Cognitive Distraction Detection Using Gaze and Pupil with an Interpretable Approach

Kimimasa Tamura*, Simon Stent, John Gideon, Kohei Shintani, Guy Rosman

***Abstract*—Cognitive distraction (CD) is one of the major causes of traffic accidents, but there remains room to improve its detection. Most prior research on CD detection has commonly used basic statistical measures (e.g., mean, standard deviation) of driver-facing camera signals such as gaze and pupil size. However, these signals often exhibit subtle and complex patterns that conventional approaches cannot fully capture. In this paper, we evaluate a wide range of machine learning models and feature extraction methods using data from 52 participants in a driving simulator under two cognitive distraction inducing tasks (n-back and statement tasks). Our results demonstrate that combining gaze, pupil, and features derived from physiological signals (e.g., fixation saccade ratio and gaze entropy) and comprehensive time-series feature extraction boosts detection performance. While deep neural networks (Transformers) excel at modeling intricate relationships, our results show that tree-based ensemble methods (e.g., CatBoost) achieve comparable or higher detection performance while maintaining their advantage of better interpretability. Cross-task experiments further show that models trained on one type of task can generalize to another task. Feature analyses (via SHAP and Sobol) reveal that nonlinearity in vertical gaze movements, baseline pupil size, and greater minimum gaze distance are related to CD. These findings suggest that integrating multiple modalities, sophisticated feature engineering, and employing models capable of capturing nonlinear interactions are effective strategies for detecting CD. To support future research in this field, we release our code and preprocessed data: https://toyotaresearchinstitute.github.io/IV25-cognitive-distraction/.**

I. INTRODUCTION

Driver distraction has been identified as one of the major causes of traffic accidents. With advancements in driver monitoring technology, there is an expectation for such technologies to help detect driver distraction and contribute to accident prevention [1]. Driver distraction can be broadly classified into three types: *Manual Distraction* (hands off the wheel), *Visual Distraction* (eyes off the road), and *Cognitive Distraction* (mind off the task) [2]. Manual distraction occurs when a driver takes their hands off the wheel to perform non-driving-related tasks, such as eating or using a cellphone. Visual distraction refers to a state in which the driver's attention is physically diverted away from the road. In contrast, cognitive distraction (CD), often called "mind wandering", occurs when a driver's cognitive focus shifts from the primary driving task to engage in unrelated thoughts. Unlike visual distraction, which can be detected more easily using gaze data [3], cognitive distraction is more subtle in its manifestations within gaze behavior and other driver-facing camera signals [4].

CD detection studies commonly used basic statistical summaries (mean, standard deviation, skewness, kurtosis) of signals [5], [6]. More recently, McDonald et al., [7] explored large-scale time-series feature extraction for vehicle-control and physiological data. In contrast, our work focuses on camera-based ocular signals (gaze and pupil). However, the effects of CD are often subtle. In order to improve detection performance, we hypothesize that it is essential to capture more complex patterns and interactions between multiple modalities. One approach which is known to be able to capture such patterns is Deep Neural Networks (DNNs). They can automatically learn complex feature representations and interdependencies, often outperforming traditional machine learning models. However, DNNs are sometimes referred to as "black-box" models [8], raising concerns for in-vehicle applications where explainability and functional safety are critical. To address this, we explore a more interpretable approach based on tree-ensemble models while incorporating complex features that conventional methods fail to capture. This approach achieves performance comparable to that of modern DNNs while offering interpretable outputs through feature importance, which could facilitate the practical deployment of CD detection for in-vehicle systems. We validate our hypothesis through extensive experiments, including validation across different types of distraction tasks, and provide in-depth analysis that offer generalizable insights into CD detection.

The key contributions of this paper are as follows:

1) We evaluate a range of machine learning models for CD detection and demonstrate that tree-based approaches, which are more interpretable, achieve performance on par with or exceeding that of DNNs.
2) We provide quantitative and qualitative analyses of modalities and features, as well as cross-CD-tasks validation, that offer generalizable insights into CD detection.
3) We release all our preprocessed data, models, and code to support future research in this field.

II. RELATED WORK

Measurements for CD. Researchers have used a variety of indicators to measure driver cognitive load, including electroencephalography (EEG), heart rate variability, and galvanic skin responses [9], [10]. Although wrist-worn or ring-type wearables are becoming more popular for vital signs

All authors are with the Toyota Research Institute, USA. This work reflects solely the opinions and conclusions of its authors, and not TRI or any other Toyota entity. *Corresponding author. Email: firstname.lastname@tri.global.

monitoring, requiring a driver to wear dedicated devices can limit real-world applicability. In contrast, gaze direction and pupil size can be observed remotely without wearing anything, making gaze- and pupillometry-based measures promising for in-vehicle use.

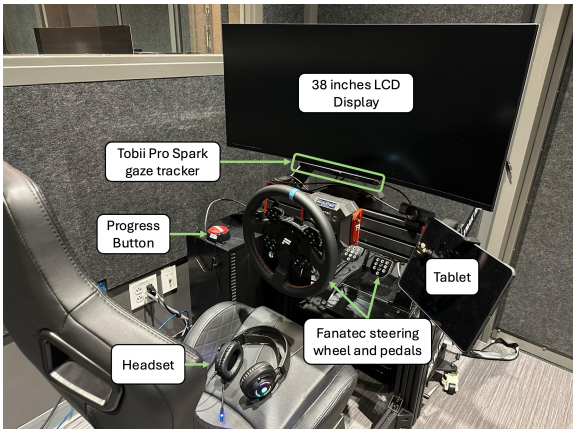
1) Pupillometry: The iris is a unique tissue in the eye controlled by the interaction of different parasympathetic and sympathetic pathways [11]. Pupillometry, which is the measurement of pupil diameter, has been investigated to better understand the internal state of individuals. Pupil dilation can be caused by different types of cognitive and mental activity, including mind wandering [12], attention [13] and surprise [14]. Pupil responses can be classified into *tonic responses*, which occur over a few minutes, and *phasic responses*, which occur briefly when faced with surprise or danger [15]. Additionally, *light reflexes* due to changes in ambient light also cause changes in pupil size, which makes it crucial to correctly interpret these different reactions [16].

2) Gaze: With regard to gaze data, it has been reported that secondary tasks reduce gaze movements related to the driving task [17]. Mind wandering will narrow down the visual attention [18]. Specifically, there is a tendency for *Gaze Transition Entropy (GTE)*—the complexity of gaze shifts—to partially decrease, and for *Gaze Stationary Entropy (GSE)*—the randomness of gaze distribution—to decrease overall [19].

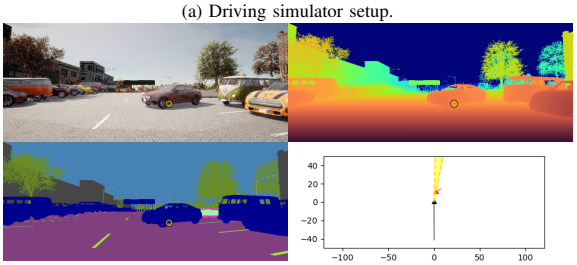
Machine Learning Methods Used for CD Detection. Representative examples of conventional machine learning methods include linear models such as Support Vector Machines (SVM) [20] and decision tree-based algorithms such as Gradient Boosting Machines (GBM), AdaBoost [21], and Random Forests [5]. These methods have been widely used because of their high interpretability, resistance to overfitting, and low computational cost. Typically, these methods classify data using basic summary statistics (e.g., mean, standard deviation), but they are said to have limitations in detecting complex time-series patterns [22]. Neural networks, which simultaneously perform feature extraction and classification, have shown good performance in many fields, and been used previously to detect CD [23]. However, they also face challenges with respect to interpretability [8], [24] and computational cost.

III. DATASET

To support our study, we leveraged the Impaired Driving Dataset (described in [25]), a driving simulator dataset collected using *CARLA* [26]. This dataset studies drivers in an urban and suburban setting (Town 15 in CARLA) under conditions of alcohol impairment, CD, and with added road hazards. For this study, we used only the CD-related subset, excluding segments specific to the study of alcohol impairment. We briefly summarize the relevant details of the dataset below. A total of 52 participants (age: 23-65, 31 identified as male, 18 as female, 2 as non-binary) with normal or corrected-to-normal took part. All held an active driver's license.



(a) Driving simulator setup.



(b) The recorded virtual sensors

Fig. 1: (a) The simulator setup from [25], including steering wheel, pedals, tablet, and progress button to advance through the study. Sensors included a Tobii Pro Spark eye tracker and a headset microphone. (b) The recorded virtual sensors (left-to-right, top-to-bottom: RGB image, depth image, semantic segmentation) as well as a birds-eye-view image of the vehicle position driver gaze/FOV. Participant's gaze target is shown in a yellow circle. An example is included in the supplementary video.

A. Simulator Setup

The simulator setup is shown in Figure 1, comprising a 38-inch ultra-wide display (60 Hz, 87 cm in width, 37 cm in height). The distance between the participants and the display was approximately 65 cm. The dataset includes gaze data collected using a *Tobii Pro Spark* [27] installed at the bottom of the display. Participants listened to engine sounds from the simulator and task audio via a headset with a microphone.

B. Cognitive Distraction (CD) Tasks

The dataset subjected participants to two forms of CD: a verbal *n-back task* and a verbal *statement task* [5]. In both tasks, participants listened to audio through the headset and their spoken responses were recorded. In the n-back task, a single-digit number was played every 2.5 seconds, and participants were asked to remember and respond with the previously presented number (1-back). In the statement task, participants listened to a short sentence and responded *subject*, *object*, and *Yes or No* about the plausibility of the sentence. The statement task does not have a fixed period, as it depends on the length of sentence. The reason for using two different tasks was to avoid the risk of overfitting to a specific task and to verify the generalization performance of CD detection. The n-back task was chosen because of its widespread use,

while the statement task was adopted as a more naturalistic task, resembling in-vehicle conversations. Both tasks relied solely on auditory input, minimizing interference with gaze- and pupil-based metrics. Instructions and examples of the CD tasks are included in the supplementary video.

C. Collection Procedure

After a practice session, participants completed two rounds of driving, each consisting of four drives, resulting in a total of eight drives. Each participant performed four drives without any CD task, two with the n-back task, and two with the statement task. Route order and task assignment were varied to ensure balanced coverage.

The dataset used in this study included 52 participants: 32 completed all routes under sober conditions, while the remaining 20 completed only the first round under sober conditions, with the second round conducted under alcohol-impaired conditions and excluded from this study.

Each route started from a predetermined starting point, and participants proceeded by following direction signs installed on the map, driving to a specified goal. The routes included speed limit signs of 30 mph and 50 mph, bumps, crosswalks, intersections, and roundabouts. Participants were instructed to drive normally, adhering to these elements. The driving time was set to be approximately 3 minutes when following the speed limits, but the actual driving duration varied depending on the participant's driving speed. Other vehicles driven by AI were present on the routes. AI vehicles randomly spawned around the participant's vehicle, so each participant encountered different patterns. This created complex tasks, such as merging and yielding to traffic in roundabouts.

IV. METHOD

A. Feature Extraction

1) Raw and Derived Modalities: In addition to raw sensor time-series signals (pupil diameter, gaze coordinates, and steering wheel angle), we calculated additional derived time-series signals such as fixation saccade ratio, gaze entropy, and attention to scene, which are reported to be relevant to CD, to investigate their contributions for detection.

PupilM: Mean pupil diameter of both eyes.

GazeRX and **GazeRY**: Gaze x and y coordinates of the right eye on the screen. We use only the right eye's gaze to avoid two issues: (1) using both eyes separately can cause multicollinearity, and (2) combining them (e.g., by averaging) can introduce artifacts, since even after calibration, residual offsets between eyes can create "phantom" saccades when one eye is briefly obstructed during steering.

Steering wheel angle: We added steering wheel angle to see the benefit used alone or combined with other pupil and gaze modalities.

Fixation Saccade Ratio: Fixation refers to a state where the gaze point remains within a certain area. During this time, the gaze hardly moves, and visual information is processed in the brain. We set the fixation flag to 1 when the gaze remained within an area corresponding to 2 degrees [28] of visual angle for at least 100 ms, and 0 otherwise. Saccades are rapid eye movements that shift the gaze from one point to another, serving as quick transitions to the next fixation point. A saccade starts when the maximum acceleration exceeds $8000\,\mathrm{deg/s^2}$ or the maximum velocity exceeds $30\,\mathrm{deg/s}$ [29], and ends when it falls below these thresholds. Saccades lasting less than 5 ms were ignored. Saccade is a binary flag similar to the fixation flag. Fixation and saccades are complementary, and their ratio is used as an indicator of a person's gaze pattern. In this study, the fixation saccade ratio is defined as the total duration of fixations divided by the total duration of saccades over a five second window. Fixation and saccade calculations were performed using *PyGaze Analyzer* [30].

Gaze Entropy: Gaze entropy consists of Gaze Transition Entropy (GTE) and Gaze Stationary Entropy (GSE). GTE and GSE are methods proposed by [31] to quantify gaze transitions. GTE models gaze switches between several Areas of Interest (AOIs) as Markov chain processes and calculates entropy to quantify the complexity of gaze movement patterns. Higher GTE indicates more random and complex gaze movements, associated with more exploratory visual attention. GSE measures how evenly a person's visual attention is distributed among AOIs. Higher GSE means the person is looking at all AOIs evenly, while lower GSE indicates concentration on certain areas. In this paper, we defined AOIs as five regions in a 5-row, 1-column division on the screen and calculated GTE and GSE over the past 5 seconds. The gaze entropy was calculated using [32].

Gaze Depth and **Look Obj**: These features were introduced to obtain information about the object of attention. Gaze Depth is the distance to the gaze point in the simulator. For example, looking at a nearby car rather than a distant road is expected to increase driving load and affect pupil size and gaze patterns. **Look Obj** consists of *lookRoad*, *lookPedestrian*, and *lookCar*, that are binary signals whether participants look at a particular category in the simulator. These features were calculated at 10 fps and linearly interpolated to 60 fps.

All time-series data were normalized. For the binary data of lookRoad, lookCar, and lookPedestrian, we subtracted 0.5. For other continuous values, Z-normalization was performed for each participant.

While the Transformer model described in Section V-B3 directly takes the normalized time-series data as input, all other models (i.e., linear models, tree-based models, and TabNet) described in Section IV-C have the same feature calculation process as described in the next Section IV-A2.

2) Feature Calculation: We used the *tsfresh* (time-series Feature Extraction based on Scalable Hypothesis tests) library [33] to compute comprehensive time-series features such as statistical measures, entropy, and autocorrelation coefficients, capturing detailed characteristics of physiological signals. Furthermore, we used the SARIMAX model (Seasonal AutoRegressive Integrated Moving Average with eXogenous factors) [34] to extract features representing longer-term temporal dependencies in the data. *tsfresh* and SARIMAX generate a 783-dimensional feature vector from each one-

dimensional signal.

After that, non-informative features that were found to be insignificant in the training data split were eliminated using the `select_features` function in *tsfresh* package. Furthermore, features with a correlation coefficient of 0.9 or higher with other features are also removed to avoid multicollinearity.

B. Data Split and Chunking

To ensure generalization across participants, we used a *between-subjects* split, assigning each participant exclusively to either the training, validation, or test set in a 3:1:1 ratio.

For each drive, 20-second chunks were extracted by sliding with a stride of 3 seconds. The first 5 seconds and the last 1 second of each drive were excluded. This is because features such as GTE do not exist for the first 5 seconds due to the calculation window. The last 1 second was also excluded to avoid potential edge effects that may arise during the synchronization of features calculated at different intervals in Section IV-A1.

For each chunk, the label was set to 0 for *no task* drives and 1 for drives with *n-back* or *statement* tasks. As a result, we obtained 31 participants and 12,709 samples for training, 10 participants and 3,303 samples for validation, and 11 participants and 4,656 samples for testing.

C. Model Training

We trained and compared the performance of linear models, decision tree-based models, and neural network models.

1) Linear Models: We evaluated logistic regression and Support Vector Machines (SVM) [35]. Logistic regression predicts the probability of an event occurrence using linear relationships and is widely used in binary classification problems. SVM enhances classification accuracy by mapping data into a high-dimensional space and defining classification boundaries, making it suitable for classification tasks requiring nonlinear separation.

2) Decision Tree-Based Models: Decision tree-based models have several advantages over linear models and neural networks. They excel at effectively capturing nonlinear relationships and interactions between features, and since they do not require data scaling adjustments, preprocessing is relatively straightforward. Additionally, decision tree-based methods are highly interpretable, making it easy to visualize important features and understand interactions. In this study, we compared and evaluated three decision tree-based methods: LightGBM [36], CatBoost [37], and XGBoost [38]. LightGBM is efficient in handling high-dimensional data, with fast training speed due to its leaf-wise growth, making it suitable for large datasets. CatBoost can utilize categorical variables without encoding and demonstrates excellent performance in handling feature interactions while improving classification accuracy; it also offers overfitting suppression. XGBoost has robust missing value handling and regularization functions, making it easy to adjust for higher accuracy and offering high computational efficiency.

3) Neural Network Models: Neural networks have superior ability to capture complex nonlinear relationships and advanced interactions between features compared to linear models and decision tree-based models. They are also strong in automatically extracting features from large amounts of data, and deep learning models with multi-layer structures can effectively learn latent patterns in data.

We examined the *TabNet* model [39] – a unique neural network that provides automatic feature selection and interpretable outputs from data, effectively extracting important features through gradient-based filtering. It excels at capturing nonlinear interactions between features, making it suitable for processing high-dimensional and unstructured data.

Furthermore, to capture feature interactions between multiple signals, we designed a network using a *Transformer Encoder* [40] with an attention mechanism and a *Classification Head*. The Transformer Encoder consists of 4 heads, 3 layers, and a model dimension of 64. The Classification Head consists of two fully connected (FC) layers, outputting a scalar logit. The input time-series data is first converted to the model dimension through a linear layer, and after adding Sinusoidal Positional Encoding, it is input into the Transformer Encoder. The intermediate representation vector is then passed to the Classification Head, resulting in the final logit output.

V. EXPERIMENTS

We conducted experiments to compare the performance of interpretable conventional models against the CD detection performance of a modern DNN model, and to understand the effect and importance of different features and the ability of the models to generalize across CD tasks. We use the area under the receiver operating characteristic curve (ROC AUC), a threshold-independent metric, as our evaluation measure. We then analyzed important features relevant to CD detection (Section V-D). Note that all experiments were performed using a between-subjects split.

A. Comparison of Modality and Feature Extraction

We compare each modality (raw sensor time-series data and derived time-series data) and the benefit of combining multiple modalities, when using LightGBM model trained and tested on (no task + n-back task) subset. The ROC AUC scores for each modality are shown in the left part of Figure 2. The results show that *GazeRY* and *GazeDepth* are particularly useful in identifying CD. Following these, *GazeRX*, *Pupil*, and the *Fixation-Saccade Ratio* (FSR) showed high contributions. The right part of Figure 2 shows the benefit of combining multiple modalities. *GazeRaw* consist of *GazeRX* and *GazeRY*. *GazeRaw+Derived* added *FSR* and *GazeEntropy* to *GazeRaw*. *+Ext* added *GazeDepth* and *LookObj*. Adding the remaining *Steer* feature becomes *All*.

The key findings are that *Derived* features (*FSR* and *GazeEntropy*) and *Pupil* improved the detection when combined with Gaze, although they have less performance than Gaze when used independently. In contrast, *Ext* and *Steer*

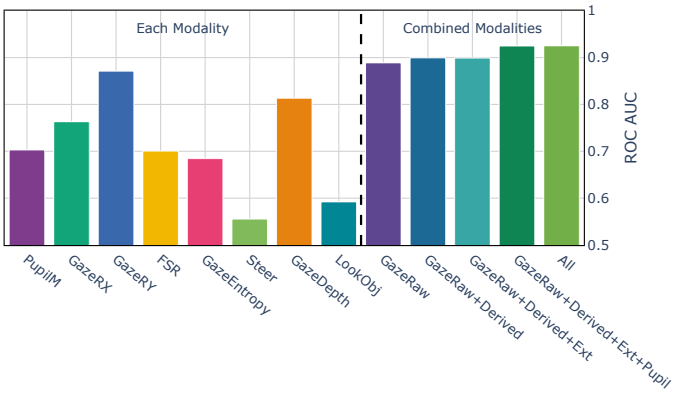


Fig. 2: The area under the receiver operating characteristic curve (ROC AUC) scores of LightGBM models trained on features extracted from each modality and combined modalities. The models were trained with (no task + n-back task) subset in training split, then tested on the same task (no task + n-back task) subset in test split. *GazeRaw* consists of *GazeRX* and *GazeRY*. *Derived* consists of *FSR* and *GazeEntropy*. *Ext* means *GazeDepth* and *LookObj*.

TABLE I: Performance comparison of different time-series feature calculation methods. Feature sets become increasingly expressive from (1) basic stats [mean, std], to (2) added moments [skewness, kurtosis], and (3) full descriptors via *tsfresh* and SARIMAX. The third method, which captures a wider range of temporal characteristics, outperforms the commonly used simpler calculations.

Modality [Feature calculation]	ROC AUC
All [mean, std]	0.869
All [mean, std, skewness, kurtosis]	0.899
All [tsfresh, SARIMAX]	**0.925**

didn't contribute further to performance. Notably, *GazeDepth* had high detection performance even when used alone.

In addition to modality comparison, we also evaluate the benefit of the comprehensive feature extraction methods using *tsfresh* and SARIMAX from time-series data, along with commonly used statistical values such as mean and standard deviation [5] in Table I. Performance improved as features became more expressive.

B. Comparison of Models Using All Modalities

Table II compares a wide range of machine learning models. They are trained on the (no task + n-back task) subset and tested on both of (no task + n-back) and (no task + statement task) subsets.

1) Linear Models: Linear models were less performative than decision trees and DNNs, similar to [5]. SVM achieved the lowest performance despite its ability to capture nonlinear decision boundaries through kernel functions, suggesting that nonlinear separability alone did not suffice to capture the subtleties of CD.

2) Gradient Boosting Decision Trees: For decision tree-based gradient boosting models, we used LightGBM, XGBoost, and CatBoost. Each model demonstrated high performance, with CatBoost achieving the highest ROC AUC in both the n-back and the statement tasks. These results confirm that decision tree-based models capture complex interactions and outperform other models.

TABLE II: Performance (ROC AUC) comparison of multiple machine learning models across Linear Models (LM), Gradient-Boosting Decision Trees (GBDT), and Deep Neural Networks (DNNs). All the models were trained on (no task, n-back task) subset of training split. The n-back row shows the test results on the same task (no task, n-bask task). The statement row shows results on the different task (no task, statement task). Both are tested on test split.

	LM		GBDT			DNN	
	Log. Reg.	SVM	Light	Xg	Cat	TabNet	Trans.
n-back	0.875	0.775	0.925	0.928	**0.931**	0.892	0.910
state.	0.691	0.599	0.724	0.727	**0.732**	0.724	0.738

3) Deep Neural Networks (DNNs): As neural network models, we examined TabNet and Transformer. TabNet showed an AUC of 0.892 in the n-back task and 0.724 in the statement task, while the Transformer achieved 0.910 and 0.738, respectively. These results suggest that neural networks are particularly adept at capturing complex features in multivariate time-series data. However, in the current setup, the Gradient Boosting Decision Trees (GBDT) models showed comparable or higher performance than DNN models. A likely explanation is that our dataset size may be insufficient for neural networks to fully leverage their representational capacity.

Although the following studies do not directly compare tsfresh+GBDT with Transformer-based approaches, Shwartz-Ziv and Armon [41] reported that XGBoost consistently outperformed deep learning models across a variety of tabular datasets, often requiring less hyperparameter tuning. Similarly, [42] demonstrated that feature extraction using XGBoost achieved better performance than a deep learning model for time-series classification of ECG signals. These findings suggest that GBDT-based methods can be highly competitive, particularly in scenarios with limited training data, as is the case in our study.

C. Cross-Task Comparison

In the previous Section V-B, a decrease in detection performance was observed when applying models trained on the n-back task to the statement task across all methods. This suggests that not only domain gaps but also differences in cognitive load between the tasks may influence model performance. Several participants reported after the experiment that they found the n-back task more difficult than the statement task.

To verify this, we conducted additional experiments. We evaluated the ROC AUC for all four possible combinations of training and testing on the (no task + n-back task) and (no task + statement task) subsets (i.e., a 2×2 setting). The results are presented in Table III. The table shows that regardless which task was used for training, models achieved high performance exceeding 0.9 when tested on the (no task + n-back task) subset. This suggests that the models did not merely capture artificial task-specific patterns (e.g., the 2.5-second periodic fluctuations in the n-back task) but rather learned general features inherent to CD. In contrast, when tested on the (no task + statement task) subset, the performance remained in the

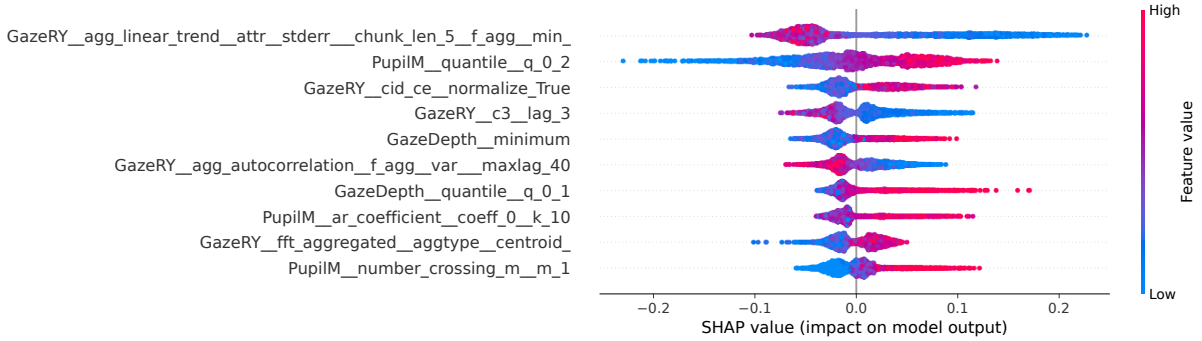


Fig. 3: A SHAP summary plot of the top 10 features impacting the model's output for CD detection. The vertical axis shows feature names; the horizontal axis shows the corresponding SHAP values (positive means an increased likelihood of distraction, negative means the opposite). Each dot represents an individual sample, colored by the raw feature value (red = high, blue = low). For example, a feature with many red points on the right side suggests that higher feature values tend to increase the model's output in the direction of predicting CD. Furthermore, a wide distribution indicates that the feature's impact varies greatly among samples, while a concentrated distribution suggests that the feature's impact is similar across samples. This plot demonstrates how variations in feature values influence the model's predictions across all samples.

TABLE III: Cross-task performance (ROC AUC) comparison using Cat boost.

		Test	
		n-back	statement
Train	n-back	0.931	0.732
	statement	0.914	0.771

0.7 range regardless of the training task. This implies that the cognitive load in the statement task is lower, making detection more challenging.

As a further evidence, we compared pupil diameters between the two tasks. For each participant, we calculated the mean pupil diameter during the n-back and statement tasks and conducted a paired Wilcoxon signed-rank test. The results showed a p-value of $1.7 \times 10^{-4} < 0.05$) and an effect size of -0.52, confirming a significant difference. The large negative effect size indicates that pupil diameter during the statement task was smaller than during the n-back task, supporting the notion that the cognitive load in the statement task is lower.

D. Analysis of Important Features

In this section, we explore important features to detect CD leveraging a LightGBM model trained in Section V-B.

1) Analysis of Independent Feature Effect: We employed SHAP (SHapley Additive exPlanations) [43] to quantify how each feature affects the model's predictions of CD. Figure 3 highlights the top 10 contributing features, revealing whether higher or lower values push the output toward the distracted or non-distracted state. Based on this analysis, we detail the top five features below:

GazeRY__agg_linear_trend__attr_stderr__chunk_len_5__f_agg_min: This feature represents the minimum standard error of linear fits over 5-frame (0.083 seconds) segments of *GazeRY*. A larger value means *GazeRY* is consistently nonlinear across the chunk (tending *not* to be CD), while a smaller value indicates some segments are more linear (linked to CD).

PupilM__quantile__q_0.2: This is the 20% quantile of the pupil size, corresponding to the baseline level amid fluctuations due to stimuli and noise during driving. A higher value increases the likelihood of predicting CD. This likely captures the tonic response, where the pupil remains enlarged continuously with increased cognitive load.

GazeRY__cid_ce__normalize_True: The cid_ce stands for complexity-invariant distance complexity estimate. This is an efficient way to estimate complexity proposed in [44]. The value is calculated by Equation (1).

$$CID_{CE} = \sqrt{\sum_{i=1}^{n-1}(x_i - x_{i-1})^2} \tag{1}$$

where n is the number of time-series data samples and x is the time-series value (*GazeRY*). This feature value increases with greater signal variation, but is normalized by the amplitude of the time-series to allow comparison across different scales.

GazeRY__c3__lag_3: This is another measure of nonlinearity in *GazeRY* [45]. A larger (more nonlinear) value tends to correlate with a *not* distracted (non-CD) state, similar to the first feature's trend.

GazeDepth__minimum: This feature represents the minimum distance to the gaze target within the 20-second chunk. A high value (i.e., continuously looking at distant objects for 20 seconds) suggests CD. Conversely, even a brief glance at something close (including the speedometer) reduces this feature's value and indicates *not* being CD.

From the analysis above, we observed the following trends:

- More complex and nonlinear vertical gaze movements (*GazeRY*) tend to indicate *not* being CD.
- The higher the baseline of pupil diameter (lower quantiles of *PupilM*), the more it acts in the direction of CD.
- Continuously gazing at distant areas for 20 seconds acts in the direction of CD.

These results deepen our understanding of how each feature can be extracted and how they contribute to CD detection, providing important guidance for future model development.

2) Feature Combination Analysis: In this section, we evaluate the effects of feature combinations. We employed Sobol

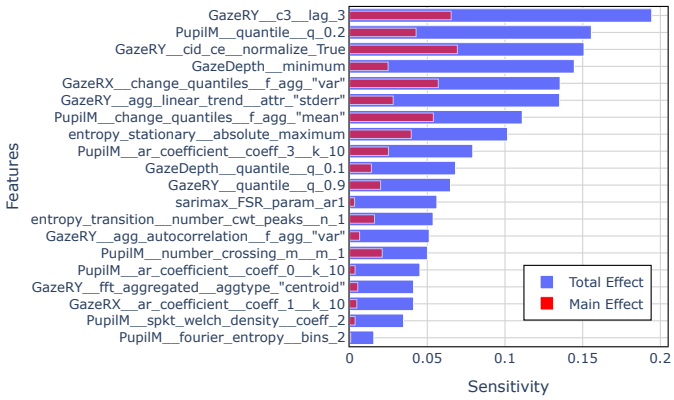


Fig. 4: Top 20 features ranked by Sobol sensitivity analysis. Long feature names are truncated. The total effects are shown in blue bars, while main effect are overlaid by red bars. Total effects and main Effects shows a notable gap for each feature. This indicates that interactions among other features contribute significantly to the output, highlighting the importance of feature combinations over individual features.

sensitivity analysis, a variance-based method that quantifies the contribution of each input variable to the variance of the model output [46]. Sobol analysis allows us to evaluate both the main effects—the variance contributed by each input individually—and the total effects, which include the effects due to interactions with other variables.

The analysis was applied to a LightGBM model retrained on the top 20 features from the previous section. Results are shown in Figure 4, with blue bars for total effects and red bars for main effects. The consistent gap between them indicates that interactions between features play a significant role. Note that feature rankings may differ from earlier figures due to differing importance metrics.

VI. DISCUSSION

The results of this study showed that combining multiple physiological signals and features improves the CD detection. Decision tree-based models showed superior performance, which can be attributed to their ability to effectively capture nonlinear relationships and interactions among features. Furthermore, while the dataset is relatively large compared to similar studies, it is still small compared with datasets often used for training deep neural networks. As such, simpler tree-based models may also help avoid overfitting.

Feature importance analysis revealed that the nonlinearity of vertical gaze movements, baseline pupil diameter, and distance to the point of gaze are important indicators of CD. Specifically, during CD, the nonlinearity of the driver's vertical gaze movement decreased, suggesting that gaze movements become less responsive to stimuli from the complex driving task. Additionally, a tendency to look at more distant areas was observed, corresponding to the phenomenon of a narrowed field of view during CD. Furthermore, the increase in baseline pupil diameter reflects sustained cognitive load, aligning with tonic pupil responses reported in previous studies.

The Sobol sensitivity analysis emphasized the importance of interactions among features, indicating that considering individual features alone is insufficient for effectively detecting CD. This underscores the necessity for models capable of capturing the complex dependencies among multiple physiological signals.

While DNN models also demonstrated high performance, they did not outperform decision tree-based models in our experiments. This may be due to factors such as dataset size and differences in network architectures. Using larger datasets and optimizing network architectures could potentially yield further improvements and insights. However, collecting large-scale human driving data, particularly under CD conditions, presents practical challenges. Therefore, striking a balance between data availability and model expressiveness will be essential for advancing CD detection in real-world applications.

Limitations. The data used in this study is relatively realistic due to the diverse road topology and AI-controlled road agents. However, there remains a gap between the simulated environment and real-world conditions. For instance, differences in the field of view, vehicle behaviors, and ambient lighting that affects pupillary light reflex could introduce discrepancies. Furthermore, the identified features are at a low level, and for practical applications in real vehicles, further validation with other data sources is essential. Additionally, integrating these features through hierarchical clustering and linking them to physical interpretations is necessary steps to enhance their applicability and robustness for real-world vehicle deployment.

VII. CONCLUSION

This study investigated cognitive distraction (CD) detection using multiple modalities, focusing on pupil diameter and gaze patterns, from 52 participants in a simulator. We extracted features using *tsfresh* and SARIMAX methods and evaluated multiple machine learning models, including linear models, decision tree-based models, and neural networks.

Our results demonstrated that combining multiple modalities, especially gaze, pupil and derived modalities such as fixation saccade ratio, and extracting comprehensive time-series features improves the detection performance. With this approach, decision tree-based models such as CatBoost achieved the highest performance with an AUC of up to 0.931 in the same task as trained (n-back) and 0.732 in the different task (statement), performing on par with or better than a Transformer model while maintaining explainability. A cross-task comparison further revealed that models trained on one task can generalize to another, although performance decreases for a task with lighter cognitive load. Feature analysis using SHAP and Sobol revealed that complex gaze patterns in vertical direction, higher baseline pupil diameter and distance to the gaze target are related to cognitive distraction. In addition, the results highlight the importance of combining various features. These findings suggest that integrating multiple modalities / features and employing models capable of capturing nonlinear interactions are effective strategies for detecting cognitive distraction. Future work could extend this research beyond chunk-level classification to continuous monitoring over time

to facilitate more robust and timely driver alerts in practical in-vehicle scenarios.

Acknowledgments. We thank Laporsha Dees and Emily Sumner for their help with running human subjects trials, and Todd Rowell and Thomas Balch for supporting the test implementation.

REFERENCES

[1] A. Palao, R. Fredriksson, and M. Lenné, "Euro ncap's current and future in-cabin monitoring systems assessment," in *Proceedings of the 27th International Technical Conference on the Enhanced Safety of Vehicles (ESV) National Highway Traffic Safety Administration*, no. 23-0286, 2023.

[2] B. Hamilton and J. Grabowski, "Cognitive distraction: Something to think about," *AAA Foundation for Traffic Safety*, 2013.

[3] K. Kircher and C. Ahlström, "The driver distraction detection algorithm AttenD," vol. 1, pp. 327–348, Jan. 2013.

[4] S. Yang, J. Kuo, and M. G. Lenné, "Analysis of gaze behavior to measure cognitive distraction in real-world driving," in *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, vol. 62, no. 1. SAGE Publications Sage CA: Los Angeles, CA, 2018, pp. 1944–1948.

[5] A. Misra, S. Samuel, S. Cao, and K. Shariatmadari, "Detection of driver cognitive distraction using machine learning methods," pp. 18 000–18 012, 2023.

[6] N. Li, J. J. Jain, and C. Busso, "Modeling of driver behavior in real world scenarios using multiple noninvasive sensors," *IEEE Transactions on Multimedia*, vol. 15, no. 5, pp. 1213–1225, 2013.

[7] A. D. McDonald, T. K. Ferris, and T. A. Wiener, "Classification of driver distraction: A comprehensive analysis of feature generation, machine learning, and input measures," *Human factors*, vol. 62, no. 6, pp. 1019–1035, 2020.

[8] C. Rudin, "Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead," *Nature machine intelligence*, vol. 1, no. 5, pp. 206–215, 2019.

[9] A. Kashevnik, R. Shchedrin, C. Kaiser, and A. Stocker, "Driver distraction detection methods: A literature review and framework," https://ieeexplore.ieee.org/abstract/document/9405644, accessed: 2024-11-1.

[10] G. Prabhakar, N. Madhu, and P. Biswas, "Comparing pupil dilation, head movement, and eeg for distraction detection of drivers," in *International BCS Human Computer Interaction Conference*, 2018.

[11] S. Mathôt, "Pupillometry: Psychology, physiology, and function," *Journal of Cognition*, 2018.

[12] T. Čegovnik, K. Stojmenova, G. Jakus, and J. Sodnik, "An analysis of the suitability of a low-cost eye tracker for assessing the cognitive load of drivers," *Applied ergonomics*, vol. 68, pp. 1–11, 2018.

[13] V. L. Benitez and M. K. Robison, "Pupillometry as a window into young children's sustained attention," *Journal of Intelligence*, vol. 10, no. 4, p. 107, 2022.

[14] J. W. Antony, T. H. Hartshorne, K. Pomeroy, T. M. Gureckis, U. Hasson, S. D. McDougle, and K. A. Norman, "Behavioral, physiological, and neural signatures of surprise during naturalistic sports viewing," *Neuron*, vol. 109, no. 2, pp. 377–390.e7, 2021.

[15] S. Joshi and J. I. Gold, "Pupil size as a window on neural substrates of cognition," *Trends in cog. sci.*, vol. 24, no. 6, pp. 466–480, 2020.

[16] H. Yokoyama, K. Eihata, J. Muramatsu, and Y. Fujiwara, "Prediction of driver's workload from slow fluctuations of pupil diameter," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, Nov. 2018, pp. 1775–1780.

[17] M. Faber, R. Bixler, and S. K. D'Mello, "An automated behavioral measure of mind wandering during computerized reading," *Behavior Research Methods*, vol. 50, pp. 134–150, 2018.

[18] J. He, E. Becic, Y.-C. Lee, and J. S. McCarley, "Mind wandering behind the wheel: Performance and oculomotor correlates," *Human factors*, vol. 53, no. 1, pp. 13–21, 2011.

[19] C. M. Goodridge, R. C. Goncalves, A. Arabian, A. Horrobin, A. Solernou, Y. T. Lee, Y. M. Lee, R. Madigan, and N. Merat, "Gaze entropy metrics for mental workload estimation are heterogenous during hands-off level 2 automation," *Accident Analysis & Prevention*, vol. 202, p. 107560, 2024.

[20] Y. Liao, S. Li, G. Li, W. Wang, B. Cheng, and F. Chen, "Detection of driver cognitive distraction: An svm based real-time algorithm and its comparison study in typical driving scenarios," pp. 394–399, 2016.

[21] M. Miyaji, H. Kawanaka, and K. Oguri, "Driver's cognitive distraction detection using physiological features by the adaboost," pp. 1–6, 2009.

[22] M. Middlehurst and A. Bagnall, "The freshprince: A simple transformation based pipeline time series classifier," in *International Conference on Pattern Recognition and Artificial Intelligence*. Springer, 2022, pp. 150–161.

[23] A. Shajari, H. Asadi, S. Alsanwy, and S. Nahavandi, "Detection of driver cognitive distraction using driver performance measures, eye-tracking data and a d-ffnn model," pp. 2093–2099, 2023.

[24] Y. Zhang, P. Tiňo, A. Leonardis, and K. Tang, "A survey on neural network interpretability," *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 5, no. 5, pp. 726–742, 2021.

[25] J. Gideon, K. Tamura, E. Sumner, L. Dees, P. Reyes Gomez, B. Haq, T. Rowell, A. Balachandran, S. Stent, and G. Rosman, "A simulator dataset to support the study of impaired driving," 2025. [Online]. Available: https://toyotaresearchinstitute.github.io/IDD/

[26] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "CARLA: An open urban driving simulator," vol. 78, pp. 1–16, 2017.

[27] Tobii AB, "Tobii pro spark," https://www.tobii.com/products/eye-trackers/screen-based/tobii-pro-spark, accessed: 2025-04-22.

[28] S. Sivaprasad, E. Pearce, and V. Chong, "Quality of fixation in eyes with neovascular age-related macular degeneration treated with ranibizumab," *Eye*, vol. 25, no. 12, pp. 1612–1616, 2011.

[29] M. Heath, F. L. Colino, J. Chan, and O. E. Krigolson, "Visuomotor mental rotation of a saccade: The contingent negative variation scales to the angle of rotation," *Vision Research*, vol. 143, pp. 82–88, 2018.

[30] E. S. Dalmaijer, S. Mathôt, and S. Van der Stigchel, "PyGaze: An open-source, cross-platform toolbox for minimal-effort programming of eyetracking experiments," *Behavior research methods*, vol. 46, pp. 913–921, 2014.

[31] K. Krejtz, A. Duchowski, T. Szmidt, I. Krejtz, F. González Perilli, A. Pires, A. Vilaro, and N. Villalobos, "Gaze transition entropy," *ACM Trans. Appl. Percept.*, vol. 13, no. 1, Dec. 2015.

[32] H. Jundi, "Eyetrackingmetrics," https://github.com/Husseinjd/EyeTrackingMetrics, 2024, accessed: 2025-04-22.

[33] M. Christ, N. Braun, J. Neuffer, and A. W. Kempa-Liehr, "Time series feature extraction on basis of scalable hypothesis tests (tsfresh–a python package)," *Neurocomputing*, vol. 307, pp. 72–77, 2018.

[34] J. Perktold, S. Seabold, K. Sheppard, ChadFulton, K. Shedden, and jbrockmendel et al., "statsmodels/statsmodels: Release 0.14.2," Apr. 2024. [Online]. Available: https://doi.org/10.5281/zenodo.10984387

[35] Y. Liang, M. Reyes, and J. D. Lee, "Real-time detection of driver cognitive distraction using support vector machines," pp. 340–350, 2007.

[36] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, "LightGBM: A highly efficient gradient boosting decision tree," *Advances in neural information processing systems*, vol. 30, 2017.

[37] L. Prokhorenkova, G. Gusev, A. Vorobev, A. V. Dorogush, and A. Gulin, "Catboost: unbiased boosting with categorical features," *Advances in neural information processing systems*, vol. 31, 2018.

[38] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 2016, pp. 785–794.

[39] S. Ö. Arik and T. Pfister, "Tabnet: Attentive interpretable tabular learning," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 35, no. 8, 2021, pp. 6679–6687.

[40] A. Vaswani, "Attention is all you need," *Advances in Neural Information Processing Systems*, 2017.

[41] R. Shwartz-Ziv and A. Armon, "Tabular data: Deep learning is not all you need," *Information Fusion*, vol. 81, pp. 84–90, 2022.

[42] L. Bickmann, L. Plagwitz, and J. Varghese, "Benchmarking approaches: Time series versus feature-based machine learning in ecg analysis on the ptb-xl dataset," pp. 589–593, 2024.

[43] S. Lundberg, "A unified approach to interpreting model predictions," *arXiv preprint arXiv:1705.07874*, 2017.

[44] G. E. Batista, E. J. Keogh, O. M. Tataw, and V. M. De Souza, "Cid: an efficient complexity-invariant distance for time series," *Data Mining and Knowledge Discovery*, vol. 28, pp. 634–669, 2014.

[45] T. Schreiber and A. Schmitz, "Discrimination power of measures for nonlinearity in a time series," *Physical Review E*, vol. 55, pp. 5443–5447, 1997.

[46] I. Sobol, "Global sensitivity indices for nonlinear mathematical models and their monte carlo estimates," *Mathematics and Computers in Simulation*, vol. 55, no. 1, pp. 271–280, 2001, the Second IMACS Seminar on Monte Carlo Methods.