Non-intrusive Distracted Driving Detection Based on Driving Sensing Data

Sasan Jafarnejad[1], German Castignani and Thomas Engel[1]
Interdiscipl. Centre for Security, Reliability & Trust (SnT), University of Luxembourg, Esch-Sur-Alzette, Luxembourg
[1]{*firstname.lastname*}*@uni.lu, german.castignani@ext.uni.lu*

Keywords: distraction, inattention, driver behavior, machine learning, driver modeling

Abstract: Nowadays Internet-enabled phones have become ubiquitous, and we all witness the flood of information that often arrives with a notification. Most of us immediately divert our attention to our phones even when we are behind the wheel. Statistics show that drivers use their phone on 88% of their trips, in 2015 in the United Kingdom 25% of the fatal accidents were caused by distraction or impairment. Therefore there is need to tackle this issue. However, most of the distraction detection methods either use expensive dedicated hardware and/or they make use of intrusive or uncomfortable sensors. We propose a distracted driving detection mechanism using non-intrusive vehicle sensor data. In the proposed method 8 driving signals are used. The data is collected, then two sets of statistical and cepstral features are extracted using a sliding window process, further a classifier makes a prediction for each window frame, lastly, a decision function takes the last l predictions and makes the final prediction. We evaluate the subject independent performance of the proposed mechanism using a driving dataset consisting of 13 drivers. We show that performance increases as the decision window gets larger. We achieve the best results using a Gradient Boosting classifier with a decision window of total duration 285 seconds which yields ROC AUC of 98.7%.

1 INTRODUCTION

Nowadays Internet-enabled smart-phones have become ubiquitous, and we all witness the flood of information that often arrives with a notification. Most of us immediately divert our attention to our smart-phones regardless of what we are doing and frequently we do so even when we are behind the wheel. A recent study have shown that drivers use their phones in about 88% of their trips, and in average they spend in 3.5 minutes of each hour of driving on their phone [1]. Although mobile-phone related distractions are only a subset of all driving distractions, it has been proven that driving distractions are dangerous to passenger safety (Klauer et al., 2010). Recent statistics indicate that a large portion of fatal accidents are caused due to driver distractions. For example according to Department of Transport of the United Kingdom, in 2015 about 25% of fatal accidents were caused by distraction or impairment [2]. This implies that any solution that mitigates distracted driving even by a small amount can save lives. This motivates us to propose a distracted driving detection mechanism that is able to detect and warn the driver at such moments and can be available for the majority of car owners. Although this work is not the first attempt at solving this problem, it follows a novel and ambitious approach. Current literature mostly uses dedicated sensors to detect distractions, sensors such as cameras for tracking head orientation (Wöllmer et al., 2011) or measuring the skin temperature (Wesley et al., 2010), however we propose to only use the standard vehicle sensors, therefore this method could be applicable to the most of the commercial vehicles on the market.

In this approach we do not use any intrusive sensors such as cameras or microphones, instead we focus on car driving data. We aim to classify driving segments as distracted or not distracted driving. Such a system can be used on-line to alert the driver in case of continuous distraction. Or alternatively can be employed off-line as a metric to judge the driving performance or for risk assessment of the driver. First we use a sliding window to extract features from driving signals, then using machine learning we classify each window as distracted or not-distracted, then we use a decision function to decide whether a sequence of window frames represent distracted driving

[1]http://blog.zendrive.com/distracted-driving/

[2]https://www.gov.uk/government/statistics/reported-road-casualties-great-britain-annual-report-2015

or not. We propose to validate this method using driving traces from 13 drivers chosen from a large driving dataset called UYANIK (Abut et al., 2007). For evaluation we present the performance of the proposed method in terms of F_1 score and area under the Receiver Operating Characteristic curve (ROC AUC).

The remainder of the paper is organized as follows, at Section 2 we present an overview of the prior work on distraction detection that is relevant to this work, in Section 3 we describe the dataset used for validation, next in Section 4 we describe our proposed methodology and the evaluation strategy. In Section 5 we present the evaluation results and in Section 6 we discuss the obtained results. Lastly in Section 7 we conclude and present the future directions.

2 RELATED WORK

Researchers have been trying to reach a consensus for the definition of distracted driving however although there have been many attempts at formalizing distracted driving there are still discussions around this subject. Inattention is a major cause of unsafe driving and accidents, however inattention itself can be caused by various means. According to the latest driver inattention taxonomy from the United States and European Union Bilateral Intelligent Transportation Systems Task Force (Engström et al., 2013) driver drowsiness and driver distraction are two major processes that give rise to inattention. There is a large body of research in the literature attending to both drowsiness and distraction detection, however in this section we focus on distraction detection as it is also the aim of this work. According to Lee et al. driver distraction is: "Driver distraction is a diversion of attention away from activities critical for safe driving toward a competing activity." (Lee et al., 2009). Distraction affects the act of driving in various ways, Dong et al. have categorized these effects into three categories (Dong et al., 2011):

1. Driver Behavior Patterns - This category covers patterns of actions such as rear-view mirror checks or forward view inspection activities during driving. Harbluck et al. show that drivers engaged in cognitively difficult tasks, reduce their visual monitoring of the mirrors and instruments to the extent that some entirely abandon these tasks (Harbluk et al., 2007). It is also shown that compared to low workloads under medium and heavy cognitive workloads the average visual field area reduced to 92.2% and 86.41%, respectively (Rantanen and Goldberg, 1999). These patterns are difficult to measure and in most cases require the use of intrusive sensors such as cameras.

2. Physiological Responses - Physiological indicators such as Electroencephalography (EEG), Electrocardiography (ECG) signals, skin conductance and blinking rate are in this category. It has been shown that EEG workload increased with working memory load and problem solving tasks (Berka et al., 2007). In a more recent work researchers demonstrated that visual and cognitive distraction lead to temperate increase on the skin surface (Wesley et al., 2010). The main issue with such metrics is the need for dedicated sensors which are also uncomfortable or intrusive.

3. Driving Performance - Maintaining speed and lane keeping are two examples of driving performance metrics that are affected by distractions. Zhou et al. show that performing secondary tasks influences checking behavior (e.g. mirror checking) both in frequency and duration, which leads to lower rate of lane changing (Zhou et al., 2008). In another study Liang and Lee establish that cognitive distraction makes steering less smooth. Steering neglect and over-compensation was associated with visual distraction and under-compensation with cognitive distraction (Liang and Lee, 2010). Although advanced sensors are required to measure metrics such as headway distance or lane-changing behavior, such sensors are not intrusive towards the driver and they are becoming more prevalent in the recent vehicles on the market.

Driver distraction literature has focused on only one or a hybrid of the above mentioned categories, however most of the studies take one of the first two approaches. This could be because they fit better in already established fields of medical and behavioral research. In addition such metrics are already well defined and they have better reliability in distraction detection. Here we present a few examples from distraction detection methods that are more relevant to this work. Wöllmer et al. proposed to use head orientation to detect distractions, particularly they detect user interactions with the instrument cluster. In addition to that they also employ some of the vehicle operation signals such as pedal and steering wheel data as well as some driving performance metrics such as deviation from the middle of the traffic lane and heading angle (Wöllmer et al., 2011). This rich set of signals then were used as input to a long short-term memory (LSTM) recurrent neural network, which results in a subject-independent detection of distraction with up to 96.6% accuracy.

In a more relevant study Jin et al. develop two models based on support vector machine (SVM) called NLModel and NHModel, which are designed to detect low and high cognitive distractions respectively. In this work they only use data from the vehicle's

CAN-bus as input. Signals include vehicle operating parameters as well as vehicle dynamics. They achieve accuracy of about 73.8% and sensitivity of approximately 61.8% (Jin et al., 2012).
In another study Tango and Botta propose a subject dependent model based on SVM that can achieve accuracies of up to 95%. They only use the dynamic signals of the vehicle for classification however data annotation is done accurately with the help of eye-tracking cameras and human supervision. The experiments are performed in mostly straight highway drive and at the speeds of around 100km/h. This over simplification may question robustness and reliability of the model in real world environments (Tango and Botta, 2013).
Lastly Özturk et al. propose the use of Gaussian Mixture Models (GMM) for distraction detection. This work is important because they use a similar dataset as this work, therefore their results can be comparable and could be taken into account as baseline for our work. They extract cepstral features from the pressure on Gas and Brake pedals. Using 16-mixture GMM classifier and a decision window of length 360 seconds they achieve 93.2% success to recognize non-distracted driving and 72.5% success in recognizing distracted driving (Öztürk and Erzin, 2012).

3 DATASET

For this study we use a data-set called UYANIK. This data-set is the result of an international consortium comprised of NEDO (Japan) and Drive-Safe (Turkey) (Abut et al., 2007; Miyajima et al., 2009). It is aimed at signal processing applications to enhance driving experience. Partner universities developed and deployed sensor-equipped vehicles sharing common requirements to collect data on driving behavior under various driving conditions. Sabanci University of Turkey, under the shared framework laid jointly by the partners; Equipped a Renault Megane with various sensors to measure dynamic state of the vehicle and its surrounding environment. Cameras installed to capture drivers reaction and road traffic. Microphones capturing the conversations carried on inside the vehicle. Inertial Measurement Unit (IMU) and CAN-bus data was recorded to capture vehicle dynamics and internal state of the car. In addition, sensors were installed underneath the Brake and Gas pedals to closely monitor driver reactions. The complete list of available sensor signals is presented in Table 1.
To collect data, experiments designed to study how people drive while performing various secondary tasks. Data collection is done in Istanbul, Turkey, and consists of a 25 km long stretch which includes a short ride inside university campus, a city traffic driving, motorway traffic driving, a dense city traffic driving and, finally, the way back to the point of departure. A typical trip lasts about 45 minutes. The whole journey is divided in 4 segments, denoting different secondary tasks: a) Reference Driving b) Query Dialogue c) Signboard Reading and Navigation Dialog d) Pure Navigational Dialog (Refer to (Abut et al., 2007) for more information.).

3.1 Synchronization and Annotation

Each driving session in the dataset is composed of files containing data for each data source, that is Location, CAN-bus, IMU, Laser range-finder, as well as audio and video files. Although data files contain synchronized timestamps, they all have different sampling rates and fusing this data together is a cumbersome task. Since in this work our aim is to build a model to detect distracted driving, we need to have our dataset annotated so it can be used to train machine learning models. Here we face two challenges a) Despite all the efforts of the research community identifying a driver as distracted is extremely subjective. b) our video or audio files are not synchronized with the sensor measurements, and without having the two synchronized it is impossible to use the dataset for our intended purpose. To tackle the synchronization issue we took inspiration from (Fridman et al., 2016). The general idea is that in a moving vehicle, rotations in steering wheel results in lat-

Table 1: Sensor Data Available in UYANIK.

Channel	Source	Details
Video facing the driver	Retrofitted	15 fps 480x640
Video facing the road	Retrofitted	15 fps 480x640
Driver microphone	Retrofitted	16 KHz 16-bit
Rear-view microphone	Retrofitted	16 KHz 16-bit
Cellphone microphone	Retrofitted	16 KHz 16-bit
Steering wheel angle	CAN-Bus	32 Hz degrees
Steering wheel rel. speed	CAN-Bus	32 Hz °/s
Vehicle speed (VS)	CAN-Bus	32 Hz km/h
Individual wheel speeds	CAN-Bus	32 Hz km/h
Engine RPM (ERPM)	CAN-Bus	32 Hz, rpm
Yaw rate (YR)	CAN-Bus	32 Hz
Clutch state	CAN-Bus	32 Hz, 0/1 state
Reverse gear	CAN-Bus	32 Hz, 0/1 state
Brake state	CAN-Bus	32 Hz, 0/1 state
Clutch	CAN-Bus	32 Hz, 0/1 state
Brake & Gas Pedal Pressure	Retrofitted	Kg-force/cm^2
XYZ directional acc.	IMU	10 Hz
Laser rage-finder	Retrofitted	1-2 Hz, 181°

eral movements of the vehicle. These lateral movements are visible in the front facing camera feed and they should correlate with steering wheel movements. Therefore if we find the time lag that results in the largest cross correlation between the two signals we can use it to synchronize the videos with other sensor data. To achieve this goal, first we use the front facing video feed to estimate vehicle's lateral movements using a dense optical flow algorithm based on Gunner Farneback's algorithm (Farnebäck, 2003). Then we use the resulting lateral movements signal and maximize its cross correlation with the steering wheel angle (SWA). This method produces satisfying results for the majority of the recordings. Having the videos and sensor data synchronized, next step is to annotate the videos. For the annotation we do not subjectively mark the segments that we believe driver is distracted, instead we mark beginning and the end of the segments that a secondary task (as mentioned above) is being performed. Our hypothesis is that there is a pattern vehicle sensor data which can be used to discriminate between the distracted and attentive driving. In this work we use data from 13 drivers, composed of 2 female and 11 male drivers.

4 METHODOLOGY

We formulate the problem as a supervised learning problem. In which the input data is a multi-variate time series recorded from a vehicle and the target is 0 for normal driving and 1 for distracted driving. We denote a driving trace as $(x^{(i)}, y^{(i)})_{i=1}^{N}$, where x is the sensor measurements from CAN-bus, $y \in \{0,1\}$ is the target label and N indicates the total number of measurements. We seek a function h that predicts $\hat{y}$ for short driving sequences. Since x is a multi-variate sequence it is not possible to apply classic machine learning algorithms, therefore we use the sliding window approach in order to apply conventional machine learning algorithms (Dietterich, 2002). We use *window classifier* h_w to map each window frame of length w into individual predictions. Let $d = (w-1)/2$ be the half-width length of window, for a window frame at time t, h_w makes prediction $\hat{y}_t$ based on window frame $\langle x_{t-d}, \cdots, x_t, \cdots, x_{t+d} \rangle$. To reduce the computational complexity we make window predictions for every $k = \lfloor w * (1 - \frac{r}{100}) \rfloor$ samples, where k denote *step size* and r indicates the percentage of overlap between two consecutive window frames. This results in $M = \frac{N}{k}$ window frames. Since it is possible that a window frame cover both distracted and non-distracted measurements, we label a window frame as distracted only if more than 50% of the measurements are distracted. Each driving trace $(x^{(i)}, y^{(i)})_{i=1}^{N}$ is converted into M window frames, then h_w is trained using feature vectors $\mathbf{x}$ computed for each window frame and its corresponding label. Similarly to classify an unseen driving trace x, it is first converted into window frames, then for each window frame at time t feature vector $\mathbf{x}_t$ is computed and h_w makes the prediction $\hat{y}_t$ based on $\mathbf{x}$. Lastly, we feed l consecutive window frame predictions $(\langle \hat{y}_{t-l}, \hat{y}_{t-l+1}, \cdots, \hat{y}_t \rangle)$ to the decision function f which produces the final prediction for the given sequence.

4.1 Evaluation Method

In order to utilize the entire driving traces $D = \{(x_i, y_i)\}_{i=1}^{|C|}$ for both training and testing, we employ a $|C|$ fold cross-validation method called leave-one-group-out. Each time we train a model using data from $|C|-1$ drivers and validate that on the data from the remaining driver. In other words, each driver is considered a group, therefore at each fold one driver is kept out of the training process, then h_w is trained and scored based on its prediction performance over the remaining slice. This is a subject independent model because model has no information about the driver that is being tested on. As mentioned in Section 3 we evaluate the proposed mechanism using data from 13 drivers therefore in this case $|C| = 13$.

Drivers typically drive attentively but sometimes they get distracted by engaging in secondary tasks. We see the same pattern in our dataset, in average only 36% of the measurements are labeled as distracted. Since the dataset is imbalanced in order to have a better measure of the proposed model's performance we choose to report F_1 score as well as ROC AUC as the main performance metrics. F_1 score is simply the harmonic mean of precision and recall:

$$F_1 = 2 \cdot \frac{1}{\frac{1}{recall} + \frac{1}{precision}} \tag{1}$$

Moreover since our classification problem is binary, based on the application needs we can modify the decision threshold. In order to avoid introducing new parameters and discuss how one may tune them, we employ the receiver operating characteristic (ROC) plot which demonstrates the diagnostic ability of a binary classifier as its discrimination threshold is being changed. In fact we use ROC AUC which is a simple way of reporting ROC plot using only one value (area under the ROC curve).

4.2 Feature Extraction

In order to remove high frequency noise, first we apply a low pass filter to all the signals to smoothen the

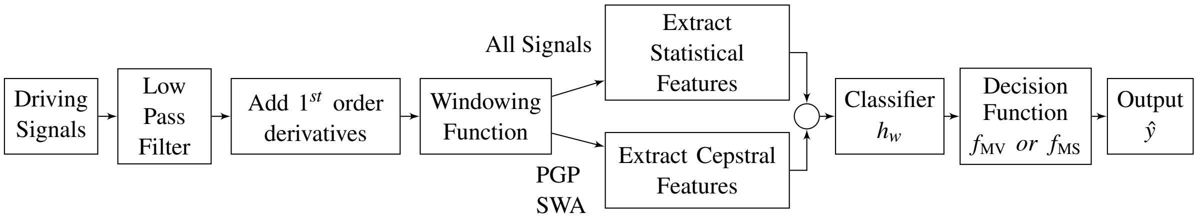

Figure 1: Distraction Detection Process

Table 2: Selected Signals for Driver Identification

Sensor	1st order Derivative	Cepstral
Percentage Gas Pedal (PGP)	Yes	Yes
Str. Wheel Angle (SWA)	No	Yes
Str. Wheel Rel. Speed (SWRS)	No	No
Vehicle Speed (VS)	Yes	No
Engine RPM (ERPM)	Yes	No
Pitch	No	No
Roll	No	No
Yaw	No	No

signals. Then for some of the signals (Shown in Table 2) we derive the temporal derivative of the signals. For both smoothing and computation of derivations we use an implementation of the Savitzky-Golay algorithm (Savitzky, 1964). Figure 1 shows the various stages that the signals go through before the feature extraction step. In this study we use 8 signals which are listed in Table 2. These signals are obtained from CAN-Bus or IMU. Then we apply a windowing function to all the signals which takes two parameters, w to determine the length of window frame and r to specify amount of overlap between two consecutive window frames. This will break down the signal into window frames of size w and is ready for feature extraction stage. Per each signal, 9 statistical features are extracted from each window frame. These descriptive statistics are selected to be representative of the distribution of sensor values covered by the window frame. This set includes minimum, maximum, mean, median, standard deviation, kurtosis, skewness and number of zero crossings. We also extract cepstral features from two signals, percentage of gas pedal (PGP) and SWA, these are the vehicle operating inputs that are directly operated by the driver. Earlier studies have proved that cepstral analysis are suitable for driver identification (Jafarnejad et al., 2017; Öztürk and Erzin, 2012), we suspect that they are also effective for detecting distracted driving. We compute the cepstral coefficients similar to Jafarnejad et al. and keep only the first 32 coefficients and use as cepstral features (Jafarnejad et al., 2017).

4.3 Feature Importance

Table 3: Average Scores For Each Signal

Signal	Mutual Info.	F-Test
SWRS	0.517	0.084
ERPM Derivative	0.510	0.059
Pitch	0.490	0.056
PGP Derivative	0.475	0.035
Roll	0.470	0.012
PGP	0.436	0.060
ERPM	0.428	0.359
VS Derivative	0.427	0.088
Yaw	0.425	0.054
VS	0.415	0.792
SWA	0.228	0.061
Cepstral Coefficients PGP	0.029	0.036
Cepstral Coefficients SWA	0.025	0.026

Table 4: Average Scores For Each Function

Function	Mutual Info.	F-Test
Min	0.955	0.310
Max	0.933	0.289
Range	0.495	0.076
Median	0.136	0.309
Mean	0.092	0.314
Standard Deviation	0.053	0.067
# Zero crossings	0.040	0.166
Cepstral Coefficients	0.034	0.039
Skewness	0.032	0.041
Kurtosis	0.030	0.081

We perform analysis on the extracted features to evaluate their fitness for our experiments. We use two metrics for this purpose, the conventional F-test and Mutual Information (MI) as two metrics to evaluate usefulness of our features as well as some insights into which features or signals are more important for our classification task. To get an idea that which signal is more important, first we compute the MI for all features, we normalize it and compute the average for 5 best features from each signal. The corresponding results are presented in Table 3. The signals are listed in descending order of their MI. We do the similar analysis for the statistical functions and cepstral coefficients, the corresponding results are presented in Table 4. Among the signals we can observe that features extracted from steering wheel relative speed (SWRS) have the highest MI score and the cepstral features are the worst. In terms of functions used for feature extraction we can see in Table 4 that very simple functions such as Min, Max, Range, Mean are more useful that Skewness and Kurtosis.

4.4 Classification Algorithms

We select five classification algorithms to be used as *window classifier* h_w. The following is the list of classifiers used in this study along with their corresponding parameters.

- **AdaBoost (AB)** - 200 decision trees as weak learners, learning rate = 0.75.
- **Gradient Boosting (GB)** 100 estimators, maximum depth = 6, maximum features = None, maximum depth = 6, learning rate = 0.05.
- **Random Forest (RF)** 200 estimators, maximum features = None, maximum depth = 7, class weight = balanced.
- **K-Nearest Neighbors (KNN)** # neighbors = 5.
- **Support Vector Machine (SVM)** RBF kernel, C = 0.1, $\gamma = 0.01$.

For our experiments we use the scikit-learn software package, all the other parameters are set to their default values as of scikit-learn version 0.19.2 (Pedregosa et al., 2011).

4.5 Decision Functions

The decision function f determines the final prediction $\hat{y}$ at time t based on the last l window predictions, we call this a decision window: $\langle \hat{y}_{t-l}, \cdots, \hat{y}_{t-1}, \hat{y}_t \rangle$, therefore l is the number of window frames covered by a decision window. Below we introduce two decision functions to obtain $\hat{y}$ and evaluate their performance later in Section 5:
1) Majority vote (MV) Let d to denote count of the window frames in the decision window that are predicted as distracted driving. Then a decision window is classified as distracted if $\frac{d}{l} > 0.5$.
2) Maximum score (MS) Let d' to be the cumulative classification score for distracted class. Then a decision window is classified as distracted if $\frac{d'}{l} > 0.5$.

5 MODEL PERFORMANCE

In this section we present the results of our experiments. In each subsection we focus on one of the components of our proposed methodology and discuss their implications.

5.1 Sliding Window Analysis

In order to find out the optimal sliding window size we try various window sizes[3] as well as overlap ratios[4]. Since we do not want to have our results influenced by the choice and working mechanism of the decision function, in these experiments we do not apply the decision function f and only consider the predictions from h_w. The results are presented in Figure 2, the reported numbers are the average cross validated scores for each combination of r and w parameters. It is evident that results improve as overlap r increases. This can be explained by the fact that an increase in overlap also increases the number of window frames and therefore having more examples for the classifier to learn from. It is also evident that performance improves as the window size w increases, since our features are mostly descriptive statistics, having larger window frames filters out noise from the features therefore larger windows yield better results. From this section we can conclude that for both parameters r and w, it is best to choose a larger value, however such choices have some drawbacks as well. Larger values for r lead to increase in number of examples and feature calculations, therefore becomes computationally more expensive. On the other hand larger w results on longer delays in the predictor, therefore w and r should be selected in a manner suitable for application needs.

5.2 Classifier Benchmarks

We run our experiments for all of the 5 selected classifiers and compare their performance as h_w, meaning we only score the classifiers for their ability to predict individual window frames. Results are presented in Figure 3 (obtained using $w = 30s$ and $r = 75\%$), when considering ROC AUC score GB yields the best performance and RF takes the second place.

We also investigate how different classifiers exhibit different behaviors due to the changes of window size w. Figure 4 depicts the average cross validated ROC AUC and F_1 scores for various window sizes and fixed overlap ratio of $r = 75\%$. For example

[3]Window sizes of 4, 7, 10, 15, 20, 30, 45 and 60
[4]Overlap ratios of 0, 25, 50, 75 and 90 percent.

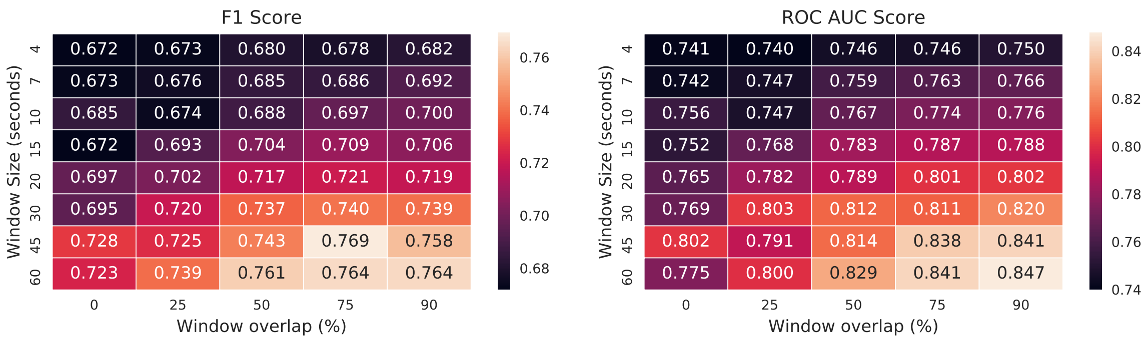

Figure 2: Effect of Window Size on Classification Score

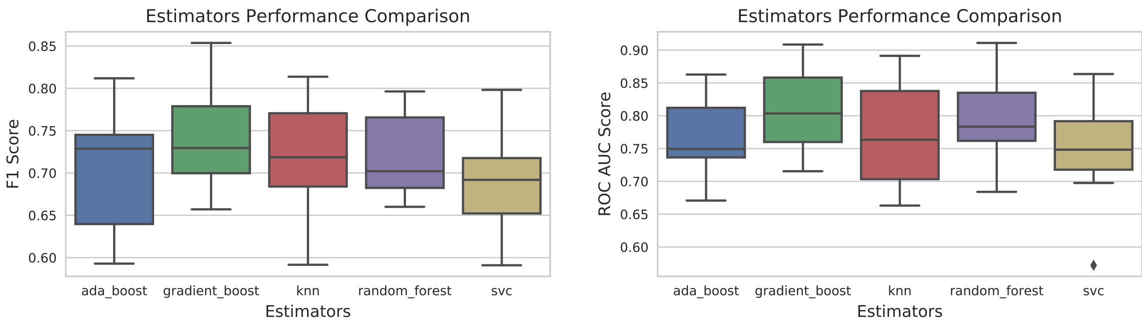

Figure 3: Estimators Comparison - $w = 30s$ and $r = 75\%$

KNN clearly benefits from a larger window frame, because as it was discussed larger window frames result in smoother features. On the contrary SVM does not improve as much as other classifiers do, this may improve by tuning the hyper-parameters for every window size however it is out of scope of this work.

5.3 Decision Function Benchmarks

Decision function f can be seen as a meta classifier which simply aggregates predictions from l consecutive window frames and outputs a single prediction. We investigate its performance by running experiments using values of 5, 10 and 15 for l and similar combinations for w as previous experiments. To compare the two proposed decision function a small sample of results (Cases with DW Duration $< 60s$) is presented in Table 5. The column DW Duration, in the table refers to the timespan covered by the decision function f to make the prediction, these values are computed considering overlap ratio of 75% between the consecutive window frames.
In order to demonstrate the advantage of using the decision function, we also present the scores for individual window frame predictions, to have a fair comparison for each row we compare the decision function with the window frame configurations that is closest. For example if $w = 4s$ and $DWLength = 15$, DW Duration will be 19 seconds, meaning the decision function effectively uses data from the past 19 seconds to make a prediction. We compare this instance with the predictions results for h_w (classifier without decision function) with the $w = 20s$, because it is the closest window frame size to 19s that we have considered. In the table we can see that in f_{MS} results in between 4 to 9.3 % (6.6% in average) improvements over not using a decision function. f_{MV} performs worse than f_{MS} with average improvement of 3.58%.

6 DISCUSSION

If we consider how the data-set is annotated and the signals we use for classification it is expected not to be able to achieve excellent results for detecting distractions. This is because we mark a long stretch of driving as distracted driving, however although in that period driver is engaged in a certain secondary task, this engagement is not uniformly present throughout the stretch, therefore we are inadvertently injecting noise into our training data. In fact one could perform the labeling more granularly and potentially improve the results. The proposed methodology is quite flexible and can be optimized for the intended applica-

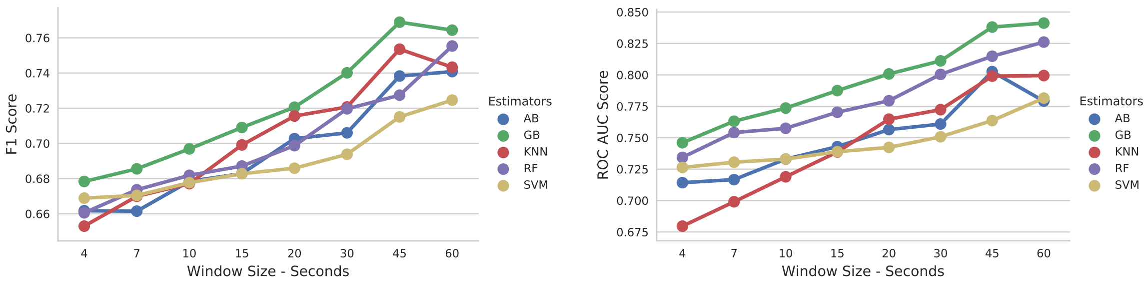

Figure 4: Estimators Performance Comparison - $r = 75\%$

Table 5: Decision Function Performance - Results are for KNN as classifier

Win. Len. (s)	Decision Window (DW) DW Len.	DW Duration (s)	Closest Window Len. (s)	ROC AUC	Decision Function ROC AUC MS	ROC AUC MV
4	5	9.00	10	0.718	0.748 (+4.07)	0.718 (-0.09)
4	10	14.00	15	0.738	0.780 (+5.61)	0.759 (+2.80)
4	15	19.00	20	0.764	0.799 (+4.60)	0.782 (+2.34)
7	5	15.75	15	0.738	0.774 (+4.84)	0.751 (+1.74)
7	10	24.50	20	0.764	0.812 (+6.28)	0.799 (+4.57)
7	15	33.25	30	0.772	0.836 (+8.29)	0.825 (+6.92)
10	5	22.50	20	0.764	0.801 (+4.83)	0.760 (-0.58)
10	10	35.00	30	0.772	0.844 (+9.38)	0.814 (+5.44)
10	15	47.50	45	0.798	0.871 (+9.05)	0.846 (+5.89)
15	5	33.75	30	0.772	0.822 (+6.48)	0.802 (+3.89)
15	10	52.50	45	0.798	0.867 (+8.59)	0.855 (+7.09)
20	5	45.00	45	0.798	0.855 (+7.12)	0.822 (+2.96)

tions, generally one needs to find the right balance between the quality of detection and delay in detection. For example using GB classifier, $w = 60$, $r = 0.75$ and $DW = 15$ (DW Duration of 285s) we achieve ROC AUC of 98.7%. If we need a shorter the detection time we get to ROC AUC of 92.7% with KNN classifier, $w = 20$, $r = 0.75$ and $DW = 15$ (DW Duration of 95s). For DW durations less than 60 seconds the results are presented on Table 5, for smaller DW durations, ROC AUC decreases. We should also point out that our evaluations are done not knowing anything about the individual, in other words the model is always trained on drivers that are not among the test set. Tango and Botta (Tango and Botta, 2013) indicate that they have gained about 20% improvement in the performance when they trained a model for each driver (Intra-subject). Another important factor is that we do not use any intrusive signals, not tracking of the driver's head nor the cabin sound, only sensor data from the car. Not only that, studies such as (Wöllmer et al., 2011) and (Tango and Botta, 2013) regardless of intrusive sensors, they do use metrics that are not readily available. Such as distance from the center of the lane. Such information are one of the key metrics influenced by the distracted driving. However here our goal is to make a similar prediction without having access to such information. With the current setup we may not be able to address applications that require accurate spontaneous and momentarily distraction detection. Instead the system performs well at detecting long lasting distractions, such as mobile-phone conversations or conversations among the passengers. Moreover one can use such a system as a way to characterize driver's riskiness. It is crucial for insurance, logistic and public transport companies, to keep track of their customers or employees risky driving behavior. Such application is equally beneficial for individuals who would like to keep track of their driving quality. For example parents who are concerned about safety of their teenagers, would like to know whether or not their children is a risky driver.

7 CONCLUSIONS

In this paper, we have proposed a mechanism to detect distracted driving based on non-intrusive vehicle sensor data. In the proposed method 8 driving signals are used. The data is collected, two types of statistical and cepstral features are extracted in a sliding window process, next a classifier makes a prediction for each window frame, and lastly, a decision function takes the last l predictions and makes the final prediction for the given window frames. We have evaluated the subject independent performance of the proposed mechanism using a driving data-set consisting of 13 drivers. We analyzed the implications of changing the size of sliding window and its overlap ratio. We have shown that the performance increases as the window size and decision window size become larger. We have compared the performance of several classifiers. The best results were achieved using GB classifier $w = 60$,$r = 0.75$ and $DW = 15$ (DW Duration of 285s) which yields ROC AUC of 98.7%. Our results show that even with poorly annotated data and only use of vehicle sensor data it is possible to accurately detect distracted driving events. In future work intra-subject models should be evaluated. It will be also of interest to see how the proposed mechanism performs on a more granular dataset, with more accurate labels.

ACKNOWLEDGEMENTS

The authors would like to thank Dr. Hüseyn Abut and VPALAB of Sabanci University for providing the data-set used for this study.

REFERENCES

Abut, H., Erdoğan, H., Erçil, A., Çürüklü, A. B., Koman, H. C., Tas, F., Argunşah, A. Ö., Akan, B., Karabalkan, H., Çökelek, E., et al. (2007). Data collection with" uyanik": too much pain; but gains are coming.

Berka, C., Levendowski, D. J., Lumicao, M. N., Yau, A., Davis, G., Zivkovic, V. T., Olmstead, R. E., Tremoulet, P. D., and Craven, P. L. (2007). EEG Correlates of Task Engagement and Mental Workload in Vigilance, Learning, and Memory Tasks.

Dietterich, T. G. (2002). Machine learning for sequential data: A review. In *Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR)*, pages 15–30. Springer.

Dong, Y., Hu, Z., Uchimura, K., and Murayama, N. (2011). Driver inattention monitoring system for intelligent vehicles: A review. *IEEE Transactions on Intelligent Transportation Systems*, 12(2):596–614.

Engström, J., Monk, C. A., Hanowski, R. J., Horrey, W. J., Lee, J. D., McGehee, D. V., Regan, M. a., Stevens, A., Traube, E., Tuukkanen, M., Victor, T., and Yang, D. (2013). A Conceptual Framework and Taxonomy for Understanding and Categorizing Driver Inattention.

Farnebäck, G. (2003). Two-Frame Motion Estimation Based on Polynomial Expansion. Springer.

Fridman, L., Brown, D. E., Angell, W., Abdić, I., Reimer, B., and Noh, H. Y. (2016). Automated synchronization of driving data using vibration and steering events. *Pattern Recogn. Lett.*, 75(C):9–15.

Harbluk, J. L., Noy, Y. I., Trbovich, P. L., and Eizenman, M. (2007). An on-road assessment of cognitive distraction: Impacts on drivers' visual behavior and braking performance. *Accident Analysis and Prevention*, 39(2):372–379.

Jafarnejad, S., Castignani, G., and Engel, T. (2017). Towards a real-time driver identification mechanism based on driving sensing data. In *20th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE.

Jin, L., Niu, Q., Hou, H., Xian, H., Wang, Y., and Shi, D. (2012). Driver cognitive distraction detection using driving performance measures. *Discrete Dynamics in Nature and Society*.

Klauer, S. G., Guo, F., Sudweeks, J., and Dingus, T. A. (2010). An analysis of driver inattention using a case-crossover approach on 100-car data. Technical report.

Lee, J., Young, K., and Regan, M. (2009). *Defining driver distraction*, pages 31 – 40. CRC Press, Australia.

Liang, Y. and Lee, J. D. (2010). Combining cognitive and visual distraction: Less than the sum of its parts. *Accident Analysis and Prevention*, 42(3):881–890.

Miyajima, C., Kusakawa, T., Nishino, T., Kitaoka, N., Itou, K., and Takeda, K. (2009). On-going data collection of driving behavior signals. In *In-Vehicle Corpus and Signal Processing for Driver Behavior*. Springer.

Öztürk, E. and Erzin, E. (2012). *Driver Status Identification from Driving Behavior Signals*, pages 31–55. Springer NY, New York, NY.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Rantanen, E. M. and Goldberg, J. H. (1999). The effect of mental workload on the visual field size and shape. *Ergonomics*, 42(6):816–834.

Savitzky, A. Golay, M. J. E. (1964). Smoothing and differentiation of data by simplified least squares procedures. *Anal. Chem.*, 36.

Tango, F. and Botta, M. (2013). Real-time detection system of driver distraction using machine learning. *IEEE Transactions on Intelligent Transportation Systems*, 14(2):894–905.

Wesley, A., Hoffman Hall, P. G., Shastri, D., and Pavlidis, I. (2010). A Novel Method to Monitor Driver's Distractions CHI 2010: Work-in-Progress.

Wöllmer, M., Blaschke, C., Schindl, T., Schuller, B., Färber, B., Mayer, S., and Trefflich, B. (2011). Online Driver Distraction Detection Using Long Short-Term Memory. *IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS*, 12(2).

Zhou, H., Itoh, M., and Inagaki, T. (2008). Influence of cognitively distracting activity on driver's eye movement during preparation of changing lanes. *Proceedings of the SICE Annual Conference*, pages 866–871.