Deep CNN, Body Pose and Body-Object Interaction Features for Drivers' Activity Monitoring

Ardhendu Behera, *Member, IEEE*, Zachary Wharton, Alexander Keidel and Bappaditya Debnath

***Abstract*—Automatic recognition and prediction of in-vehicle human activities has a significant impact on the next generation of driver assistance and intelligent autonomous vehicles. In this paper, we present a novel single image driver action recognition algorithm inspired by human perception that often focuses selectively on parts of the images to acquire information at specific places which are distinct to a given task. Unlike existing approaches, we argue that human activity is a combination of pose and semantic contextual cues. In detail, we model this by considering the configuration of body joints, their interaction with objects being represented as a pairwise relation to capture the structural information. Our body-pose and body-object interaction representation is built to be semantically rich and meaningful, and is highly discriminative even though it is coupled with a basic linear SVM classifier. We also propose a Multi-stream Deep Fusion Network (MDFN) for combining high-level semantics with CNN features. Our experimental results demonstrate that the proposed approach significantly improves the drivers' action recognition accuracy on two exacting datasets.**

***Index Terms*—Transfer learning, intelligent vehicles, in-vehicle activity monitoring, deep learning, body pose and contextual descriptor, neural network-based fusion.**

I. Introduction

There is a growing interest in the area of smart and connected control towards a fully Autonomous Vehicle (AV). It offers our desire for a better world in which injuries and fatalities from accidents are rare, congestion is lesser, and many societal and environmental benefits are far greater. It is suggested that this desire is unlikely to become a reality unless mindful attention is paid to human behaviour [1] since human error is overwhelmingly to blame for the vast majority of automobile accidents [2]. Today's automobile is nearly autonomous due to Advanced Driver Assistance System (ADAS) and is feasible due to ultrasonic sensors, cameras, radars and lidars. Such sensors mainly focus on surrounding environmental perception and minimal work has focused on human driver perspectives. The role of the driver could be taken over by automation, but the vehicle also requires to deliver performance identical to that of a driver if it is to be trusted. Therefore, the ADAS must focus on understanding, modelling and predicting human agents, as well as on the surrounding traffic conditions since a real-world driving scenario is a multi-agent system in which diverse participants interact with each other and with infrastructures. This will also contribute towards solving the complex problem of fully autonomous driving, including the assessment of traffic situations, reasoning nearby road-users' intentions, perception of the potential hazards, planning ego-trajectory, and finally executing the driving task.

A. Behera is with the Department of Computer Science, Edge Hill University, Lancashire, UK, e-mail: beheraa@edgehill.ac.uk

Z. Wharton, A. Kiedel and B. Debnath are with the Edge Hill University.

A complete understanding of driver's activities is a challenging problem. It is a key component of knowing how vehicles will learn to adapt to various driving conditions and environments. To address this, recent research on recognising basic driver's actions such as eating, drinking, interacting with the vehicle controls, and so on [3], [4], [5], [6], [7], [8], is only the first step. This study advances this by proposing a novel approach to enhance the performance of the automatic recognition of driver's activities from still images captured by vehicle cameras. There is significant progress in low-cost and low-power AI system such as NVIDIA Jetson (e.g. Nano, TX2 and AGX Xavier), which is targeted for AVs. This hardware advancement paired with the latest deep Convolutional Neural Network (CNN) makes it a reality to implement computer vision approaches for real-time monitoring of drivers' activity.

Driver behaviour recognition from images/videos is closely linked to vision-based human action/activity recognition, which has been extensively studied by the computer vision community over the past two decades. A complete survey of these methods is beyond the scope of this paper, and we refer the readers to recent survey papers [9]. Driver action recognition is often focused on vital cues such as head/body pose of the driver, and their interaction with objects in a given scene. Most of the recent approaches [6], [10], [3], [4], [11], [5] focus selectively on these vital cues to improve the recognition accuracy. In machine learning, this kind of processes is often referred to as the attention mechanism. Inspired by this, we model attention as a high-level semantic feature that combines body pose and body-objects interactions as pairwise relations to capture the structural information in discriminating various activities of a driver. All the body parts are not equally important in differentiating various actions. For example, drivers' activity types are often inferred from the configuration of upper-body parts. Moreover, many actions (e.g. eating, drinking, smoking, makeup, etc.) in images exhibit similar body part configuration and therefore, it is difficult to discriminate them. In such scenarios, contextual information (e.g. cues of body-objects interactions) plays a vital role [12], [13]. To improve the recognition accuracy, recently researchers have developed deep models focusing on spatio-temporal structures [14], [12], [15], [16], [17]. The main drawbacks in such approaches are: 1) mainly for solving video classification problems (complete observation) whereas our focus is on monitoring driver's on-going activity from partial observation so that the vehicle should be able to anticipate a distraction activity at the beginning. 2) These models are complex and computationally expensive, requiring estimation of a vast number of parameters and tuning of many hyper-parameters. This has a significant impact on the time taken to train such models, even when multiple GPUs are used.

To overcome these drawbacks, we take still image-based approach in which our novel contextual scene descriptor consists of high-level semantic information (i.e. spatial arrangements of body parts and objects in an image), benefiting from the available pre-trained body parts and objects detectors. The descriptor involves body poses, which encodes pairwise relations between various body parts (e.g. shoulder, elbow, wrists, etc.), as well as between objects of interest (e.g. mobile phones, bottles, etc.) and body parts. We justify that the proposed pose and human-objects interaction descriptor is simple yet rich and meaningful by exploring saliency around human pose keypoints and involved objects. The computation for generating our descriptor is simple and fast. Most of the execution time is consumed in inferencing the location of body joints and objects of interests in a given image. Moreover, we propose a novel lightweight model called Multi-stream Deep Fusion Network (MDFN) for combining transferable CNN features with the high-level semantic feature for the efficient recognition of the driver's state.

The article is organised as follows: Section II discusses related work on in-vehicle activity monitoring. Section III describes our aims and objectives. Section IV presents the proposed approach for recognising activities. Experimental results are discussed in section V, and the concluding remarks are given in Section VII.

II. Related Work

Human activity recognition research direction has made considerable headway in the computer vision community [9]. Whereas, in the automotive environment, it is still in its infancy. This could be due to the challenge faced by computer vision researchers to provide a powerful standard language that can adequately and concisely describe human actions.

Recently, deep learning has made a major advances in recognising driver activities from images/videos [18], [14], [6], [17]. Driver's activity recognition can be seen as a subset of the traditional human activity recognition problem. Therefore, these models are inherited from traditional human activities models consisting of highly distinctive human actions involving discriminative body poses, body-object, and/or human-human interactions. Moreover, these actions are often performed by different subjects. Whereas, driver's behaviour commonly involves different activities performed by the same subject (e.g. talking vs texting using a phone, eating vs drinking, etc.) resulting in subtle changes in the image. Deep models over the full image have shown great promise, but it raises the question of whether fine-grained driver's action recognition can be treated as a general classification problem. To address this, Leekha et al. [5] propose a CNN model to focus on foreground information consisting of driver's hand and faces to recognize various activities. Similarly, Huang et al. [18] present a hybrid CNN framework (HCF) to detect the behaviours of distracted drivers by using three deep CNNs and concatenated their outputs to recognise the behaviours. The framework is computationally expensive for real-time applications. To measure the distraction severity of a driver, Fasanmade et al. [10] introduce an expert knowledge-based rule system to predict the severity of distraction in a contiguous set of video frames features. The model performance is dependent on the accuracy of other modules such as face detection, its orientation, hand detection and previous drivers' activities. Moreover, these modules are required to run simultaneously to provide input features. As a result, the model is unsuitable for real-time applications. Likewise, Deo and Trivedi [4] suggest an LSTM-based deep model for continuous estimation of the drivers take-over readiness and is based on a holistic representation of the drivers state, gaze, hand, pose and foot activity. The approach is similar to [10] in the sense that it requires information from various modules (e.g. face detector, depth-based hand analysis, gaze analysis, etc.), which are required to run concurrently. Moreover, recurrent networks, such as LSTMs are known to be computationally expensive, resulting in their practicability in a resource-constrained environments such as robots and AVs. To improve the recognition accuracy, Kose et al. [16] propose a deep model to combine temporal and spatial information in videos. The model extracts features from sparsely selected frames using a BN-Inception network. Moslemi et al. [15] propose a method that uses the existing I3D deep model to combine optical flow and appearance features from videos using two-stream 3D-ConvNet. Martin et al. [17] advocate a method to combine multiple streams involving body pose and contextual information in videos to recognise driver's activities. Similarly, Behera et al. [13] describe a multi-stream LSTM for recognising driver's activities by combining high-level body pose and body-object interaction with CNN features. These models [16], [15], [17], [13] are similar to video classification methods, which require complete observation and is unsuitable for live activity recognition. Simialrly Alotaibi and Alotaibi [19] describe an approach that combines the inception module with a residual block and a hierarchical recurrent neural network to enhance the recognition performance of the distracted behaviours of drivers. To improve the image-based driver activity recognition, Xing et al. [11] describe an approach that applies segments using a GMM-based segmentation algorithm to identify the driver position and remove the irrelevant background information. The segmented image is used by standard CNN (e.g. AlexNet and ResNet) for activity recognition. Baheti et al. [7] propose a method to recognise driver's state by modifying VGG16 architecture to improve classification accuracy. The simplified VGG16 architecture is computationally efficient for real-time applications. Abouelnaga et al. [8] advocate a solution that considers the weighted ensemble of five different CNNs for high classification accuracy, but computationally expensive.

Our driver's activity recognition approach is based on the high-level semantic features (e.g. human pose and hand-object interactions). The novelty is that it combines CNN features, body pose and relationships between objects and body parts in innovative ways to recognise in-vehicle activities. The reason for using this high-level feature is to minimise the computational complexities by using simple classification algorithms such as linear SVM and the proposed lightweight MDFN, targeting real-world applications involving robotics and intelligent/autonomous vehicles.

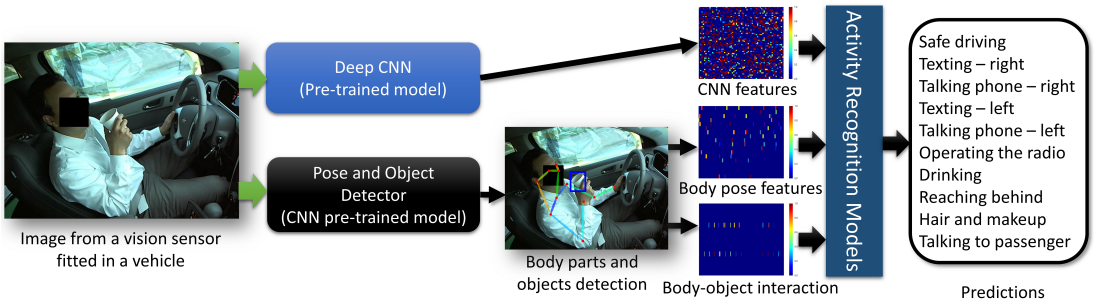
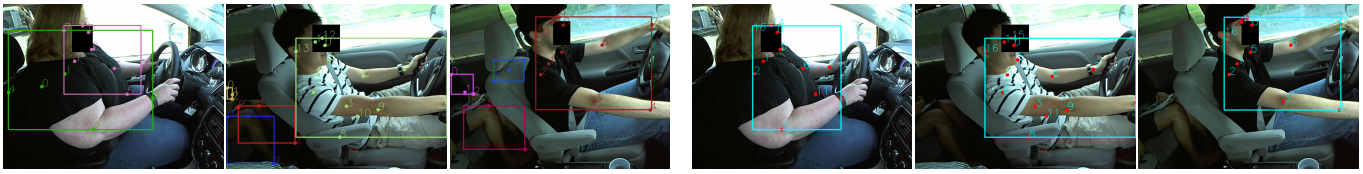


Fig. 1: The pipeline of the proposed approach: 1) an observed image passes through pre-trained deep CNN models to extract CNN features, detect various body joints and manipulated objects. 2) The proposed high-level semantic feature involving body pose and body-objects interaction (pairwise relations) is computed. 3) CNN features along with semantic feature are used by a classifier (e.g. SVM and our Multi-stream Deep Fusion Network) for recognising 10 different activities during driving.



(a) OpenPose's [20] output: single person's joints are detected as two people (left), passenger's body parts as three people (middle) and passenger's body parts and chair parts as three people (right)

(b) After pre-processing: removal of noisy joints detected as multiple people

Fig. 2: Pre-processing step of removing noisy multi-person detection and locating body joints of a driver.

III. Aims and Objectives of the Study

The overall aim is to recognise driver's activities. Within this broad theme, the research addresses the objectives below:

(i) to explore the high-level semantic feature involving body pose and body-object interactions and in particular, on their roles in discriminating various distraction activities;

(ii) to examine the various ways to combine the above semantic feature with generic feature descriptors such as CNN features, which are extracted using state-of-the-art CNN models, and its impact on recognising driver's state.

A series of experiments are carried out involving transferable CNN features, which are extracted from VGG16 [21], Inception-V3 [22] and Inception ResNet-V2 [23] deep models. These features are combined with the semantic information using our novel MDFN to recognise in-vehicle activities.

IV. Proposed Drivers' Activity Monitoring

The pipeline of our approach is shown in Fig 1. An observed image is processed to extract the transferable CNN features, and detect various body parts and objects of interest. Our novel semantic feature consisting of body pose and body-objects interactions are computed and used by the activity recognition model to recognise in-vehicle activities.

A. Transferable CNN Features

Given performance and wider usages, we use VGG16 [21], Inception-V3 [22] and Inception ResNet-V2 [23] models. We follow the finding in [24] to extract the CNN features just before the last layer of these models.

B. High-level Semantic Features to Represent Body Pose

Our goal is to model the configuration of body parts/joints as a high-level semantic feature for action recognition. Therefore, we detect and locate these parts in the given images. To achieve this, we use off-the-self body parts/joints detector using the state-of-the-art AlphaPose [25], which can detect the key joints of multi-person in real-time.

1) Pre-processing of the detected key joints: There are noises in the detected joints due to unavoidable partial occlusions and/or lighting conditions resulting from driving circumstances and environmental situations. These noises are: 1) detected joints of the co-passengers, 2) roadside pedestrian (visible via vehicle's windows), 3) joints of a single person are split into multiple people, 4) missing joints and 5) false detection. The aim is to recognise activity from the noisy outputs instead of improving the detection accuracy by re-training on a target dataset. We apply the following simple logic on the detected joints to minimise these noises.

(i) A Bounding Box (BB) containing detected joints per person is computed. If more than one BB is detected, then we look for the overlap, which is computed as an Intersection over Union, $IoU = \frac{\text{Area of Overlap}}{\text{Area of Union}}$.

(ii) If there is significant overlap among two or more BBs ($IoU \geq 5\%$) then we look into whether the detected joints are divided among multiple people, i.e. multiple BB (Fig. 2a: left). This is carried out by simply looking at the joints correspondence. Let's say we have two overlapping BB (B_1 and B_2). The proposed approach looks for correspondence of every joint in the overlapped BB. There are three possibilities: 1) joint presents in both B_1 and B_2, 2) it is absent in both or 3) it is present in one of them. In the first scenario, we simply consider the joint that has a higher detection score. In the second case, it is marked as absent, and in the third, we simply consider the joint. This process will continue until all the

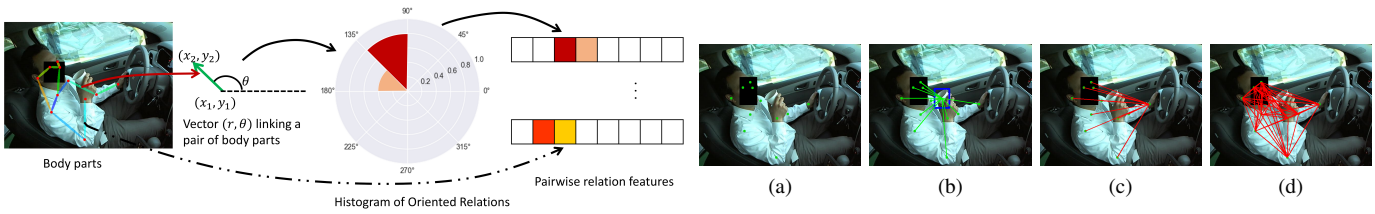


Fig. 3: Features representing human body pose, i.e. body parts configuration (left). a) Detected body joints, b) pairwise relations between object (cup) and body joints representing human-object interaction, c) pairwise relations between the joint "left wrist" and the rest, and d) all possible pairwise relations representing body pose.

detected joints are visited.

(iii) When there is no/minimal overlap, then we consider the BB with the largest area and highest number of detected joints as the target person (e.g. Fig. 2a: middle and right) since the majority of the image is covered by the driver.

(iv) It is observed that one or more smaller BBs often appear within a large BB. In such cases, the smaller BBs are typically roadside pedestrians, which are visible through vehicle windows and appeared in images. We simply consider the larger BB (i.e. driver) since the driver is clearly visible and occupies a large image area.

The above-mentioned pre-processing step might not completely be free from noise. However, our pairwise relation feature is based on the spatial configuration of the detected joints and therefore, it has the ability to handle such noises. For example, if a wrist joint is noisy (false detection) or undetected, then the relationships between other detected joints (e.g. neck, shoulder, elbow, etc.) would capture the body pose.

2) Pairwise body joints feature: We use a novel pairwise relation feature encoding the configuration of a pair of joints. If we have N joints, then there are $N(N-1)/2$ numbers of possible pairs. For each pair, we compute a relational feature vector f by considering the joints' x, y positions in image-plane. Given a pair of joints j_1 and j_2, and their respective positions (x_1, y_1) and (x_2, y_2), their relation is represented using distance r and orientation θ:

$$r = \sqrt{(y_2 - y_1)^2 + (x_2 - x_1)^2}, \theta = tan^{-1}\left(\frac{y_2 - y_1}{x_2 - x_1}\right) \quad (1)$$

The angle $\theta = [-\pi, \pi]$ is mapped into $[0, 2\pi]$ by applying the modulo operator. It is then binned into h number of bins and the magnitude r contributes to the respective bin(s) where the θ falls into. The length of the feature f is the bin size (i.e. h). The extraction procedure for an 8-bin ($h = 8$) feature is shown in Fig. 3. The relational feature vector f between a pair of joints is computed only if both joints are detected. Otherwise, f is assigned to zero. The process continues for all possible $(N(N-1)/2)$ pairs and concatenates the extracted pairwise feature into a single feature vector $F = [f_1, f_2, \cdots, f_{N(N-1)/2}]$ of length $N(N-1)/2 \times h$.

C. Semantic Features Representing Body-Objects Interactions

To model body-objects interactions, we need to detect the targeted objects commonly used or interacted with (e.g. mobile phone, cup, etc.) during driving. In this work, we use the Faster R-CNN with Inception ResNet-V2 detector [26] and is one of the best so far. It is also faster than its descendants (R-CNN and Fast R-CNN) and is considered based on is overall performance (computational complexity and recognition accuracy) for real-time application.

1) Filtering out unwanted detected objects: Our objects of interest (e.g. phone, cup, makeup brush, etc.) are very small, and are considered based on their size and aspect ratio with respect to the driver's bounding box. The filtering could be done simply by considering the object types (e.g. water bottle, cup, etc.). However, we have noticed that there is often an incorrect assignment of labels to objects. For example, the label of cellphone, coffee cup and remote are often exchanged. A makeup brush is often detected as a toothbrush. We are interested in visual cues (configuration of objects with respect to joints) involving human-object interactions to recognise non-driving secondary activities. Thus, we argue that if an object is wrongly labelled, then the combined configuration of body parts and objects would provide enough cues for discriminating various non-driving activities. For example, if a cellphone is labelled as a cup, based on the arm configuration, its position with respect to other body parts (e.g. torso, shoulder, etc.) and the location of the detected object with respect to body parts, would provide information in discriminating texting against talking or drinking.

2) Pairwise joints-objects feature: The histogram of oriented relation feature $\hat{f}$ representing the relationship between body joints and objects is extracted in a similar way as it is in pose feature f (Fig. 3). For N body joints and O objects, there are $N \times O$ possible pairs. The feature $\hat{f}$ is computed for each pair and stacked into a single feature vector $\hat{F} = [\hat{f}_1, \hat{f}_2, \cdots, \hat{f}_{N \times O}]$ of size $N \times O \times h$ (h orientation bins).

D. Drivers' Activity/State Recognition

We use three different experiments: 1) linear Support Vector Machine (SVM) as a classifier that takes CNN and semantic features. 2) Feature-level, classifier-level and Deep Neural Network (DNN) fusion strategies to combine the CNN and semantic features. 3) Fine-tune the existing state-of-the-art deep CNN models on the target dataset.

1) Linear SVM-based recognition: For the SVM-based recognition, we use a linear SVM (LIBLINEAR) [27] to solve:

$$\underset{w}{\text{minimize}} \frac{1}{2}\|w^2\| + C\sum_i \max(1 - y_i w^T X_i, 0)^2 \quad (2)$$

Where (X_i, y_i) represents feature-label pair of i^{th} image and C is a penalty parameter. We use the well-known probability

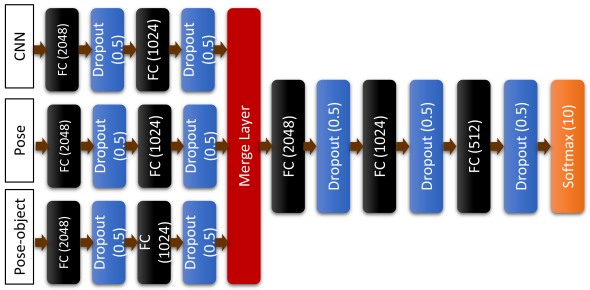


Fig. 4: Proposed Multi-stream Deep Fusion Network (MDFN) for in-vehicle activity recognition.

TABLE I: Performance of linear SVM with various features (CNN, body pose and body-objects interactions). The **bold** font represents the best performance for a given combination and evaluation type.

SVM-based Method	**Trained on Set A**			**Trained on Set B**		
(Section IV-D1)	ACC	LRAP	Loss	ACC	LRAP	Loss
VGG16 [21] (V_1)	57.70	71.85	2.920	71.29	82.13	1.127
Incep-ResNet-V2 [23] (V_2)	68.55	79.64	1.787	79.31	87.32	0.755
Inception-V3 [22] (V_3)	65.98	78.25	1.783	75.97	85.08	0.896
Body pose	86.62	91.73	0.524	87.27	92.29	0.479
Body objects	58.50	70.16	1.322	66.34	76.44	1.049
Body pose + Body objects	**89.15**	**93.18**	**0.453**	**89.66**	**93.73**	**0.393**
V_1 + Pose + Objects	81.91	88.81	0.709	89.67	93.79	0.397
V_2 + Pose + Objects	**89.74**	**93.68**	**0.424**	**90.96**	**94.59**	0.354
V_3 + Pose + Objects	89.28	93.40	0.450	90.88	94.56	**0.347**

calibration method [28], which transforms linear SVM predictions to posterior probabilities (Platt calibration).

2) Recognition by combining various features: We use different combinations of features via: a) feature-level fusion (concatenating various features), and b) classifier-level fusion. For classifier-level fusion, we use the below two strategies:

i) Fusion using SVM classifier's output - In this case, the final decision is a combined output from multiple linear SVMs. Let's say we train linear SVM S_1 for CNN, S_2 for pose and S_3 for the joints-object feature. The goal is to infer the activity class label L from above three SVMs i.e. $P(L|S_1, S_2, S_3)$. By applying Bayes' theorem:

$$P(L|S_1, S_2, S_3) = \frac{P(S_1, S_2, S_3|L)P(L)}{P(S_1, S_2, S_3)} \tag{3}$$

S_1 S_2 and S_3 are independently trained and therefore:

$$P(L|S_1, S_2, S_3) = \frac{P(S_1|L)P(S_2|L)P(S_3|L)P(L)}{P(S_1)P(S_2)P(S_3)} \tag{4}$$

Applying the Bayes' theorem again on the right-hand side:

$$\begin{aligned} &= \frac{\frac{P(L|S_1)P(S_1)}{P(L)}\frac{P(L|S_2)P(S_2)}{P(L)}\frac{P(L|S_3)P(S_3)}{P(L)}P(L)}{P(S_1)(PS_2)P(S_3)} \\ &= \frac{P(L|S_1)P(L|S_2)P(L|S_3)}{P(L)P(L)} \\ &\simeq P(L|S_1)P(L|S_2)P(L|S_3) \end{aligned} \tag{5}$$

Where $P(L)$ is the prior probability of individual action class and is constant. $P(L|S_1)$, $P(L|S_2)$ and $P(L|S_3)$ the action class probability from S_1, S_2 and S_3, respectively.

ii) Fusion using deep neural network - we propose a Multi-stream Deep Fusion Network (MDFN) consisting of Fully-Connected (FC), dropout and softmax layers as shown in Fig. 4. The model takes three different input features and is flexible to add more input streams. The network is lightweight and could be trained using the CPU. The depth of the network, the number of layers, the number of nodes in each layer and the dropout rate are experimentally determined based on the best performance (Fig. 4).

V. Experiments

In our experiment, we use "State Farm" [29] and "Distracted Driver" [8] datasets which are the first to consider a wide variety of distractions and are publicly accessible. These datasets consist of inward-facing dashboard camera images depicting ten activities: 1) safe driving, 2) texting - right, 3) talking on the phone - right, 4) texting - left, 5) talking on the phone - left, 6) operating the radio, 7) drinking, 8) reaching behind, 9) hair and makeup, and 10) talking to a passenger.

A. State Farm (SF) Dataset

It is used in Kaggle competition and consists of two sets: i) A - training set (22,424 images) and ii) B - testing set (79,726 images). The train (A) and test data (B) are split among the drivers, such that one driver can only appear on either train or test set. The class labels of the training images are available and not for the test images. We manually labelled all the test images. In our experiments, we train our models on images in A and validate images in B and vice versa. The aim is also to evaluate the effect of dataset size on performance since the set B is significantly larger than the A.

B. Distracted Drivers (DD) Dataset

This dataset [8] is similar to the State Farm [29] and consists of 12,977 training and 4,331 testing images from 31 drivers. Train and test images are not split among drivers.

C. Evaluation Criteria

The evaluation metrics of accuracy (ACC) and multi-class log loss are used. The log loss quantifies the accuracy of a classifier by penalising confident false classifications. An ideal classifier would have a zero log loss.

Our models produce a list of possible responses to a queried image, ordered by the probability of correctness. Thus, we also consider the Label Ranking Average Precision (LRAP) [30]. This metric is linked to average precision but is based on the notion of label ranking instead of precision and recall. Given a binary indicator matrix of the target labels $y \in R^{N\times 10}$ and the predicated probabilities $p \in R^{N\times 10}$, it is defined as:

$$LRAP(y,p) = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{10}\sum_{j:y_{i,j}=1}\frac{|\,\gamma_{i,j}\,|}{rank_{i,j}} \tag{6}$$

where $\gamma_{i,j} = \{k : y_{i,k} = 1, p_{i,k} \geq p_{i,j}\}$, $rank_{i,j} = |\,\{k : p_{i,k} \geq p_{i,j}\}\,|$ and $|\cdot|$ is the L_0-norm.

TABLE II: Performance of feature-level and classifier-level fusion using linear SVM, as well as fusion using the deep neural network (Fig. 4) with various combination. $V_1 \rightarrow$ VGG16 [21], $V_2 \rightarrow$ Inception ResNet-V2 [23] and $V_3 \rightarrow$ Inception-V3 [22]. The **bold** font represents the best performance for a given combination and evaluation type.

Various Fusion	**Trained on Set A**			**Trained on Set B**		
(Section IV-D2)	ACC	LRAP	Loss	ACC	LRAP	Loss
	Fusion using feature concatenation (SVM)					
$V_1 + V_3$	65.33	77.42	2.186	79.99	87.69	0.793
$V_1 + V_2$	63.04	75.68	2.450	77.09	85.98	0.876
$V_2 + V_3$	**72.09**	**82.12**	**1.674**	**81.89**	**88.86**	**0.695**
$V_1 + V_2 + V_3$	67.95	79.15	2.021	81.52	88.77	0.715
	Pose and Objects with CNN features					
$V_1 + V_3$ + Pose + Objects	83.28	89.69	0.673	90.12	94.08	0.379
$V_1 + V_2$ + Pose + Objects	83.05	89.54	0.677	89.87	93.92	0.393
$V_2 + V_3$ + Pose + Objects	**89.54**	**93.60**	**0.436**	**91.19**	**94.73**	**0.340**
$V_1 + V_2 + V_3$+Pose+Objects	83.29	89.71	0.671	90.22	94.17	0.372
	Fusion using SVM classifier's output					
$V_1 + V_3$	70.22	81.19	2.749	81.33	88.79	0.978
$V_1 + V_2$	70.90	81.55	2.745	82.45	89.52	0.924
$V_2 + V_3$	74.80	84.26	**2.123**	83.21	89.89	**0.832**
$V_1 + V_2 + V_3$	**76.23**	**85.17**	2.853	**84.75**	**91.02**	1.058
	Pose and Objects with CNN features					
$V_1 + V_3$ + Pose + Objects	80.87	88.16	2.008	89.09	93.65	0.815
$V_1 + V_2$ + Pose + Objects	81.27	88.38	2.003	89.57	93.89	0.730
$V_2 + V_3$ + Pose + Objects	**84.19**	**90.25**	**1.584**	**89.71**	**94.04**	**0.674**
$V_1 + V_2 + V_3$+Pose+Objects	82.65	89.30	2.271	89.63	93.99	0.879
	Fusion using MDFN (Fig. 4)					
V_1 + Pose + Objects	88.00	92.47	0.533	91.26	94.81	**0.326**
V_2 + Pose + Objects	88.88	93.08	0.492	91.34	94.77	0.345
V_3 + Pose + Objects	**89.13**	**93.25**	**0.478**	**91.39**	**94.87**	0.425

D. Various Feature Extraction and Model Parameters

For readability, we use the notation of V_1 for VGG16 [21], V_2 for Inception ResNet-V2 [23] and V_3 for Inception-V3 [22] in the rest of the paper. For all our experiments, unless stated otherwise, we use the last layer before the softmax layer to extract CNN features. We use the pre-trained models' default image size: 224×224 for V_1, and 299×299 for V_2 and V_3. Our pose descriptor uses $N = 16$ body joints (120 pairwise relations) since driver's feet are occluded by the dashboard (section IV-B). A total of 25 objects of interest is selected (section IV-C1), resulting in 16×25 joints-objects pairwise relations (section IV-C2). We re-emphasise that the body joints and objects detectors are not fine-tuned on the target datasets.

The number of bins ($h = 6, 9, 12, 18$) for the pose feature is selected experimentally and found that $h = 12$ with L_2-norm performed better than the rest. The final feature-length is 120×12 (body pose) and $16 \times 25 \times 12$ (pose-objects interaction). For linear SVM, the parameter C is decided through the cross-validation strategy. In State Farm, we use subject-wise cross-validation (leave 2-subjects out) in the training set A and 5-fold cross-validation in the testing set B. We have also used 5-fold cross-validation for the distracted driver dataset. For fine-tuning the deep models, we use RMSProp optimiser [31] to minimise the categorical cross-entropy $E_i = -\sum_{j=1}^{10} y_{i,j} log(p_{i,j})$, where $p_{i,j}$ and $y_{i,j}$ are the respective prediction and target for i^{th} image belonging to j^{th} class. The learning rate is set to 0.001.

TABLE III: Evaluation of the state-of-the-art deep models on the State Farm's train-set (i.e. set A). * represents the evaluation is carried out using a random split of the set A and does not represent the actual dataset size and split criteria (cross driver split, which is difficult than the random split).

Model	ACC	LRAP	Log loss
NASNet mobile [32]	84.46	89.20	1.262
DenseNet169 [33]	86.74	91.57	0.994
Inception ResNet-V2 [23]	87.65	92.15	0.871
Inception-V3 [22]	89.30	93.00	0.741
ResNet + HRNN* (video) [19]	99.30	-	-
I3D two-stream* (video) [15]	94.40	-	-
HCF* [18]	96.74	-	-
GrabCut + ConvNet* [5]	98.48	-	-
V_2 + Pose + Objects (Ours-SVM)	**89.74**	**93.68**	**0.424**

The MDFN is evaluated using three input streams: one for CNN, other for pose and the third one for pose-object interaction feature. The optimal batch size is decided experimentally and is presented in the supplementary figure.

VI. Results and Discussion

There is a significant impact of our semantic features (objective i & ii) on recognising driver's state. It is evident in the performances in both the SF [29] (Table I & Table II) and DA [8] (Table IV) datasets.

A. Performance on State Farm (SF) dataset

The performance of body pose alone (Table I: row 4) is far better than the respective CNN features (V_1, V_2 and V_3). However, when these CNN features are combined with the proposed semantic feature representing body-object relations, it gives the best performance (Table I: rows 7-9). This shows the impact of our feature in recognising drivers' state. It is observed that the performance improved significantly (ACC: 20-25% on A and 12-15% on B). The performance of CNN feature using model V_2 and V_3 (as well as combined with semantic information) is better than the V_1. However, it catches up with V_2 and V_3 on larger set B (V_1:89.67%, V_2:90.96% and V_3:90.88%) in comparison to set A (V_1:81.91%, V_2:89.74% and V_3:89.28%). This implies that CNN features using model V_2 and V_3 are more appropriate for a smaller dataset.

The accuracy using V_1, V_2 and V_3 trained on B (larger set) is more than 10% in comparison to the A (Table I). However, this improvement using our semantic feature is less than 1%. This shows our feature is semantically rich and meaningful and does not depend much on the size of a dataset.

1) Performance of various fusion: The performance of various fusion strategies is presented in Table II. It is clear that the accuracy using the proposed classifier (linear SVMs) fusion is $3 - 10\%$ and $0.5 - 5\%$ better than the feature fusion (Section IV-D2) for the set A and B, respectively. The only exception is the classifier-level fusion using our semantic feature. Nevertheless, the difference is less than 2%. This suggests about the importance of our high-level semantic features that can be efficiently used to discriminate activities using simple linear SVM. The main takeaway message from

TABLE IV: Performance of various combinations using "Distracted Drivers" dataset [8]. The **bold** font represents the best performance for a given combination and evaluation type.

Method	ACC	LRAP	Log Loss
VGG16 [21] (V_1)	**91.71**	**95.26**	0.939
ResNet-V2 [23] (V_2)	75.11	85.29	1.269
Inception-V3 [22] (V_3)	78.11	87.05	1.266
Body pose	80.74	88.19	0.876
Body objects	53.82	68.03	1.565
Pose + Objects	83.63	90.09	**0.741**
V_1 + Pose + Objects	**92.27**	**95.54**	0.913
V_2 + Pose + Objects	87.64	92.60	**0.735**
V_3 + Pose + Objects	88.71	93.38	0.749
	Fusion - feature concatenation		
$V_1 + V_2 + V_3$	92.27	95.54	0.913
$V_1 + V_2 + V_3$ + Pose + Objects	92.27	95.54	0.914
	Fusion - SVM classifier's output		
$V_1 + V_2 + V_3$	89.03	93.63	0.434
$V_1 + V_2 + V_3$ + Pose + Objects	90.05	94.21	0.379
	Multi-stream Fusion (MDFN, Fig. 4)		
V_1 + Pose + Objects	94.74	96.77	0.399
$V_1 + V_2 + V_3$	**95.57**	**97.31**	**0.396**
	State-of-the-art approaches		
Abouelnaga et al. [8] ensemble	**95.98**	–	**0.158**
Abouelnaga et al. [8] real-time	94.29	–	0.273
Baheti et al. [7]	95.54	–	–
ResNet + HRNN (video) [19]	92.36	–	–
I3D two-stream (video) [15]	73.00	–	–
GrabCut + ConvNet [5]	95.64	–	–

various fusion approaches are: 1) using feature-level fusion, V_2 and V_3 are more appropriate and 2) for the SVM-level fusion, V_1, V_2 and V_3 perform better than their feature-level fusion.

The performance of our MDFN is better than the rest of the fusion in Table II, except the combination V_2+V_3+pose+objects on A. We believe that this exception is due to the smaller training size and thus, the proposed SVM approach is preferred to the MDFN for a small dataset. The confusion matrices are given in the supplementary document.

2) Comparison with the existing approaches: We use *transfer learning* to re-train the state-of-the-art models using State Farm's [29] train-set (i.e. set A) including a lightweight NASNet mobile model [32], which is more suitable for a resource-constrained environment. The performance is shown in Table III. The Inception-V3 [22] (89.30%) outperforms the rest. The proposed feature-level fusion using the CNN feature and our novel semantic feature is better (89.74%) than the Inception-V3. Moreover, the feature-level fusion uses simple SVM and could execute in real-time. We have also fine-tuned the VGG16 [21], Inception-V3 [22] and Inception ResNet-V2 [23] models on the target dataset. The results and discussions are included in the supplementary document (Table VII & VIII). We have submitted our results to the Kaggle and appeared on the top 17% in the leaderboard. The winner[1] used empirical tricks such as focusing head and driver's right hand regions for fine-grained representations resulting in higher recognition accuracy. We would like to emphasise that our approach using the proposed high-level feature is simple yet effective and semantically rich. A linear SVM gives a very good performance.

Recently, researchers [19], [15], [18], [5] use State Farm's set A to evaluate their approach due to the unavailability of ground-truth labels for set B (i.e. test set). The results are marked as *. It is expected that the performance will be better as their experiments are carried out on set A using a random split, which does not represent the actual dataset size and split criteria (cross driver split, which is difficult than the random split). Moreover, many of these approaches [19], [15] use video-based analysis (optical flow and recurrent network), which are computationally expensive and require complete observation to recognize drivers states and thus, are unappropriate for the real-time application involving recognition from partial observation. Furthermore, our approach outperforms these approaches when the same train-test split is used in the Distracted Drivers dataset [8] (following discussion).

B. Performance on Distracted Drivers (DD) dataset

The performance of our approach and various state-of-the-art methods is presented in Table IV. In our approach, the CNN feature using VGG16 [21] are extracted from the block5 pooling layer performs better than the rest (rows 1-6). However, when the semantic features are added, the accuracy has improved to 92.27%. We have also experimented with various fusion using feature concatenation and classifier-level fusion.

The proposed MDFN (Fig. 4) outperforms (95.57%) the rest using only CNN features. It is better than the state-of-the-art baselines (AlexNet: 93.65%, Inception-V3: 95.17%, real-time: 94.29%) as well as the video-based approaches in [19], [15]. It is also competitive in comparison to the weighted ensemble [8], and GrabCut-ConvNet in [5]. These methods use multiple models (e.g. weighted ensemble consists of 5 AlexNet and 5 Inception-V3 networks) and thus computationally heavy. The proposed MDFN is lightweight and is suitable for real-time applications (for computational time see section VI-C).

C. Model Inference Time

Our approach is evaluated on a standard Windows 10 PC (Intel i7-6700 CPU, 3.40GHz) fitted with low-end 8GB GPU (NVIDIA M4000). Our method is implemented using TensorFlow and Keras. Below are the average execution time per-frame in seconds using our PC: 1) CNN features (VGG16: 0.024, Inception V3: 0.047 and ResNet V2: 0.099), object detection [26]: 1.817, semantic features: 0.013, linear SVM: 5.04×10^{-4}, MDFN: 4.95×10^{-4}. When we use MDFN for three CNN features the per-frame execution time is $0.024 + 0.047 + 0.099 + 5 \times 10^{-4} = 0.171$, which is 5.86 fps and suitable for real-time applications.

VII. Conclusion

In this paper, we have presented a novel approach for driver activity recognition from still images. The approach uses a

[1]https://www.kaggle.com/c/state-farm-distracted-driver-detection/discussion/22906

semantically rich and meaningful descriptor by exploring the configuration of body parts, as well as the interaction between body parts and objects. We have found that the descriptor is highly discriminative in recognising various activities. We have also proposed a novel Multi-stream Deep Fusion Network (MDFN) and classifier-level fusion for combining the CNN features with the proposed descriptor. We have shown experimentally that the performance of MFDN is superior to the classifier-level fusion, which is better than the feature-level fusion, using two challenging datasets. The source code[2] is available. In future work, we plan to extend it to video-based activity monitoring by including spatio-temporal attention.

ACKNOWLEDGMENT

This research is supported by Research Investment Fund (RIF) at the Edge Hill University (EHU) and the UKIERI-DST grant CHARM (DST UKIERI-2018-19-10). The authors would like to thank Taylor Smith at the State Farm for the dataset. We thank Erik Thomas at EHU for his contribution to data annotation.

REFERENCES

[1] D. L. Fisher, M. Lohrenz, D. Moore, E. D. Nadler, and J. K. Pollard, "Humans and intelligent vehicles: The hope, the help, and the harm," *IEEE Trans. on Intel. Vehicles*, vol. 1, no. 1, pp. 56–67, March 2016.

[2] B. D. Seppelt, S. Seaman, J. Lee, L. S. Angell, B. Mehler, and B. Reimer, "Glass half-full: On-road glance metrics differentiate crashes from near-crashes in the 100-car data," *Accident Analysis & Prevention*, vol. 107, pp. 48 – 62, 2017.

[3] A. El Khatib, C. Ou, and F. Karray, "Driver inattention detection in the context of next-generation autonomous vehicles design: A survey," *IEEE Trans. on Intelligent Transp. Sys.*, pp. 1–14, 2019.

[4] N. Deo and M. M. Trivedi, "Looking at the driver/rider in autonomous vehicles to predict take-over readiness," *IEEE Trans. on Intelligent Vehicles*, vol. 5, no. 1, pp. 41–52, 2019.

[5] M. Leekha, M. Goswami, R. R. Shah, Y. Yin, and R. Zimmermann, "Are you paying attention? detecting distracted driving in real-time," in *IEEE Fifth Int'l Conf. on Multimedia Big Data (BigMM)*, 2019, pp. 171–180.

[6] M. H. Alkinani, W. Z. Khan, and Q. Arshad, "Detecting human driver inattentive and aggressive driving behavior using deep learning: Recent advances, requirements and open challenges," *IEEE Access*, vol. 8, pp. 105 008–105 030, 2020.

[7] B. Baheti, S. Gajre, and S. Talbar, "Detection of distracted driver using convolutional neural network," in *IEEE Computer Vision and Pattern Recognition (CVPR) Workshop*, 2018.

[8] Y. Abouelnaga, H. M. Eraqi, and M. N. Moustafa, "Real-time distracted driver posture classification," *arXiv preprint arXiv:1706.09498*, 2017.

[9] S. Herath, M. Harandi, and F. Porikli, "Going deeper into action recognition: A survey," *Image and Vision Computing*, vol. 60, pp. 4 – 21, 2017.

[10] A. Fasanmade, Y. He, A. H. Al-Bayatti, J. N. Morden, S. O. Aliyu, A. S. Alfakeeh, and A. O. Alsayed, "A fuzzy-logic approach to dynamic bayesian severity level classification of driver distraction using image recognition," *IEEE Access*, vol. 8, pp. 95 197–95 207, 2020.

[11] Y. Xing, C. Lv, H. Wang, D. Cao, E. Velenis, and F.-Y. Wang, "Driver activity recognition for intelligent vehicles: A deep learning approach," *IEEE Trans. on Vehicular Tech.*, vol. 68, no. 6, pp. 5379–5390, 2019.

[12] P. Weyers, D. Schiebener, and A. Kummert, "Action and object interaction recognition for driver activity classification," in *IEEE Intelligent Transp. Sys. Conf. (ITSC)*, 2019, pp. 4336–4341.

[13] A. Behera, A. Keidel, and B. Debnath, "Context-driven multi-stream lstm (m-lstm) for recognizing fine-grained activity of drivers," in *German Conference on Pattern Recognition*, 2018, pp. 298–314.

[14] M. Shahverdy, M. Fathy, R. Berangi, and M. Sabokrou, "Driver behavior detection and classification using deep convolutional neural networks," *Expert Systems with Applications*, vol. 149, p. 113240, 2020.

[15] N. Moslemi, R. Azmi, and M. Soryani, "Driver distraction recognition using 3d convolutional neural networks," in *Int'l Conf. on Pattern Recog. and Image Analysis (IPRIA)*, 2019, pp. 145–151.

[16] N. Kose, O. Kopuklu, A. Unnervik, and G. Rigoll, "Real-time driver state monitoring using a cnn based spatio-temporal approach," in *IEEE Intelligent Transp. Sys. Conf. (ITSC)*, 2019, pp. 3236–3242.

[17] M. Martin, A. Roitberg, M. Haurilet, M. Horne, S. Reiß, M. Voit, and R. Stiefelhagen, "Drive&act: A multi-modal dataset for fine-grained driver behavior recognition in autonomous vehicles," in *Proc. of the IEEE Int'l Conf. on Computer Vision*, 2019, pp. 2801–2810.

[18] C. Huang, X. Wang, J. Cao, S. Wang, and Y. Zhang, "Hcf: A hybrid cnn framework for behavior detection of distracted drivers," *IEEE Access*, vol. 8, pp. 109 335–109 349, 2020.

[19] M. Alotaibi and B. Alotaibi, "Distracted driver classification using deep learning," *Signal, Image and Video Processing*, pp. 1–8, 2019.

[20] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2d pose estimation using part affinity fields," in *IEEE Computer Vision and Pattern Recognition (CVPR)*, 2017.

[21] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *CoRR*, vol. abs/1409.1556, 2014.

[22] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *IEEE Computer Vision and Pattern Recognition (CVPR)*, 2016.

[23] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," in *AAAI Conf. on Artificial Intelligence*, 2017, pp. 4278–4284.

[24] A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, "Cnn features off-the-shelf: An astounding baseline for recognition," in *IEEE Computer Vision and Pattern Recognition Workshop (CVPRW)*, 2014, pp. 512–519.

[25] H.-S. Fang, S. Xie, Y.-W. Tai, and C. Lu, "RMPE: Regional multi-person pose estimation," in *Int'l Conf. on Computer Vision (ICCV)*, 2017.

[26] J. Huang, V. Rathod, C. Sun, M. Zhu, A. Korattikara, A. Fathi, I. Fischer, Z. Wojna, Y. Song, S. Guadarrama, and K. Murphy, "Speed/accuracy trade-offs for modern convolutional object detectors," in *IEEE Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 3296–3297.

[27] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, "Liblinear: A library for large linear classification," *J. Mach. Learn. Res.*, vol. 9, pp. 1871–1874, Jun. 2008.

[28] A. Niculescu-Mizil and R. Caruana, "Predicting good probabilities with supervised learning," in *Proc. of the 22nd Int'l Conf. on Machine Learning (ICML)*, 2005, pp. 625–632.

[29] S. F. Corporate, "State farm distracted driver detection." [Online]. Available: https://www.kaggle.com/c/state-farm-distracted-driver-detection

[30] G. Tsoumakas, I. Katakis, and I. Vlahavas, *Mining Multi-label Data*. Boston, MA: Springer US, 2010, pp. 667–685.

[31] T. Tieleman and G. CHinton, "Coursera: Neural networks for machine learning," http://www.cs.toronto.edu/~tijmen/csc321/slides/lecture_slides_lec6.pdf, 2012.

[32] B. Zoph, V. Vasudevan, J. Shlens, and Q. V. Le, "Learning transferable architectures for scalable image recognition," in *IEEE Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 8697–8710.

[33] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *IEEE Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 4700–4708.

[2] https://github.com/ArdhenduBehera/DistractedDriver/

Ardhendu Behera received the PhD degree in computer science from the University of Fribourg. He is currently a Reader in the Department of Computer Science, Edge Hill University. He is a Fellow of HEA and member of IEEE, BMVA, AVA, affiliated member of IAPR and ECAI. His main interests include computer vision, human-robot social interaction, activity analysis and recognition.

Zachary Wharton is an MRes student in the Department of Computer Science, Edge Hill University. His interests include computer vision, human-robot interaction (HRI) and pattern recognition.

Alexander Keidel is a research assistant in the Department of Computer Science, Edge Hill University. He received his MComp in computing from the Edge Hill University. His interests include deep learning, data visualisation, computer vision, machine learning and pattern recognition.

Bapaditya Debnath is a PhD student in the Department of Computer Science, Edge Hill University. He received his MSc in computer science from the Loughborough University. He is interested in deep learning, pose estimation, action and activity recognition.