], Naoki Akai[1], Takatsugu Hirayama[2], Luis Yoichi Morales[2],
Daisuke Deguchi[1], Yasutomo Kawanishi[1], Ichiro Ide[3] and Hiroshi Murase[1]

Abstract— **It is intuitively obvious that eye-gaze behaviors of experienced drivers are different from those of novice drivers. However, it is not easy to understand the difference in their behavior quantitatively. In this work, we present an explainable eye-gaze behavior modeling method for electric wheelchair drivers based on Inverse Reinforcement Learning (IRL). We first create feature maps that represent risk factors during driving. These feature maps are able to represent not only to what but also from where drivers pay attention. IRL uses the feature maps to learn the reward representing the eye-gaze behaviors and allows us to see important features via the automatic acquisition of the reward. Through analysis of the learned model, we show quantitative evidence that eye-gaze behaviors of experienced drivers are better-balanced by paying attention to multiple risks.**

I. INTRODUCTION

Since there are many traffic accidents involving personal mobility vehicles like an electric wheelchair, the demand for assisting their driving is increasing. It is known that while driving a car, we obtain information more than 90 percent from eyesight [1]. Likewise, wheelchair drivers also obtain a lot of information from eyesight and use this information for risk aversion. We consider that it should be effective to understand eye-gaze behaviors that lead to the prevention of accidents by electric wheelchairs. It is intuitively obvious that eye-gaze behaviors of experienced drivers are different from that of novice drivers. Therefore, we aim to analyze the eye-gaze behaviors of experienced and novice drivers of an electric wheelchair to quantitatively understand their differences. Our target application is visual cognition support for novice drivers during risky area driving.

To realize this application, a model on experienced drivers' eye-gaze behaviors that novice drivers can understand easily is needed. We consider that the model that explains to what object, i.e., obstacles or pedestrians, and from where the driver should pay attention, is efficient for this application because only presenting the gaze distribution of an experienced driver is insufficient to allow the understanding to which object we should pay attention.

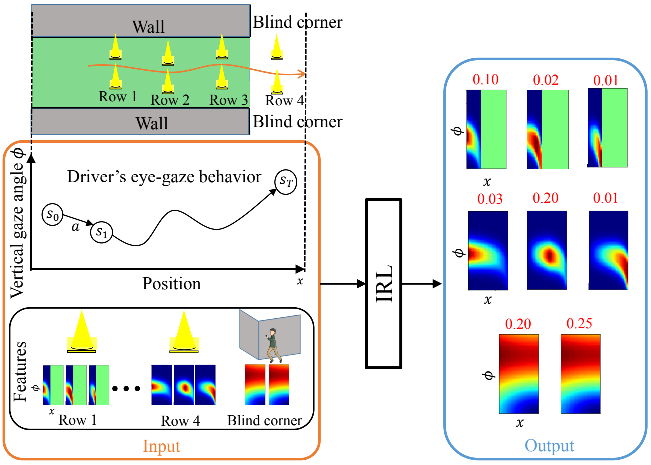


Fig. 1: Overview of our work. The collected drivers' eye-gaze behaviors and features that represent risk factors such as pylons and blind corners during driving are input to IRL. Red values are the weights learned by IRL.

In order to model drivers' eye-gaze behaviors, some works use Convolutional Neural Networks (CNN) [2], [3], [4]. Since these models make use of end-to-end frameworks, they cannot explain the reason for their output even if they visually output attention maps. Thus, in this work, we model the eye-gaze behaviors of experienced electric wheelchair drivers by means of an explainable model, i.e., to what and from where they pay attention.

To construct the model of eye-gaze behaviors, accurately measured eye-gaze vectors of the drivers are needed. Some researchers analyze eye-gaze behaviors of drivers using cameras facing the driver [5] and eye-gaze measurement glasses [6]. However, these measurements are limited due to the difficulty of obtaining accurate distance information. The information where drivers pay attention is quite important for visual cognition support [7]. Therefore, we have previously developed a novel platform that allows us to measure the driver's eye-gaze vector in the 3D space that allows us to know from where drivers pay attention [8], and this platform is also used in this paper.

In this paper, we make use of Inverse Reinforcement Learning (IRL) to extract the differences of eye-gaze behaviors between experienced and novice drivers. Figure 1

[1]Yamato Maekawa, Naoki Akai, Daisuke Deguchi, Yasutomo Kawanishi and Hiroshi Murase are with the Graduate School of Informatics, Nagoya University, Nagoya 464-8601, Japan maekaway@murase.is.i.nagoya-u.ac.jp, akai@nagoya-u.jp, {ddeguchi, kawanishi, murase}@i.nagoya-u.ac.jp

[2]Takatsugu Hirayama and Luis Yoichi Morales are with the Institute of Innovation for Future Society (MIRAI), Nagoya University, Nagoya 464-8601, Japan takatsugu.hirayama@nagoya-u.jp, morales_yoichi@coi.nagoya-u.ac.jp

[3]Ichiro Ide is with the Mathematical and Data Science Center, Nagoya University, Nagoya 464-8601, Japan ide@i.nagoya-u.ac.jp

978-1-7281-4148-0/20/$31.00 ©2020 IEEE

illustrates the overview of this work. In IRL, a reward that is used to imitate demonstrations is learned based on features that can be manually designed [9], [10]. Here, we design feature maps that are able to represent not only to what but also from where drivers pay attention. IRL allows us to see important features via the automatic acquisition of the reward. Through analysis of learning results using IRL, we show quantitative evidence that eye-gaze behaviors of the experienced drivers are better-balanced by paying attention to multiple risks.

The contribution of this work is twofold:

- Constructing an explainable model of eye-gaze behavior using IRL.
- Describing the differences between eye-gaze behaviors of experienced and novice drivers, i.e., to what and from where they pay attention.

The rest of this paper is organized as follows. Section II summarizes related work. Section III describes IRL in our application. Section IV details an experiment and shows its result. Finally, we conclude with a summary in Section V.

II. RELATED WORK

A. Experienced and Novice Drivers' Eye-gaze Behaviors

It has been reported that the eye-gaze behaviors of experienced and novice drivers have some differences [11], [12], [13]. Underwood et al. [12] showed that scanpaths in eye movements differ between experienced and novice drivers. They analyzed the scanpath between the subdivided regions of a driving scene such as near-left, far-ahead, and mid-right on the road. Van Leeuwen et al. [13] found that the horizontal gaze variance decreases and the percentage of gazing at the center of the road increases as drivers gain driving experience. Pradhan et al. [11] reported that the percentage that experienced drivers recognize risks is higher than that of novice drivers in many situations with risks. Both works conducted experiments on driving simulators [11], [13]. They also considered neither to what object the drivers gazed nor from where they gazed. In contrast, our work models to what and from where the drivers pay attention in an actual environment.

B. Explainable Model

Some works modeled the driver's gaze from on-vehicle camera images [2], [3], [4]. They used Convolutional Neural Networks (CNN) which make them end-to-end frameworks. Recently, explainable models are considered important to understand the decision making of deep CNN [14]. Mori et al. applied this to autonomous driving [15] and visually analyzed the decision making process such as steering and braking by predicting an attention map. Even with such frameworks, it is difficult to know to what object and from where we should pay attention while driving.

Drivers' eye-gaze behaviors have often been modeled using Hidden Markov Models (HMM) [16], which is regarded as an explainable model, where the hidden states might represent the driver's intention. However, it is still not easy to explain the eye-gaze behaviors by interpreting the constructed model.

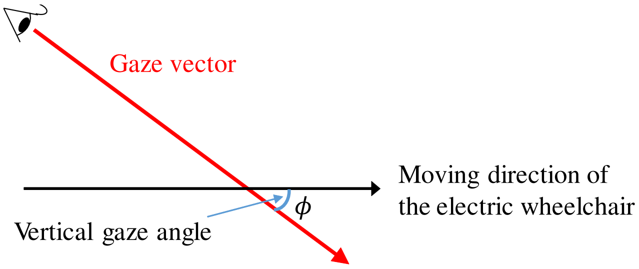


Fig. 2: Definition of vertical gaze angle [7].

Another approach to construct explainable models is the use of Inverse Reinforcement Learning (IRL). IRL is used for modeling human behaviors represented by a reward to each behavior [10], [17], [18], [19]. Activity forecasting [17] which is extended from maximum entropy IRL [19] modeled pedestrian trajectory and showed that humans prefer to walk on sidewalks more than on streets. Hirakawa et al. applied this approach to predict animal movements [10].

IRL is also used in driving situations. Wulfmeier et al. applied IRL to path planning in driving situations [20]. Shimosaka et al. modeled risk anticipation using the velocity of the car and show that expert drivers take brakes before an intersection with blind corners on residential roads [9], [21]. In contrast to their work modeling risk aversion focused on velocity, we focus on eye-gaze behaviors of drivers to model to what and from where we should pay attention because experienced drivers efficiently pay attention to risks using eyesight.

III. IRL FOR EYE-GAZE BEHAVIOR MODELING

As shown in Fig. 1, we model the eye gaze behavior of electric wheelchair drivers when driving at a place where multiple risks exist. The risks anticipated in this situation are as follows.

- Collision with pylons
- Collision with pedestrians running out from blind corners

A. Preliminary Analysis

We previously analyzed the eye-gaze behavior in the above mentioned situation [7]. The result of the analysis showed that the mean of the vertical gaze angles of experienced drivers is higher than that of novice drivers. The definition of the vertical gaze angle is shown in Fig. 2. To understand this difference between experienced and novice drivers, we analyzed to what and from where they paid attention. The results showed that the experienced drivers paid more attention to blind corners and focused on the pylons from distant positions for efficient risk aversion. In this work, we attempt to construct an explainable model of such gaze dynamics which represent to what and from where drivers pay attention as shown in Fig. 1.

B. Model Representation with Markov Decision Process

Inverse Reinforcement Learning (IRL) is based on Markov Decision Process (MDP). MDP is represented as $M = (S, A, P, R)$, where S is a finite state space, A is a finite action space, $P(s'|s,a)$ is the probability that state s transits to state s' by action a, and $R(s)$ is the reward at state s.

Ziebart et al. [19] and Kitani et al. [17] used 2D coordinates as state $s = (x, y)$ for modeling a trajectory. Generally, path prediction methods in IRL define negative rewards and the predicted trajectory tends to be relatively short because it takes negative rewards as less as possible to maximize the reward represented by the sum of negative immediate rewards $r(s)$. For presenting a detoured trajectory, Hirakawa et al. extended the state to 3D coordinates as $s = (x, y, z)$, where z represents the discrete time step [10]. By introducing the discretized elapsed time as a variable that represents the state, we can implicitly consider the length of the trajectory, i.e., state transition with longer elapsed time indicates a longer detoured trajectory. We use this maximum entropy IRL to model the eye-gaze behavior. Here, state s is represented as a combination of position x, vertical gaze angle ϕ, and discrete time step z, i.e. $s = (x, \phi, z)$. Note that the state is discretized to make use of MDP. The position is discretized at 0.2 m intervals into thirty steps from 0.0 m to 6.0 m, and the vertical gaze angle is discretized at 2 degrees intervals into sixty steps from -30 degrees to 90 degrees. Action a is defined as $a = (\mathrm{d}x, \mathrm{d}\phi, 1)$, which is the differential in adjacent states. A trajectory is represented as $\zeta = \{(s_0, a_0), (s_1, a_1), \dots\}$, where s_t is the t-th state of the trajectory. Note that the discrete time step $z_t = t$.

C. Designing Feature Maps

In IRL, the reward that is used to imitate demonstrations is learned based on features, which can be manually designed [9], [10]. In this work, we design feature maps that can represent not only to what but also from where drivers pay attention.

The preliminary analysis [7] described in Sec. III-A showed that there were some differences to what and from where experienced and novice drivers paid attention. Based on this result, we design feature maps considering the risk factors and the distance from them. A feature map $f(s)$ consists of a driver's position x, vertical gaze angle ϕ, and discrete time step z. First, we design the feature map $f(x, \phi)$ consisting of x and ϕ. Then we duplicate this along the z-axis, i.e., $f(x, \phi, 1) = \cdots = f(x, \phi, z)$. The reward function R is trained as a weighted combination of these feature maps as described in Sec. III-D.

The feature maps for pylons are designed according to the distance from them to model from where drivers pay attention to each pylon. Algorithm 1 shows how to generate feature maps for the pylon at each row. This algorithm is applied to all rows. Each value on feature maps takes a negative value within the range $[-1, 0]$. If the driver's gaze vector with vertical gaze angle ϕ at position x hits the pylon, the value at (x, ϕ) on the feature map $f(x, \phi)$ takes zero. The values follow (normal distribution -1) along the ϕ-axis direction around the pixel on the map. Then, we divide the distance from start to the pylon into five segments and apply linear filtering according to the divided points. An example of this process applied to pylon row 3 is shown in Fig. 3. The values of the pixels on the feature map farther than the position of the pylon take $(\min + \max)/2 = 0.5$ as shown in the rightmost area of each map in Fig. 3. We set the height of the driver's eye at 1.5 m as the average of drivers.

Algorithm 1 Generating the feature map of the pylon at each row

```
$divided\_point \leftarrow [0, \frac{1}{5}x_{\mathrm{pylon}}, \frac{2}{5}x_{\mathrm{pylon}}, \frac{3}{5}x_{\mathrm{pylon}}, \frac{4}{5}x_{\mathrm{pylon}}, x_{\mathrm{pylon}}]$
for $j$ in $x$ do
  if $x < x_{\mathrm{pylon}}$ then
    $hit\_angle_{bottom} \leftarrow \arctan\left(\frac{x_{\mathrm{pylon}} - j}{\mathrm{height}_{\mathrm{driver's\_eye}}}\right)$
    $hit\_angle_{top} \leftarrow \arctan\left(\frac{x_{\mathrm{pylon}} - j}{\mathrm{height}_{\mathrm{driver's\_eye}} - \mathrm{height}_{\mathrm{pylon}}}\right)$
    $norm_{bottom} \leftarrow N(\mu = hit_{bottom})$
    $norm_{top} \leftarrow N(\mu = hit_{top})$
    for $i$ in gaze do
      if $hit\_angle_{top} < i < hit\_angle_{bottom}$ then
        $feature[i, j] \leftarrow \max(norm_{bottom}[i], norm_{top}[i])$
      else
        $feature[i, j] \leftarrow \max(norm_{bottom})$
      end if
      normalize in [-1,0]
    end for
    $feature[j] \leftarrow (1 + |divided\_point - j|) \times feature[j]$
    normalize in [-1,0]
  else
    $feature[j] \leftarrow 0.5$
  end if
end for
```

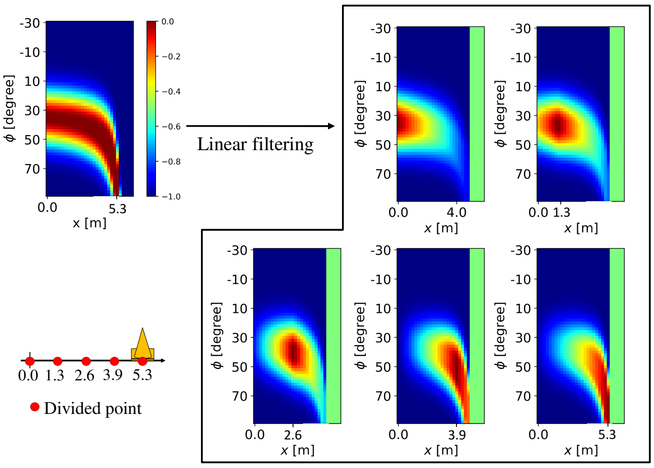


Fig. 3: Feature maps for pylon row 3. Linear filtering is applied according to the divided points. The values at positions farther than the pylon are fixed at 0.5 (= (min + max)/2).

We also prepare feature maps for blind corners as shown in Fig. 4. These feature maps are represented as a combination of several Gaussian kernels. These features for blind corners

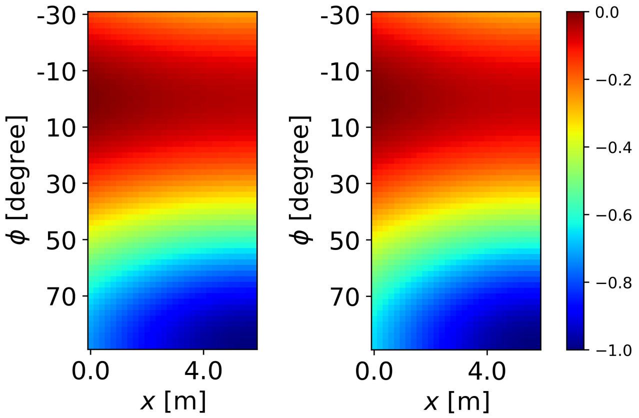


Fig. 4: Feature maps for blind corners.

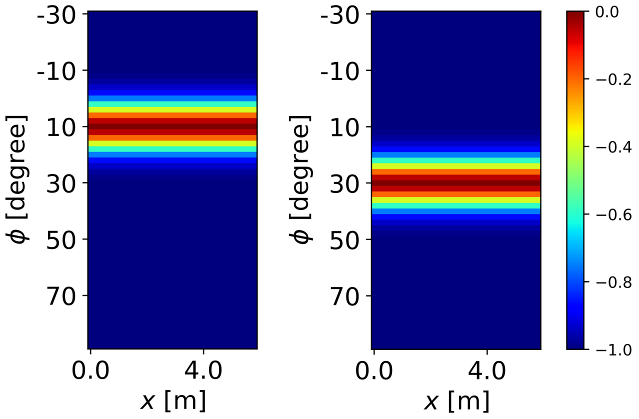


Fig. 5: Examples of feature maps with constant values for rows.

are generated as $f(s) = -\exp(s - s_{\mathrm{bc}})^T \sum (s - s_{\mathrm{bc}})$, where $s_{\mathrm{bc}} = [x_{\mathrm{bc}}, 90]^T$ is a vector corresponding to the position of blind corners x_{bc} and $\phi = 90$. $\sum$ is the covariance matrix. The value becomes higher as the vertical gaze angle becomes close to zero, i.e., $\phi = 0$, because we look ahead when we pay attention to blind corners while driving. As x becomes close to the position of blind corners, the area with such high values narrows to penalize gazes looking down on the floor or up to the ceiling far from the pedestrian's face or body running out. This means that the weight of these features decreases if the driver's gaze shifts away from blind corners near them. The feature maps with constant values along the x-axis are also designed as simple basis vectors that take fixed vertical gaze angles as shown in Fig. 5.

Finally, these feature maps $f(x, \phi)$ are expanded along the z-axis $f(x, \phi, z)$. Consequently, 29 feature maps are designed and their breakdown is shown in Table I.

TABLE I: Breakdown of feature maps.

Feature map	# of feature maps
Pylon row 1	5
Pylon row 2	5
Pylon row 3	5
Pylon row 4	5
Blind corner	2
Constant value	7
Total	29

D. Inverse Reinforcement Learning

For IRL, we train a reward function. The reward is represented as a weighted combination of features $\boldsymbol{f}(s) = [f_1(s), \ldots, f_K(s)]^T$ designed in Sec. III-C, and the weight vector $\boldsymbol{\theta}$.

$$R(\zeta; \boldsymbol{\theta}) = \sum_t r(s_t; \boldsymbol{\theta}) = \sum_t \boldsymbol{\theta}^T f(s_t). \tag{1}$$

For maximum entropy IRL, the policy is given by $\pi(a|s) \propto \exp(Q(s,a) - V(s))$, where value function $V(s)$ and action value function $Q(s,a)$ are computed from the reward by solving the Bellman equation [19] as follows.

$$\begin{aligned} V(s) &= \max_a Q(s,a) \\ &+ \log\left[1 + \exp\left\{\min_a Q(s,a) - \max_a Q(s,a)\right\}\right]. \end{aligned} \tag{2}$$

$$Q(s,a) = r(s; \boldsymbol{\theta}) + E_{p(s'|s,a)}[V(s')]. \tag{3}$$

Weight vector $\boldsymbol{\theta}$ is estimated as $\hat{\boldsymbol{\theta}}$ by maximizing the log-likelihood $L(\boldsymbol{\theta})$ of the probability distribution, where Z is a set of trajectories $Z = \{\zeta_1, \ldots, \zeta_n\}$, as follows:

$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} L(\boldsymbol{\theta}) = \arg\max_{\boldsymbol{\theta}} \frac{1}{|Z|} \sum_{\zeta \in Z} \log P(\zeta|\boldsymbol{\theta}). \tag{4}$$

The probability distribution with maximum entropy is given by

$$P(\zeta|\boldsymbol{\theta}) = \frac{\exp(\sum_t \boldsymbol{\theta}^T f(s_t))}{Z(\boldsymbol{\theta})}, \tag{5}$$

where $Z(\boldsymbol{\theta})$ is a normalizer. The exponentiated gradient method [22] is used to update the weight with learning rate λ, as follows:

$$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} \exp(\lambda \nabla L(\boldsymbol{\theta})), \tag{6}$$

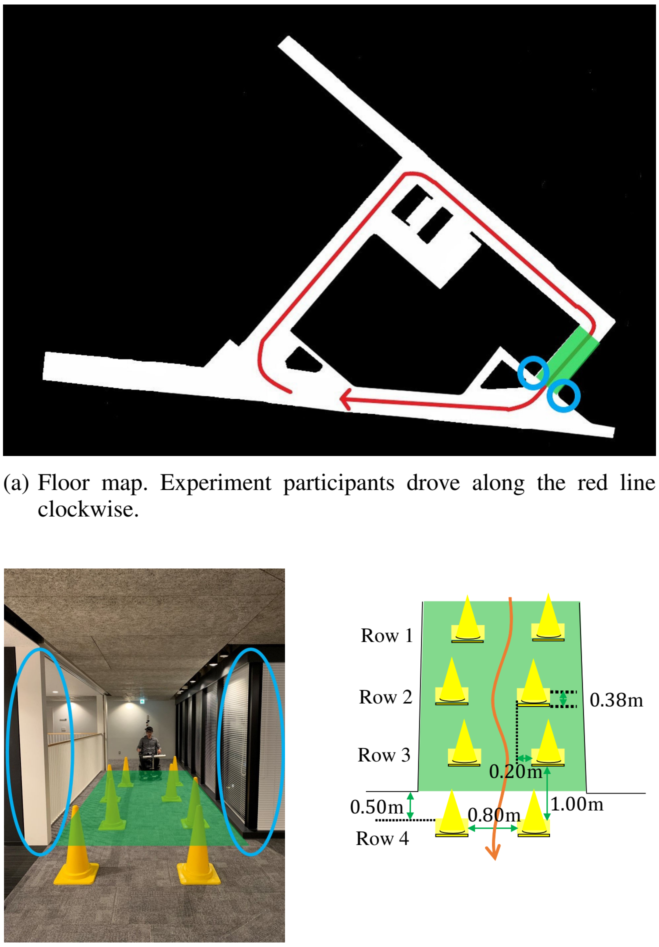
$$\nabla L(\boldsymbol{\theta}) = \widetilde{\mathbf{f}} - \sum_\zeta P(\zeta|\boldsymbol{\theta}) \boldsymbol{f}_\zeta = \widetilde{\mathbf{f}} - \frac{1}{|Z|} \sum_{\zeta \in Z} \sum_t f(s_t) D_\zeta(s_t), \tag{7}$$

where D is the expected visitation count of state s and $\widetilde{\mathbf{f}}$ is the expected emprical feature count represented as $\widetilde{\mathbf{f}} = \frac{1}{|Z|} \sum_{\zeta \in Z} [\sum_t f(s_t)]$.

IV. Experiment

A. Experimental Setup

We conducted an experiment in an indoor environment at Nagoya University, Japan. Fig. 6(a) shows the map of the indoor environment. A narrow course was simulated by placing pylons on a corridor. We analyzed the data acquired in the green segment shown in Fig. 6(b). Fig. 6(c) shows the layout of the pylons. As shown in Fig. 6(b), there are blind corners on both sides.



(a) Floor map. Experiment participants drove along the red line clockwise.

(b) Data collection segment.

(c) Layout of pylons. The electric wheelchair cannot keep running straight when passing between pylons.

Fig. 6: Experiment course and layout of pylons [7]. Data in the green segment were collected. The blue circles indicate blind corners. While there were unrelated people walking in the vicinity, only the experiment participants and collaborators entered the green segment.

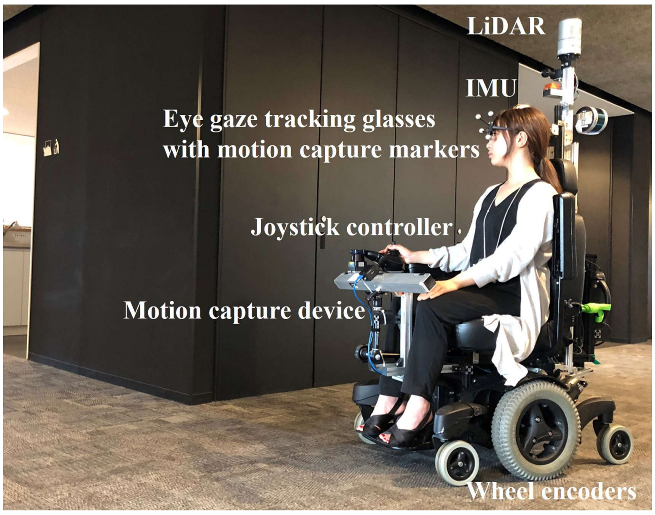


Fig. 7: Electric wheelchair [8].

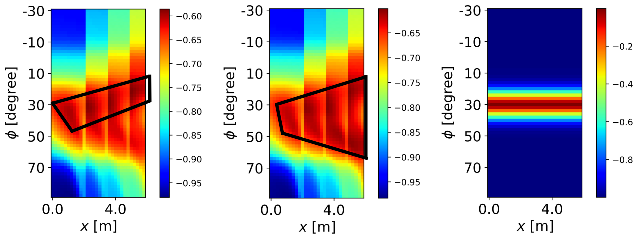
The electric wheelchair for acquiring driving data is shown in Fig. 7. A LiDAR sensor[1], an Inertial Measurement Unit (IMU)[2], and a motion capture device[3] were attached to it. Tobii Pro Glasses 2 was used as eye-tracking glasses to acquire eye-gaze behavior data. We made a 3D map with Simultaneous Localization And Mapping (SLAM) [23] using LiDAR beforehand. Then the position of the electric wheelchair on the map was estimated with Adaptive Monte Carlo Localization (AMCL) [24], [25] using the 3D point cloud acquired from the LiDAR, the 3D map and odometry information attached from velocity, acceleration, and angular velocity of the electric wheelchair. The 3D gaze vector in world coordinates was estimated by tracking spherical markers equipped to the eye-tracking glasses using the motion capture device and the vertical gaze angle ϕ was calculated from the gaze vector. If the collected data had the same states $s = (x, \phi)$ continuously, then the first one was kept by removing others to ensure that a trajectory consists of a sequence of different states s.

We measured the eye-gaze behaviors of ten participants; male students in their twenties and were inexperienced with driving electric wheelchairs before participating. The data include 29 laps per each participant. An experiment collaborator ran out in front of the electric wheelchair from each blind corner in order to make the participants aware of risks. The events took place twice from each blind corner on laps 6, 12, 18, and 24. In this paper, we consider data from laps 1–5 as novice driver data and laps 25–29 as experienced driver data. The elapsed time was then replaced with the number of state transitions from the beginning. We used this as the discrete time step corresponding to approximately 0.05 s on average. Note that the experiment was conducted after obtaining the approval of the Ethics Committee of the University. More details are described in [7].

B. Extracted Features

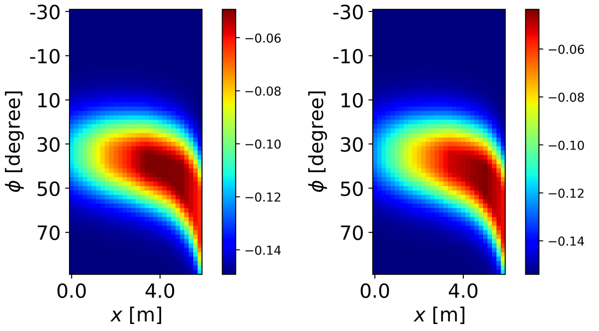
The reward maps of experienced and novice drivers are calculated as a combination of features and learned weights as described in Eq. 1. The learned reward maps of experienced and novice drivers are shown in Fig. 8. The reward map of the novice drivers (Fig. 8(b)) shows that the vertical gaze angle diverges near blind corners represented in dark red. In contrast, the reward map of the experienced drivers (Fig. 8(a)) shows that their gaze vector rises near blind corners. These results show that we could represent the difference between experienced and novice drivers' eye-gaze behaviors.

[1] Velodyne HDL-32
[2] Xsens MTi-300-2A5G4
[3] Optitrack V120:TRIO



(a) Experienced (b) Novice (c) Experienced and novice combined

Fig. 8: Reward maps of experienced and novice drivers. The black lines enclose the highest value of the reward colored in dark red. The enclosed area in the reward map of experienced drivers is slanted upward as they approach blind corners.



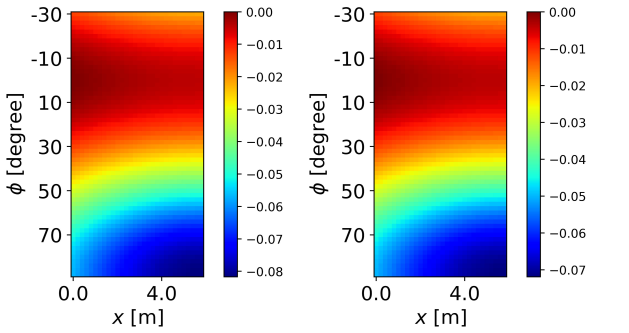
(a) Experienced (b) Novice

Fig. 9: Reward maps for pylon row 4 of experienced and novice drivers.

Fig. 8(c) shows the reward map learning both data of experienced and novice drivers, which was not learned well. We consider that this is because the probability distribution estimated by Inverse Reinforcement Learning (IRL) was averaged by mixing different types of trajectory sets. As another reason, we consider that the weights plunged into a locally optimal solution.

We investigate which features contributed to these differences. Here we focus on pylon row 4 based on our preliminary analysis [7], as shown in Fig. 9. The reward maps represent the weighted combination of five feature maps for pylon row 4. The reward map of the experienced drivers is high at around $x = 4.0$. In contrast, the reward map of the novice drivers is high at around $x = 5.0$. Next, we focus on the blind corners. The reward maps for the blind corner of the two models have little difference as shown in Fig. 10. However, the learned weight of the experienced drivers is higher than that of the novice drivers as shown in Fig. 11. Therefore, we can say that the experienced drivers paid more attention to the blind corners. We consider that they paid attention to blind corners to prepare for the risk that a pedestrian may run out from them.

These results show that to what and from where the drivers pay attention thanks to using the explainable model.



(a) Experienced (b) Novice

Fig. 10: Reward maps for blind corners of experienced and novice drivers.

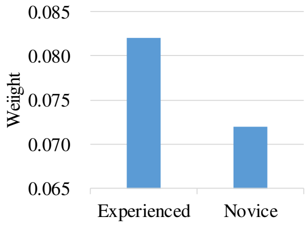


Fig. 11: Learned weight of the feature map for the blind corner.

C. Model Evaluation

We first evaluate our model in prediction accuracy. We make use of Modified Hausdorff Distance (MHD) [26] which is commonly used to calculate the difference between two trajectories in IRL [9], [10], [17], [21] as an evaluation metric. MHD is calculated as

$$\mathrm{MHD}(A,B) = \max\left\{\frac{1}{N_a}\sum_{a\in A} d(a,B), \frac{1}{N_b}\sum_{b\in B} d(b,A)\right\} \quad (8)$$

$$d(a,B) = \min_{b\in B} ||a-b|| \quad (9)$$

where $A = \{a_1,\dots,a_{N_a}\}$ and $B = \{b_1,\dots,b_{N_b}\}$ are the predicted trajectory and ground-truth trajectory of eye-gaze behaviors, respectively. We randomly split the data into training and testing data at a ratio of 4 to 1 to evaluate our model. These models were tested by swapping the novice and experienced drivers' data. The result is shown in Table II. The model of the experienced drivers which was tested with the experienced drivers' data achieved smaller mean MHD than that with the model of the novice drivers. Also the model of the novice drivers which was tested with the novice drivers' data achieved smaller mean MHD than that with the model of experienced drivers. Thus we confirmed that our model has a high ability to represent the eye-gaze behaviors of experienced and novice drivers.

To model drivers' eye-gaze behaviors by IRL, we confirmed that designing appropriate features that represent the eye-gaze behaviors related to the surrounding environment is

TABLE II: Prediction accuracy in mean MHD.

		Testing data	
		Experienced	Novice
Training data	Experienced	**3.25**	4.14
	Novice	3.33	**3.77**

needed because the rewards are represented by the combination of weighted features. To represent the relationship, 3D measurement of eye-gaze behaviors is needed. This enables us to model from where drivers pay attention to risks. Also, collecting trajectories with the same patterns is needed for better training. We consider that we have successfully modeled the eye-gaze behavior of drivers in IRL since we satisfied the above conditions.

V. CONCLUSION

We presented an explainable modeling method of the eye-gaze behaviors of electric wheelchair drivers based on Inverse Reinforcement Learning (IRL). For IRL, we designed features that represent risk factors explicitly to understand to what and from where drivers pay attention. IRL automatically chose important features to imitate the behaviors and we analyzed the learning results, i.e., the weights for the features. Analysis of the learned models showed that experienced drivers tend to pay attention to blind corners more than novice drivers while paying attention to the pylons. Therefore, we concluded that the experienced drivers' eye-gaze behaviors are better-balanced by paying attention to multiple risks.

Future work includes analyzing other situations like interaction with pedestrians, adopting the 3D information of the surrounding environment, and introducing the velocity of the electric wheelchair in the modeling.

ACKNOWLEDGMENT

This work was supported by JST-Mirai Program Grant No. JPMJMI17C6 and JSPS KAKENHI Grant No. 17H00745 and 19K12080.

REFERENCES

[1] E. Hartman, "Driver vision requirements," *Society of Automotive Engineers, Technical Paper Series*, no. 700392, pp. 629–630, 1970.

[2] A. Palazzi, D. Abati, S. Calderara, F. Solera, and R. Cucchiara, "Predicting the driver's focus of attention: The dr(eye)ve project," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 41, no. 7, pp. 1720–1733, 2018.

[3] Y. Xia, D. Zhang, J. Kim, K. Nakayama, K. Zipser, and D. Whitney, "Predicting driver attention in critical situations," in *Proc. 14th Asian Conf. on Computer Vision*, vol. 5, 2019, pp. 658–674.

[4] M. Ning, C. Lu, and J. Gong, "An efficient model for driving focus of attention prediction using deep learning," in *Proc. 2019 IEEE Intelligent Transportation Systems Conf.*, 2019, pp. 1192–1197.

[5] S. Martin and M. M. Trivedi, "Gaze fixations and dynamics for behavior modeling and prediction of on-road driving maneuvers," in *Proc. 2017 IEEE Intelligent Vehicles Symposium*, 2017, pp. 1541–1545.

[6] S. S. Mateo, E. Perez-Moreno, F. Jiménez, J. E. Naranjo, C. G. P. Flores, and J. A. Teruel, "Study of a driver assistance interface for merging situations on highways," in *Proc. 2018 IEEE Int. Conf. on Vehicular Electronics and Safety*, 2018, pp. 1–5.

[7] Y. Maekawa, N. Akai, T. Hirayama, L. Y. Morales, D. Deguchi, Y. Kawanishi, I. Ide, and H. Murase, "An analysis of how driver experience affects eye-gaze behavior for robotic wheelchair operation," in *Proc. IEEE Int. Conf. on Computer Vision 2019 Workshops*, 2019, pp. 4443–4451.

[8] N. Akai, T. Hirayama, L. Y. Morales, and H. Murase, "Safety criteria analysis for negotiating blind corners in personal mobility vehicles based on driver's attention simulation on 3D map," in *Proc. 2019 IEEE Int. Conf. on Intelligent Transportation Systems*, 2019, pp. 2367–2374.

[9] M. Shimosaka, T. Kaneko, and K. Nishi, "Modeling risk anticipation and defensive driving on residential roads with inverse reinforcement learning," in *Proc. 17th IEEE Int. Conf. on Intelligent Transportation Systems*, 2014, pp. 1694–1700.

[10] T. Hirakawa, T. Yamashita, T. Tamaki, H. Fujiyoshi, Y. Umezu, I. Takeuchi, S. Matsumoto, and K. Yoda, "Can AI predict animal movements? Filling gaps in animal trajectories using Inverse Reinforcement Learning," *Ecosphere*, vol. 9, no. 10, p. e02447, 2018.

[11] A. Pradhan, K. R Hammel, R. Deramus, A. Pollatsek, D. Noyce, and D. Fisher, "Using eye movements to evaluate effects of driver age on risk perception in a driving simulator," *Human Factors*, vol. 47, no. 4, pp. 840–52, 2005.

[12] G. Underwood, P. Chapman, N. Brocklehurst, J. Underwood, and D. Crundall, "Visual attention while driving: Sequences of eye fixations made by experienced and novice drivers," *Ergonomics*, vol. 46, no. 6, pp. 629–649, 2003.

[13] P. M. van Leeuwen, R. Happee, and J. C. de Winter, "Changes of driving performance and gaze behavior of novice drivers during a 30-min simulator-based training," *Procedia Manufacturing*, vol. 3, pp. 3325–3332, 2015.

[14] H. Fukui, T. Hirakawa, T. Yamashita, and H. Fujiyoshi, "Attention branch network: Learning of attention mechanism for visual explanation," in *Proc. 2018 IEEE/CVF Conf. on Computer Vision and Pattern Recognition*, 2018, pp. 10 697–10 706.

[15] K. Mori, H. Fukui, T. Murase, T. Hirakawa, T. Yamashita, and H. Fujiyoshi, "Visual explanation by attention branch network for end-to-end learning-based self-driving," in *Proc. 2019 IEEE Intelligent Vehicles Symposium*, 2019, pp. 1577–1582.

[16] N. Akai, T. Hirayama, L. Y. Morales, Y. Akagi, H. Liu, and H. Murase, "Driving behavior modeling based on hidden Markov models with driver's eye-gaze measurement and ego-vehicle localization," in *Proc. 2019 IEEE Intelligent Vehicles Symposium*, 2019, pp. 828–835.

[17] K. M. Kitani, B. D. Ziebart, J. A. Bagnell, and M. Hebert, "Activity forecasting," in *Proc. 12th European Conf. on Computer Vision*, vol. 4, 2012, pp. 201–214.

[18] K. Mülling, A. Boularias, B. J. Mohler, B. Schölkopf, and J. Peters, "Learning strategies in table tennis using inverse reinforcement learning," *Biological Cybernetics*, vol. 108, pp. 603–619, 2014.

[19] B. Ziebart, A. Maas, J. Bagnell, and A. K. Dey, "Maximum entropy inverse reinforcement learning," in *Proc. 23rd AAAI Conf. on Artificial Intelligence*, 2008, pp. 1433–1438.

[20] M. Wulfmeier, D. Rao, D. Z. Wang, P. Ondruska, and I. Posner, "Large-scale cost function learning for path planning using deep inverse reinforcement learning," *Int. J. of Robotics Research*, vol. 36, no. 10, pp. 1073–1087, 2017.

[21] M. Shimosaka, K. Nishi, J. Sato, and H. Kataoka, "Predicting driving behavior using inverse reinforcement learning with multiple reward functions towards environmental diversity," in *Proc. 2015 IEEE Intelligent Vehicles Symposium*, 2015, pp. 567–572.

[22] J. Kivinen and M. K. Warmuth, "Exponentiated gradient versus gradient descent for linear predictors," *Information and Computation*, vol. 132, no. 1, pp. 1–63, 1997.

[23] G. Grisetti, C. Stachniss, and W. Burgard, "Improved techniques for grid mapping with Rao-Blackwellized particle filters," *IEEE Trans. on Robotics*, vol. 23, no. 1, pp. 34–46, 2007.

[24] N. Akai, L. Y. Morales, and H. Murase, "Mobile robot localization considering class of sensor observations," in *Proc. 2018 IEEE/RSJ Conf. on Intelligent Robots and Systems*, 2018, pp. 3159–3166.

[25] S. Thrun, D. Fox, W. Burgard, and F. Dellaert, "Monte Carlo localization for mobile robots," in *Proc. 1999 IEEE Int. Conf. on Robotics and Automation*, 1999, pp. 1322–1328.

[26] M.-P. Dubuisson and A. K. Jain, "A modified Hausdorff distance for object matching," in *Proc. 12th IAPR Int. Conf. on Pattern Recognition*, vol. 1, 1994, pp. 566–568.