IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received August 17, 2021, accepted August 26, 2021, date of publication September 1, 2021, date of current version September 17, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3109815

TML: A Triple-Wise Multi-Task Learning Framework for Distracted Driver Recognition

DICHAO LIU[1], (Member, IEEE), TOSHIHIKO YAMASAKI[2], (Member, IEEE), YU WANG[3], (Member, IEEE), KENJI MASE[1], (Senior Member, IEEE), AND JIEN KATO[3], (Senior Member, IEEE)

[1]Graduate School of Informatics, Nagoya University, Nagoya 464-8601, Japan
[2]Graduate School of Information Science and Technology, The University of Tokyo, Tokyo 113-8656, Japan
[3]College of Information Science and Engineering, Ritsumeikan University, Kusatsu-shi, Shiga 525-8577, Japan

Corresponding author: Dichao Liu (liu_di_chao@yahoo.co.jp)

This work was supported in part by the Ph.D. Professional Toryumon Program of Nagoya University.

This work involved human subjects or animals in its research. The authors confirm that all human/animal subject research procedures and protocols are exempt from review board approval.

ABSTRACT We propose a multi-task learning framework for improving the performance of vision-based deep-learning approaches for driver distraction recognition. The most popular tool so far for solving this task is convolutional neural networks (CNNs) that have proven to be strongly biased toward local features. Such bias causes CNNs to neglect global structural information, adversely affecting the robustness of the distracted driver recognition task. To solve this problem, we generate positive and negative samples of each given input, and construct a triplet of images (i.e., raw image, positive sample, and negative sample). The positive sample is generated by applying structure-aware illumination to the human body region of each given input. The negative sample is generated by randomly shuffling the local regions of each given input. The networks are then trained with the triplets using a multi-task learning strategy to force the networks to explore global information by multiple tasks: (a) recognizing the raw input and positive sample as the given ground truth; (b) recognizing the negative sample as an extra ''meaningless'' label; (c) pulling closer the distance between the features obtained from the raw input and positive sample while pushing away the distance between the features obtained from the raw input and negative sample. By doing so, the model can be trained so that it neglects the background information and pays more attention to the global structual information of the scene. The proposed approach reaches state-of-the-art performance on the AUC Distracted Driver Dataset and performs better than state-of-the-art studies on the Drive and Act Dataset. With raw images as input, we have achieved an accuracy of 96.0% for the AUC distracted driver dataset and 66.8% for the Drive and Act Dataset. Our approach does not introduce extra overhead during the testing procedure (i.e., utilization procedure), which is helpful for real-life applications. Moreover, better accuracy can be achieved by fusing the predictions respectively obtained with the raw input and positive sample. As a result, we have achieved an accuracy of 96.3% for the AUC distracted driver Dataset and 66.9% for the Drive and Act Dataset. The class activation map (CAM) of our proposed method is subjectively more reasonable, which would enhance the reliability and explainability of the model.

INDEX TERMS Action recognition, advanced driver assistance, contrastive learning, multi-task learning, intelligent vehicles.

I. INTRODUCTION

Nowadays, distracted driving has become a huge threat to human society. According to the report issued by the National Highway Traffic Safety Administration (NHTSA) in the United State in 2019, traffic accidents caused by distracted driving led to 3,142 or 8.7 percent of all accidents of this year in the United States [1], and most of them were involved in texting or talking on mobile phones. Owing to this situation, a reduction in traffic accidents can be realized if we can

The associate editor coordinating the review of this manuscript and approving it for publication was Razi Iqbal.

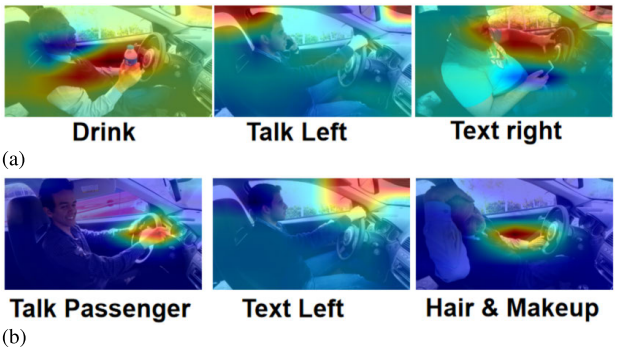


FIGURE 1. Examples of images from the AUC Distracted Driver Dataset [13] that are wrongly classified by Efficientnet-b3-pruned [14] trained on this dataset. The network's focus is visualized using class activation maps (CAMs) [15], which are saliency maps denoting the region the CNN uses as clues to identify the predicted category. (a) shows some examples that the CNN makes no distinction between the major and the minor clues. (b) shows some examples that the CNN model only focuses on a certain small local region while neglecting other crucial clues.

develop distracted driving detectors. Such detectors can be used in cars to alert the driver when distracted driving is sensed [2]–[7].

According to NHTSA, distracted driving is defined as ''any activity that diverts attention from driving.'' Examples of such actions include eating, drinking, talking to passengers, etc. [8], [9]. A more specific explanation of distracted driving is given by the Centers for Disease Control and Prevention (CDC), which defines three situations of distracted driving: visual distraction (i.e., looking around rather than visually focusing on the road), cognitive distraction (i.e., looking at the road but not mentally focusing on it), and manual distraction (i.e., the driver taking his/her hands off the steering wheel) [10]. In this paper, we focus on computer-vision-based approaches for recognizing different distracted driving behaviors based on dashcam videos. Firstly, vision-based approaches always act as the base of many driver assistance systems. Some driver assistance systems also use other sensors together with a dashcam, but the information captured by the dashcam is still very important [11], [12]. Secondly, vision-based solutions are often lower cost. Dashcam videos are often much cheaper than many other sensors, such as LIDAR and NIR cameras.

This work was developed on two public datasets, namely, the AUC Distracted Driver Dataset [13] and the Drive and Act Dataset [16], both of which consider the manual categories of distractions. Following the success of CNNs in various computer vision fields, there has been increasing interest in developing CNN-based approaches to obtain a better recognition of human action in different situations [10], [17], [18]. However, recent studies [19], [20] have proven that CNNs tend to be strongly biased toward local features instead of global information. Consequently, it is difficult for CNNs to utilize global information, such as the spatial relations of local features, as clues for driver behavior recognition. As shown in Figure 1, specifically in a distracted driving recognition task, a CNN tends to only focus on a certain local region or puts its focus everywhere because it does not properly ''understand'' the spatial relations of different local regions. In some wrongly classified cases (e.g., Figure 1(a)), the attention of the network is too scattered. Consequently, when predicting the category, the network cannot determine which regions are important. In some other wrongly classified cases (e.g., Figure 1(b)), the CNN model only focuses on a certain small local region, which decreases the robustness of the network. For example, focusing only on whether both of the driver's hands are placed on the steering wheel can sometimes predict the correct action. However, the network may make a mistake if it does not explore more information, such as the driver's pose. We assume that these problems are caused by the local bias of CNNs because of which the network lacks awareness of the semantically important global structure. And this degrades the reliability of the detection systems.

The above-mentioned problems result in the following research question addressed in this paper: Can forcing CNNs to explore more global information improve the accuracy for distracted driver recognition?

To answer the research questions, in this paper, we propose a triple-wise multi-task learning (TML) framework, which forces the model to reduce the bias toward local features. Firstly, for each given input, the TML generates a positive sample by applying structure-aware illumination to the human body region, and it generates a negative sample by randomly shuffling the local regions. Clearly, compared with the raw input image, the positive sample keeps the same global structure as the raw input, while the local texture of the most crucial regions (i.e., human) is smoothed. The negative sample has the same small local regions while the spatial relationships of those small local regions are destructed. Secondly, the TML forces the model to explore global information by a multi-task learning strategy that requires the model to find the difference between the input and negative samples, as well as the commonalities between the input and positive samples. The multiple tasks that we use to train the model are summarized as follows:

- Classification: to recognize the original input and positive sample as the given ground truth, and to recognize the negative sample as an extra ''meaningless'' category (Figure 2(b)).
- Contrastive learning: to push away the distance between the deep features learned from the original input and negative sample, and to pull closer the distance between the deep features learned from the original input and positive sample (Figure 2(c)).

Our contributions are summarized as follows:

- We solve the research question by proposing a novel multi-task learning framework that forces the networks to strengthen the awareness of the global information, such as the spatial relations of different local regions.
- Our approach is easy to implement with different network backbones.

- With different backbones, the proposed framework outperforms baselines by 0.7%–1.2% on the AUC Distracted Driver Dataset [13], and 1.8%–3.1% on the Drive and Act Dataset [16]. Our best result is 96.3% on the AUC Distracted Driver Dataset [13], and 66.9% on the Drive and Act Dataset [16].
- We are proposing a leaning scheme, not a new deep learning architecture. Therefore, our method can generally boost the recognition accuracy as demonstrated in Section V.l

The rest part of this paper is organized as follows. Section II introduces some prior studies that are related to our work. Section III describes the proposed framework in details. Section IV introduces the experiments for evaluating the effectiveness of the proposed framework, and Section V summarizes the experimental results. Finally, Section VI presents our conclusions.

II. RELATED WORKS

In this section, we introduce the previous studies that are related to this work. Subsection II-A introduces the previous studies on the topic of vision-based distracted driving recognition. Subsection II-B introduces the previous studies on revealing the local bias of convolutional neural networks.

A. DISTRACTED DRIVING RECOGNITION

Earlier, many types of research have been done on vision-based distracted driver classification from the video recorded by the dashcam of vehicles. In recent years, researchers have proposed various method to explore significant visual information for recognizing distracted driving from image and video data. Such visual information includes the eye gaze [21]–[23], head pose [24]–[26], fatigue cues extracted from the face [27]–[29], and body pose [30], [31].

Recently, with the significant progress in the development of deep learning models, especially CNNs in the computer vision field, a common approach has been to use deep learning models to solve distracted driving tasks [10], [17], [18]. For example, Yan *et al.* [18] embeded local neighborhood operations and trainable feature selectors within a deep CNN, and by doing so, meaningful features could be selected automatically to recognize distracted driving.

Abouelnaga *et al.* [10] proposed a technique to combine multiple streams of CNN model. These streams were constructed with different backbones (i.e., AlexNet [32] and Inception-v3 [33]) and were trained with different types of input (i.e., raw images, face-segmented images, hands-segmented images, skin-segmented images, etc.). The prediction logits of multi-stream CNNs were applied with a weighted sum to compute the final prediction score.

Arefin *et al.* [34] proposed to combine a modification of AlexNet [32] model with the aggregation of HOG features and brought improvement on classification accuracy.

Hu *et al.* [35] used multi-scale CNN blocks with different kernel sizes to generate hierarchical feature maps and then fused multi-scale information for distracted driver recognition.

Behera *et al.* [36] explored the configuration of body parts, as well as the interaction between body parts and objects. They also proposed a multi-stream deep fusion network to combine image features, pose features, and pose-object interaction features.

Qin *et al.* [37] proposed a light weight detector by decreasing the filter size. The number of parameters was as small as 0.76 million.

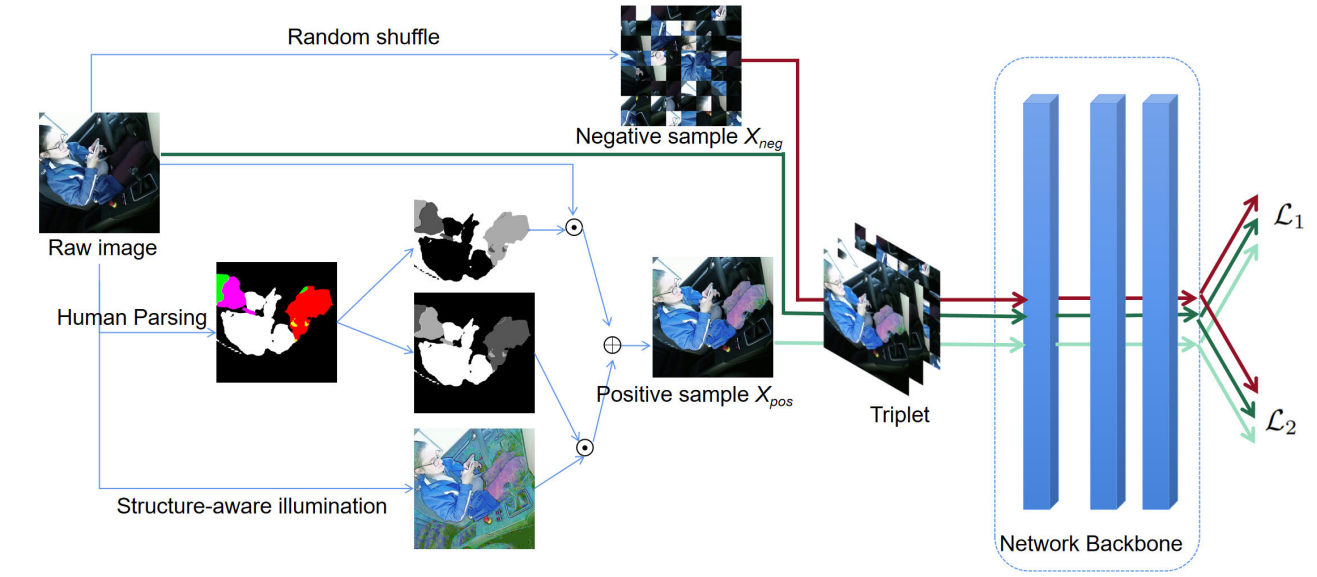
The above-mentioned studies yielded improvement in classification accuracy. However, the common weakness of them is that they require sophisticated strategies to obtain complementary information, such as multi-region or multi-scale information. It is because the local bias makes it hard for CNNs to capture all-sided information with a single original image. Multi-region and multi-scale images are needed for capturing information in all aspects. Instead of fusing multiple information obtained from multiple inputs, we use a triplet of different inputs to force the CNN models to improve global awareness. The benefit is that our framework does not require to increase the overhead of assembling multiple CNN streams to fuse the information obtained with different types of input.

B. LOCAL BIAS OF CONVOLUTIONAL NEURAL NETWORKS

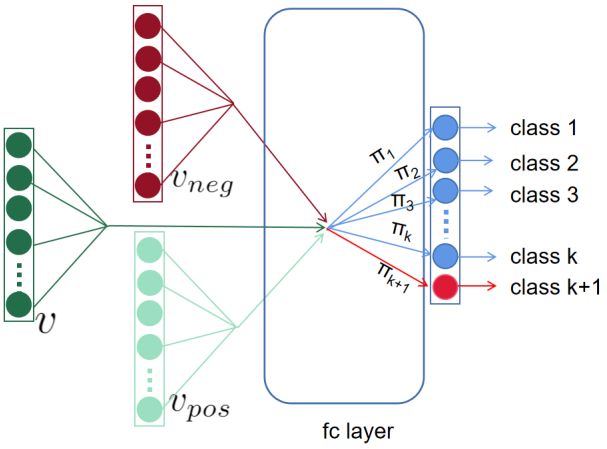
While realizing significant developments in a wide range of computer vision tasks, the internal mechanism of CNNs still remains to be a ''black box'' to human as we do not know exactly how the CNNs solve the given problem. However, some recent studies have moved a step forward in understanding the mechanisms of CNNs, and most of them report that CNNs are strongly biased toward local features [19], [20], [38].

Geirhos *et al.* [19] used a well-designed experiment to analyze whether CNNs were more receptive to local or global features and provided evidence that CNNs are biased towards local features. The authors conducted a careful psychophysical experiment to analyze how humans and CNNs act differently in terms of shape and texture cues. They created pictures with a texture-shape cue conflict and allowed human or CNNs to distinguish the pictures. For example, they generated an image of a cat shape with an elephant texture and let humans or CNNs distinguish whether the image was a cat or an elephant. After 48,560 psychophysical trials across 97 observers, they found that CNNs had a very strong texture bias, while humans tended to recognize the category according to their shape.

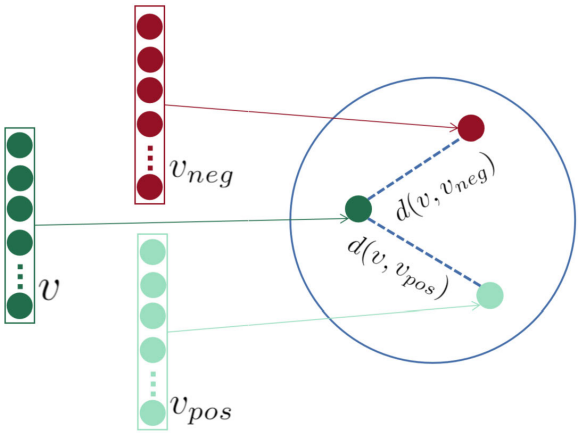
Brendel and Bethge [38] proposed a model named BagNet, and trained BagNet with the features of small cropped local regions without using any information about the global spatial structure. BagNet showed similar performance on ImageNet [39] when compared to AlexNet, which meant that even though it was given full images, the CNN found no more clues than the BagNet, which only saw small local images. should



(a) Illustration of the process of TML



(b) Illustration of $\mathcal{L}_1$



(c) Illustration of $\mathcal{L}_2$

FIGURE 2. (a) illustrates the process of the proposed framework. (b) illustrates $\mathcal{L}_1$, which is based on the softmax loss. (c) illustrates $\mathcal{L}_2$, which is based on triplet loss. More details are provided in Section III.

Such a bias toward local features does adversely affect the robustness of CNNs [19], [20]. Some previous studies, such as [10], stack multi-stream networks to learn information from different perspectives. Such studies, although they are not directly intended to reduce local bias, have some effect on the capture of global information. However, a multi-stream strategy significantly increases the number of model parameters and the consumption of computational resources. In contrast, our work reinforces the model's awareness of global spatial information with raw-positive-negative triplets, which introduces no extra parameter numbers to the backbone network.

III. PROPOSED FRAMEWORK

In this section, we introduce the details of the proposed framework. Firstly, TML generates triplets composed of a raw image, a positive sample, and a negative sample. The positive sample maintains the same global spatial structure as the raw input but smoothens the local texture of the human body region. The negative sample is generated by keeping the same local information as the raw input but destroying the global spatial structure. Thereafter, TML reduces the CNN's local bias by exploring information among the triplets with a multi-task learning strategy.

The rest part of this section is organized as follows. Subsection III-A introduces the key definitions and notations of this section. Subsection III-B describes how we generate the negative and positive samples in details. Subsection III-C introduces the multi-task learning strategy we use to train the framework.

A. DEFINITION AND NOTATION

Let X, X_{pos} and X_{neg} respectively be the raw input, positive sample and negative sample. v, v_{pos}, and v_{neg} respectively denote the deep features learned from X, X_{pos} and X_{neg}.

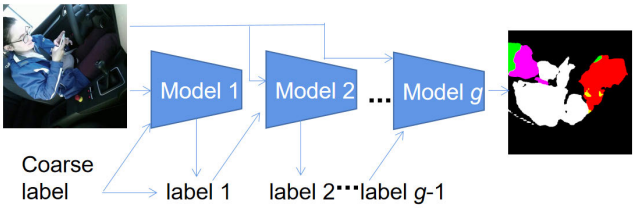


FIGURE 3. Simplified illustration of a self-correction human parsing framework that has g progressive models. Here, we mainly illustrate the parts that are closely related to our work.

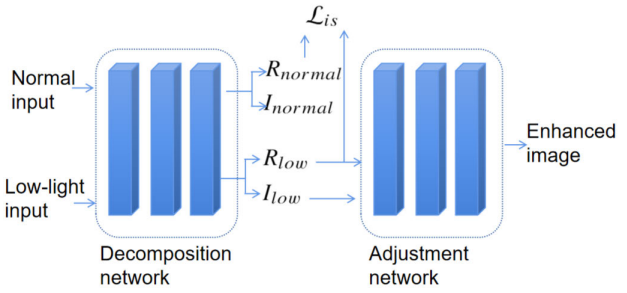


FIGURE 4. Simplified illustration of the architecture of Retinex-Net. Here, we mainly illustrate the parts that are closely related to our work.

To generate the positive samples, we need a mask denoted as X_{mask} to segment the human body region of each image. X_{mask} is computed from human parsing result of X, which is defined as X_{hp}.

The number of channels is neglected, and the two-dimensional (2D) size of X, X_{mask}, and X_{hp} is $H \times W$ (height, width). In this paper, we regard X, X_{mask}, and X_{hp} as sets of pixels and $X = \{x_{(1,1)}, x_{(1,2)}, \ldots, x_{(\alpha,\beta)}, \ldots, x_{(H,W)}\}$, $X_{mask} = \{x^m_{(1,1)}, x^m_{(1,2)}, \ldots, x^m_{(\alpha,\beta)}, \ldots, x^m_{(H,W)}\}$, $X_{hp} = \{x^{hp}_{(1,1)}, x^{hp}_{(1,2)}, \ldots, x^{hp}_{(\alpha,\beta)}, \ldots, x^{hp}_{(H,W)}\}$.

To generate the negative samples, we need to divide X into small sub-regions, and $N \times N$ denotes the numbers of the sub-regions. $\mathbb{R}$ and $\mathbb{R}'$ respectively denote the sub-region in X and X_{neg}.

$l_1, \ldots, l_k$ denote the logits outputted by the final fully-connected (fc) layer. $\Pi = \{\pi_1, \ldots, \pi_k\}$ denotes the parameters of the final fc layer, and each of $\pi_1, \ldots, \pi_k$ corresponds to each of $l_1, \ldots, l_k$.

$\mathcal{L}_1$ and $\mathcal{L}_2$ denote the loss functions we use for our multi-task learning.

B. GENERATION OF POSITIVE AND NEGATIVE SAMPLES

As shown in Figure 2(a), given an input, the proposed framework generates a positive sample X_{pos} by applying structure-aware illumination to the human body region that is masked out by human parsing and a negative sample X_{neg} by randomly shuffling local regions. Both of them are designed to force the neural networks to be less biased toward local information and to explore more global structure information.

1) POSITIVE SAMPLE GENERATION

The positive sample is generated by changing the illumination conditions of the human body regions. First, we compute the human pose mask X_{mask} of a single channel. The human pose mask X_{mask} is defined as:

$$x^m_{(\alpha,\beta)} = \frac{x^{hp}_{(\alpha,\beta)} - \min(X_{hp})}{\max(X_{hp}) - \min(X_{hp})}, \tag{1}$$

where $X_{hp} = \{x^{hp}_{(1,1)}, x^{hp}_{(1,2)}, \ldots, x^{hp}_{(\alpha,\beta)}, \ldots, x^{hp}_{(H,W)}\}$ is defined as:

$$X_{hp} = SCHP(X). \tag{2}$$

Here, *SCHP* denotes the self-correction human parsing [40], which is a state-of-the-art human parsing strategy. In human parsing tasks, pixel-level semantic regions must be manually annotated. However, owing to the difficulty associated with annotating pixel-level labels, the annotators tend to be confused by the unclear edges between different semantic regions and introduce noise to the labels. To solve this problem, *SCHP* is proposed to improve the reliability of pixel-level labels with a progressive schedule. As shown in Figure 3, *SCHP* has a progressive schedule with G learning cycles. In the first cycle, the model is trained using coarse manual labels. In each of the other cycles, the model takes the training data as the input and uses a pixel-wise mask output from the former cycle as the label. By doing so, the model and labels are progressively improved to be more robust and accurate. Let θ_g be the model weights learned at the end of cycle g ($g \in \{1, 2, 3, \ldots, G\}$). The model weights are updated as follows:

$$\theta_g = \frac{g}{g+1}\theta_{g-1} + \frac{1}{g+1}\theta_g. \tag{3}$$

In this work, we train the *SCHP* mdoel on the Pascal-Person-Part dataset [41] and then obtain *SCHP*(X).

Then, the positive sample is generated as follows:

$$X_{pos} = RET(X) \odot X_{mask} + X \odot (1 - X_{mask}), \tag{4}$$

where $\odot$ denotes the element-wise product, and *RET* denotes a pretrained Retinex-Net [42], which we use to generate the structure-aware illumination of X. Structure-aware illumination can preserve the overall structure boundary while smoothing the local texture [42], [43]; Retinex-Net was initially designed for low-light image enhancement. As shown in Figure 4, Retinex-Net is composed of a decomposition network and an adjustment network. Given low-right and normal-light image pairs, the decomposition network first decomposes each image into their reflectance and illumination. Thereafter, the adjustment network takes the reflectance and illumination of the low-light image as the input and generates the target image. During training, the framework is forced to improve the structure awareness using the structure-aware smoothness loss $\mathcal{L}_{is}$. Let R_{low} and I_{low} be the reflectance and illumination of the low-light image, respectively. Let R_{normal} and I_{normal} be the reflectance and

illumination of the normal-light image, respectively. $\mathcal{L}_{is}$ is defined as:

$$\mathcal{L}_{is} = \sum_{j=normal,low} \|\nabla I_j \odot \exp(-C\nabla R_j)\|, \tag{5}$$

where ∇ denotes the gradient, and C denotes the coefficient that balances the structure-awareness strength (C is manually set to 10 in [42], and we follow the same setting). $\mathcal{L}_{is}$ considers the locations that have steep reflectance gradients as global structures and relaxes the constraint. Otherwise, the locations are regarded as local textures, and the constraint is strengthened. In this work, we first train Retinex-Net on the LOL dataset [42], and then treat X as the low-light input to obtain $RET(X)$.

Clearly, X and X_{pos} share the same background, and the only difference between them lies in the human body regions. Thus, X and X_{pos} can be regarded as different views of the same action. When neural networks are required to find the commonalities between X and X_{pos}, they must explore the global structure of the human body regions. This is because: (a) the background regions of X and X_{pos} are already the same, and will output the same deep features (e.g., features in the penultimate layer) when passing through neural networks. That is, the background regions do not contribute to the loss that we designed to pull close the deep features of X and X_{pos} (details in subsection III-C). This loss can only be reduced by exploring the human body part. (b) In X_{pos}, the local texture is to some extent smoothed with RET, and the clues have to be explored in a more global style.

2) NEGATIVE SAMPLE GENERATION

The negative sample is designed to break the spatial structure of the image while simultaneously maintaining the same local information. Given the input X, we first divide X into $N \times N$ sub-regions denoted by $\mathbb{R}_{i,j}$, where $i \in \{1, 2, 3, \ldots, N\}$ and $j \in \{1, 2, 3, \ldots, N\}$ are the horizontal and vertical indices, respectively. Assuming that X_{neg} is similarly divided into $N \times N$ sub-regions $\mathbb{R}'_{i,j}$, we then obtain the negative sample X_{neg} by randomly shuffling the sub-regions and ensuring that $\mathbb{R}'_{i,j} \neq \mathbb{R}_{i,j}$. By doing so, the spatial relationship of objects is destroyed, and X_{neg} should not correspond to the same class as X.

Note that although the global spatial structure is destroyed, the local information and global statistical information remain the same between X and X_{neg}. Thus, the global spatial correlation of the sub-regions must be explored to compare the difference between X and X_{neg}.

C. MULTI-TASK TRAINING

Our proposed architecture is trained in an end-to-end manner by solving multiple tasks. Our work focuses on recognizing different categories of driver actions, and thus, the first task is trained with a softmax loss (as shown in Figure 2(b)). Let us assume that the classification problem is k-class, and we treat the negative samples of all the k classes as the $(k+1)$th class. The softmax loss adopted in this study is defined as

$$\mathcal{L}_1 = -\left(\sum_{c=1}^{k} (\lambda_c \log\ p(\rho = c)) + \lambda_{(k+1)} \log\ p(\rho = k+1)\right). \tag{6}$$

Here, λ_c is a binary indicator (0 or 1), which equals 1 if c is the true label of the input instance and 0 otherwise. ρ is the prediction of the category for the input instance and $p(\rho = c) = \frac{\exp(l_c)}{\sum_{i=1}^{(k+1)} \exp(l_i)}$. $\{l_1, \ldots, l_k, l_{(k+1)}\}$ is a $k+1$ dimension vector of logits, which is output by the network as

$$\mathcal{L}_1 = \text{fc}(v, \text{concat}(\Pi, \pi_{k+1})), \tag{7}$$

where v is the deep feature learned by an adopted network backbone (e.g., the output of the final pooling layer of a ResNet-50), fc denotes the fully connected (fc) layer, and $\Pi = \{\pi_1, \ldots, \pi_k\}$ is a set of parameters of the fc layer. Each parameter in $\{\pi_1, \ldots, \pi_k\}$ corresponds to each logit in $\{l_1, \ldots, l_k\}$. π_{k+1} is an extra parameter that we added to correspond to the $(k+1)$th logit. During the training procedure, we concatenate π_{k+1} with Π as the parameters for the fc layer. Therefore, this task is changed from a k-class classification problem to a $(k+1)$-class classification problem.

As shown in Figure 2(c), the second task is to explore the commonalities and differences between X and X_{pos}, or X and X_{neg}. Let us assume that v, v_{pos}, and v_{neg} are the deep features extracted from X, X_{pos}, and X_{neg}, respectively. The second task is defined as in the following:

$$\mathcal{L}_2 = \max(d(v, v_{pos}) - d(v, v_{neg}) + \sigma, 0), \tag{8}$$

where d denotes the Euclidean distance.

To train the networks, we simultaneously solve the two above-mentioned losses as follows:

$$\mathcal{L} = \mathcal{L}_1 + \gamma \mathcal{L}_2, \tag{9}$$

where γ is a weight coefficient.

During the training procedure, $\mathcal{L}_1$ actually treats X_{neg} as the ''meaningless'' class, which semantically acts as a ''none of the above.'' It can build a default categorization to introduce additional images as a ''meaningless'' class, which provides better separation between the main classes within the feature space [44]–[46]. Specifically, in our work, optimizing $\mathcal{L}_1$ trains the model to know that same-category images do not only have the same local features, but also have the same global structure.

$\mathcal{L}_2$ ensures that the distance between v and v_{pos} is close, while the distance between v and v_{neg} is large. Consequently, the model has to explore clues in terms of the global structure to find differences in X and X_{neg} pairs and similarities between X and X_{pos} pairs. $\mathcal{L}_1$ and $\mathcal{L}_2$ together help the model to reduce local biases and to improve the awareness of the global structure.

During the testing procedure, we remove π_{k+1}, and then the trained networks can be directly used to validate the k-class classification accuracy.

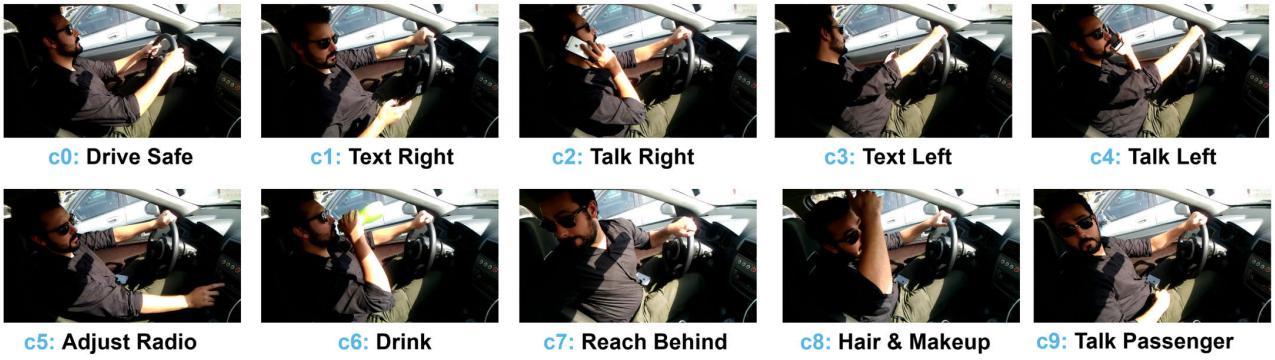


FIGURE 5. Ten different driver behaviors categorized in the AUC Distracted Driver Dataset [13].

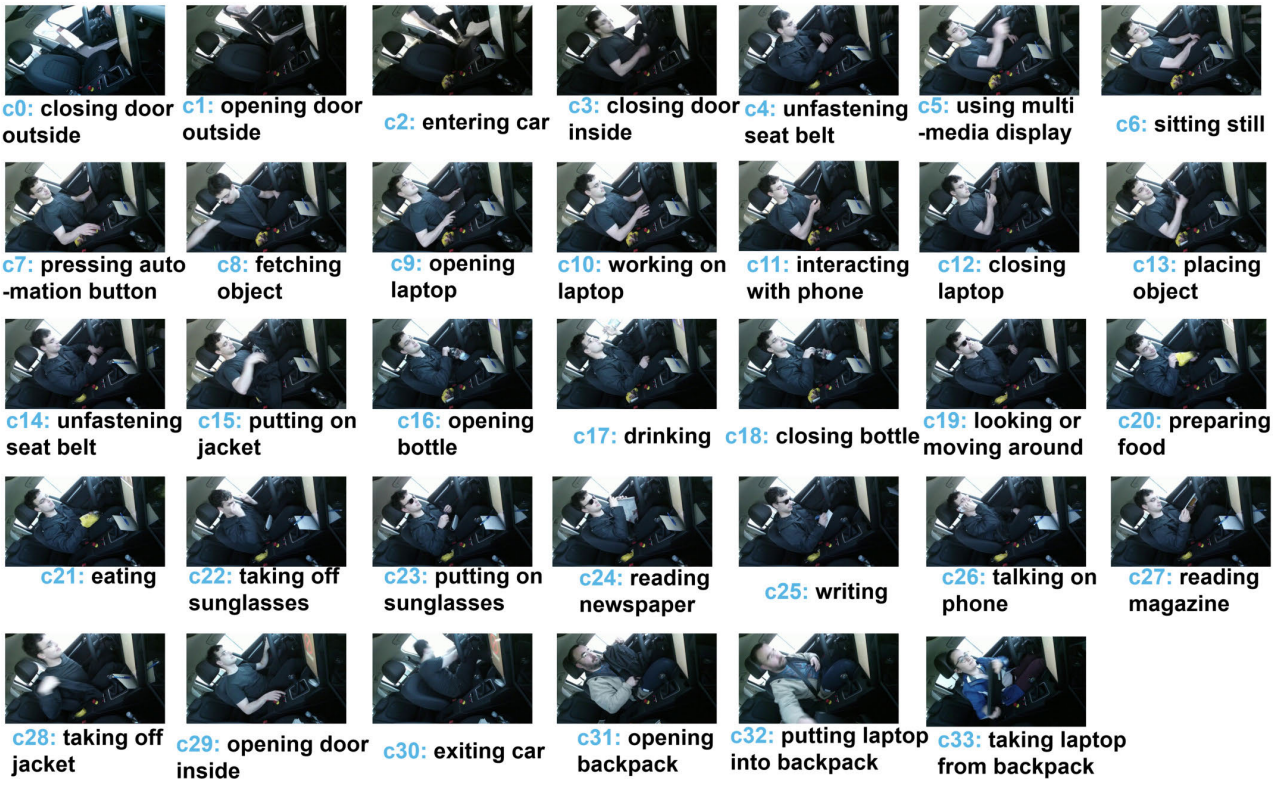


FIGURE 6. Thirty-four different driver behaviors categorized in the Drive and Act Dataset [16].

IV. EXPERIMENTS

In this section, we introduce the experiments for evaluating the effectiveness of the proposed framework. Subsection IV-A introduces the dataset in details. Subsection IV-B introduces the network backbones we use for experiments. Subsection IV-C introduces the setup of experimental environment.

A. DATASET DETAILS

As mentioned before, we carried out experiments on two standard benchmark datasets: the AUC Distracted Driver Dataset [13] and the Drive and Act Dataset [16]. The former requires the recognition of 10 classes of video frame-based distracted driving behavior. It has 17,308 RGB frames, of which 12,977 are for training, while the remaining 4,331 are for testing. The latter requires the recognition of 34 classes of video-clip-based distracted driving behavior.

TABLE 1. Comparison between the proposed approach and baselines in terms of classification accuracy.

AUC Distracted Driver Dataset [13]	Resnet-50		Efficientnet-b3 -pruned	
	Baseline	TML	Baseline	TML
	94.9%	95.6%	94.8%	96.0%
Drive and Act Dataset [16]	C3D		R3D	
	Baseline	TML	Baseline	TML
	57.1%	58.9%	63.7%	66.8%

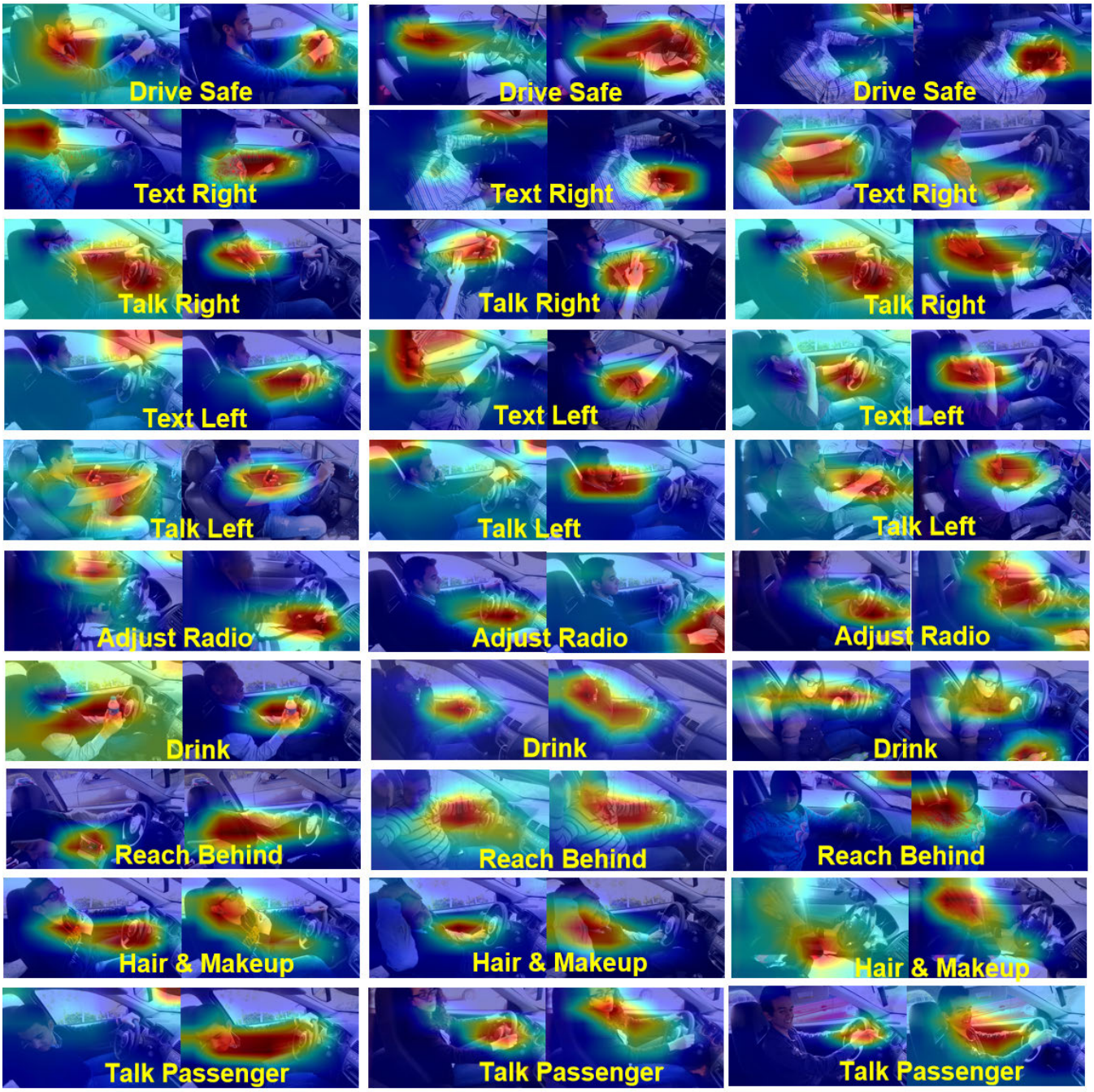


FIGURE 7. Visualization results of the baseline (the left image of each pair) and TML (the right image of each pair).

It has 8,864 video clips, of which 6,642 are for training, while the remaining 2,222 are for testing. The 10 classes of the AUC Distracted Driver Dataset [13] are labeled as c0–c9, and Figure 5 shows the driving behaviors corresponding to c0–c9 in this dataset. The 34 classes of the Drive and Act Dataset [16] are labeled as c0–c33, and Figure 6 show the driving behaviors that correspond to c0–c33 of this dataset.

B. NETWORK BACKBONES

Because the two datasets have different types of data, we use 2D CNNs and three-dimensional (3D) CNNs for video-frame-based and video-clip-based datasets to test our proposed approach. We use Resnet-50 [47] and Efficientnet-b3-pruned [14] as backbones for the video-frame-based dataset, and C3D [48] and R3D [49] for the video-clip-based dataset. We compare the performance of the networks trained using the standard training strategy and the proposed framework.

C. EXPERIMENTAL ENVIRONMENT SETUP

We set the learning rate as 0.01 with cosine annealing [50] for the video-frame-based dataset, and 0.001 with cosine

TABLE 2. Class-wise sensitivity on the AUC Distracted Driver [13].

		(a) Baseline			(b) TML		
Class	Total samples	Right predictions	Wrong predictions	Sensitivity	Right predictions	Wrong predictions	Sensitivity
c0	922	856	66	92.80%	875	47	94.90%
c1	326	305	21	93.60%	315	11	96.60%
c2	341	328	13	96.20%	330	11	96.80%
c3	494	472	22	95.50%	474	20	96.00%
c4	306	288	18	94.10%	290	16	94.80%
c5	305	294	11	96.40%	295	10	96.70%
c6	403	394	9	97.80%	395	8	98.00%
c7	301	280	21	93.00%	284	17	94.40%
c8	290	266	24	91.70%	270	20	93.10%
c9	643	622	21	96.70%	629	14	97.80%
Average	**433.1**	**410.5**	**22.6**	**94.8%**	**415.7**	**17.4**	**96.0%**

TABLE 3. Confusion matrix on the AUC Distracted Driver [13].

(a) Baseline

Ground truth \ Predicted label	c0	c1	c2	c3	c4	c5	c6	c7	c8	c9
c0	856	2	3	6	0	14	11	10	6	14
c1	1	305	14	1	3	1	1	0	0	0
c2	2	9	328	0	0	1	0	0	1	0
c3	9	2	1	472	6	2	1	0	0	1
c4	0	1	0	17	288	0	0	0	0	0
c5	7	0	0	0	0	294	3	0	1	0
c6	4	0	0	0	0	3	394	0	0	2
c7	18	0	0	0	0	0	1	280	0	2
c8	12	1	0	0	0	2	4	3	266	2
c9	13	0	0	0	0	0	7	0	1	622

(b) TML

Ground truth \ Predicted label	c0	c1	c2	c3	c4	c5	c6	c7	c8	c9
c0	875	1	2	3	0	6	3	5	3	24
c1	0	315	5	0	4	0	1	1	0	0
c2	2	9	330	0	0	0	0	0	0	0
c3	9	3	0	474	5	2	0	0	0	1
c4	0	2	0	14	290	0	0	0	0	0
c5	9	0	0	0	0	295	1	0	0	0
c6	4	0	0	0	0	2	395	0	0	2
c7	13	0	0	0	0	0	0	284	0	4
c8	13	0	0	0	0	1	4	1	270	1
c9	7	0	0	0	0	0	5	1	1	629

annealing for the video-clip-based dataset. The batch size was set as 64 for all network backbones except C3D. We set the batch size for training C3D to 16 because of the limited GPU memory in our computer (1080Ti×2). We trained the networks for 100 or 50 epochs on the video frame-based or video-clip-based datasets when the training status was already saturated. We set σ as 0.1, γ as 0.5, and N as 32.

V. RESULTS AND DISCUSSIONS

In this section, we summarize the experimental results and discuss the limitations of the proposed framework. Subsection V-A presents the results of comparing TML with the baselines. Subsection V-B presents the results of comparing TML with the previous studies. Subsection V-C discusses the limitations of the proposed framework.

TABLE 4. Class-wise sensitivity on the Drive and Act Dataset [16].

		(a) Baseline			(b) TML		
Class	Total Samples	Right Predictions	Wrong Predictions	Sensitivity	Right Predictions	Wrong Predictions	Sensitivity
c0	4	1	3	25.0%	4	0	100.0%
c1	6	6	0	100.0%	6	0	100.0%
c2	9	4	5	44.4%	7	2	77.8%
c3	6	5	1	83.3%	5	1	83.3%
c4	20	14	6	70.0%	14	6	70.0%
c5	100	86	14	86.0%	86	14	86.0%
c6	707	622	85	88.0%	622	85	88.0%
c7	60	39	21	65.0%	43	17	71.7%
c8	148	61	87	41.2%	71	77	48.0%
c9	16	0	16	0.0%	0	16	0.0%
c10	52	15	37	28.8%	20	32	38.5%
c11	83	78	5	94.0%	77	6	92.8%
c12	7	0	7	0.0%	0	7	0.0%
c13	122	29	93	23.8%	29	93	23.8%
c14	18	0	18	0.0%	2	16	11.1%
c15	53	12	41	22.6%	12	41	22.6%
c16	32	7	25	21.9%	7	25	21.9%
c17	24	1	23	4.2%	7	17	29.2%
c18	20	3	17	15.0%	8	12	40.0%
c19	17	0	17	0.0%	0	17	0.0%
c20	16	2	14	12.5%	4	12	25.0%
c21	235	177	58	75.3%	182	53	77.4%
c22	8	0	8	0.0%	0	8	0.0%
c23	6	0	6	0.0%	1	5	16.7%
c24	152	115	37	75.7%	127	25	83.6%
c25	53	19	34	35.8%	19	34	35.8%
c26	85	10	75	11.8%	19	66	22.4%
c27	115	104	11	90.4%	104	11	90.4%
c28	23	1	22	4.3%	1	22	4.3%
c29	8	0	8	0.0%	0	8	0.0%
c30	8	5	3	62.5%	7	1	87.5%
c31	5	0	5	0.0%	0	5	0.0%
c32	2	0	2	0.0%	0	2	0.0%
c33	2	0	2	0.0%	0	2	0.0%
Average	**65.4**	**41.6**	**23.7**	**63.6%**	**43.6**	**21.7**	**66.7%**

A. COMPARISON WITH THE BASELINES

Table 1 shows the comparison results obtained between the proposed approach and baselines (our implementation). The proposed approaches and baselines adopt the same networks but are trained with different strategies. For the baselines, the networks are trained with the standard procedure. For the proposed approaches, the networks are trained with TML. Our proposed approach clearly surpasses the baselines for each backbone network on both datasets. Our framework improves 0.7% points with Resnet-50 [47] and 1.2% points with Efficientnet-b3-pruned on the AUC Distracted Driver Dataset [13], and improves 1.8% points with C3D and 3.1% points with R3D on the Drive and Act Dataset [16]. Considering that the accuracy for the AUC Distracted Driver Dataset is almost saturated, it is interesting to see there is still room for the improvement by our proposed method.

Figure 7 visualizes some example CAMs generated by the baseline Efficientnet-b3-pruned (the left image of each pair) and the Efficientnet-b3-pruned trained with the proposed framework (the right image of each pair). Each row shows three examples of one of the ten categories of actions of the AUC Distracted Driver Dataset [13]. The proposed framework allows the network to obtain more accurate clues. For example, for the action "Talk Passenger," the network trained with the proposed approach does not only focus on the driver's hand, but also likely focuses on the face orientation. For actions such as "Drink" and "Text Right," the network trained with the proposed approach mainly focuses on the

TABLE 5. Confusion matrix of the baseline network on the Drive and Act Dataset [16].

Ground truth \ Predicted label	c0	c1	c2	c3	c4	c5	c6	c7	c8	c9	c10	c11	c12	c13	c14	c15	c16	c17	c18	c19	c20	c21	c22	c23	c24	c25	c26	c27	c28	c29	c30	c31	c32	c33
c0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	2	0	0	0
c1	0	6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
c2	0	0	4	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	3	0	0	0	0	0	0	0	0	0
c3	0	0	1	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
c4	0	0	0	0	14	0	1	0	0	0	0	0	0	0	0	0	0	0	2	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0
c5	0	0	0	0	3	86	4	0	1	0	0	0	0	0	0	0	0	0	0	0	0	6	0	0	0	0	0	0	0	0	0	0	0	0
c6	0	0	0	0	0	0	622	0	0	0	0	1	0	0	0	0	0	0	1	0	0	83	0	0	0	0	0	0	0	0	0	0	0	0
c7	0	0	0	0	0	0	10	39	0	0	1	3	0	1	0	0	0	0	0	0	0	5	0	0	0	0	1	0	0	0	0	0	0	0
c8	0	0	0	0	1	0	13	0	61	0	1	2	0	13	0	5	1	0	3	0	1	30	0	1	4	10	1	1	0	0	0	0	0	0
c9	0	0	0	0	1	0	2	0	0	0	2	3	0	0	0	0	1	0	3	0	0	1	0	1	1	0	0	1	0	0	0	0	0	0
c10	0	0	0	0	0	0	0	0	0	0	15	33	0	0	0	0	0	1	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0
c11	0	0	0	0	0	0	0	0	0	0	0	78	0	0	0	0	0	0	1	0	0	2	0	0	1	0	0	1	0	0	0	0	0	0
c12	0	0	0	0	0	0	0	0	0	0	1	2	0	0	0	0	0	0	2	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0
c13	0	0	0	0	2	0	10	0	31	0	1	4	0	29	0	1	1	0	7	0	0	22	0	0	2	9	2	1	0	0	0	0	0	0
c14	0	0	0	0	2	0	0	0	0	0	0	0	0	1	0	0	4	0	2	0	0	6	0	0	0	0	3	0	0	0	0	0	0	0
c15	0	0	0	0	5	0	17	0	11	0	0	0	0	4	0	12	0	1	0	0	0	1	0	0	0	0	2	0	0	0	0	0	0	0
c16	0	0	0	0	0	0	1	0	0	0	0	10	0	0	0	0	7	0	4	0	0	9	0	0	0	1	0	0	0	0	0	0	0	0
c17	0	0	0	0	0	0	5	0	0	0	0	0	0	0	0	0	0	1	0	0	0	16	0	0	0	1	1	0	0	0	0	0	0	0
c18	0	0	0	0	0	0	0	0	0	0	0	4	0	0	0	0	5	0	3	0	0	8	0	0	0	0	0	0	0	0	0	0	0	0
c19	0	0	0	0	0	0	3	0	4	0	0	0	0	1	0	0	0	0	0	0	0	9	0	0	0	0	0	0	0	0	0	0	0	0
c20	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	2	0	2	7	0	0	0	2	0	0	0	0	0	0	0	0
c21	0	0	0	0	0	0	21	0	1	0	1	11	0	0	0	0	2	3	4	0	0	177	0	0	0	9	0	6	0	0	0	0	0	0
c22	0	0	0	0	0	0	1	0	0	0	1	3	0	0	0	0	0	0	2	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
c23	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
c24	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	115	14	0	21	0	0	0	0	0	0
c25	0	0	0	0	0	0	8	0	0	0	0	15	0	0	0	0	0	0	1	0	0	9	0	0	0	19	0	1	0	0	0	0	0	0
c26	0	0	0	0	1	0	60	0	0	0	0	1	0	0	0	0	0	0	0	0	0	13	0	0	0	0	10	0	0	0	0	0	0	0
c27	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	9	0	104	0	0	0	0	0	0
c28	0	0	0	0	0	0	1	0	5	0	0	0	0	3	0	7	0	0	0	0	0	4	0	0	2	0	0	0	1	0	0	0	0	0
c29	0	0	0	0	0	1	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	1	1	0	0	1	0	0	0
c30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	2	0	0	0	0	0	5	0	0	0
c31	0	0	0	0	0	0	0	0	2	0	0	0	0	2	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
c32	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
c33	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

objects related to the action, such as *bottle* or *mobile-phone*. For some other examples, sometimes the baseline networks tend to put their focus everywhere on the image. It can be observed that the network trained with TML captures the visual attention that is more semantically meaningful than those captured by the baseline network. It is because the baseline networks are strongly biased towards local regions, they can consider different unimportant regions as important clues by chance during the training procedure. Therefore, they fail to capture the real attention during the testing procedure. On the contrary, the proposed TML framework can help reduce this bad effect. This shows that our proposed method properly captures the right characteristics of distracted driving, resulting in more reliable and explainable detection.

In this subsection, we also compare the confusion matrix and the class-wise sensitivity between the baseline and the proposed method. As defined in [51] and [52], class-wise sensitivity (S), which is also known as the true positive rate (TPR), is computed as

$$S = \frac{Q_{TP}}{Q_{TP} + Q_{FN}}, \tag{10}$$

where Q_{TP} denotes the true positive (i.e., the number of correctly classified images), and Q_{FN} denotes false negatives (i.e., the number of incorrectly classified images).

Tables 2 (a) and (b) respectively show the class-wise sensitivity computed on the AUC Distracted Driver Dataset [13] of the baseline Efficientnet-b3-pruned and the Efficientnet-b3-pruned trained with TML. Tables 4 (a) and (b) respectively show the class-wise sensitivity computed on the Drive and Act Dataset [16] of the baseline R3D and the R3D trained with TML. It can be observed that, on both datasets, the proposed framework helps the network improve the class-wise sensitivity. On the AUC Distracted Driver Dataset [13], TML improves the class-wise sensitivity in all ten categories. On the Drive and Act Dataset [16], TML improves the class-wise sensitivity in 14 categories out of 34, and the class-wise sensitivity in the other categories do not decrease.

Tables 3 (a) and (b) respectively show the confusion matrix computed on the AUC Distracted Driver Dataset [13] of the baseline Efficientnet-b3-pruned and the Efficientnet-b3-pruned trained with TML. Tables 5 and 6 respectively show the confusion matrix computed on the Drive and Act Dataset [16] of the baseline R3D and the R3D trained

TABLE 6. Confusion matrix of the network trained with TML on the Drive and Act Dataset [16].

Ground truth \ Predicted label	c0	c1	c2	c3	c4	c5	c6	c7	c8	c9	c10	c11	c12	c13	c14	c15	c16	c17	c18	c19	c20	c21	c22	c23	c24	c25	c26	c27	c28	c29	c30	c31	c32	c33
c0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
c1	0	6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
c2	0	0	7	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
c3	0	0	1	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
c4	0	0	0	1	14	0	0	0	2	0	0	0	0	0	0	0	0	0	1	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0
c5	1	0	0	0	1	86	1	1	5	0	0	0	0	0	0	1	0	0	0	0	1	3	0	0	0	0	0	0	0	0	0	0	0	0
c6	0	0	0	0	0	0	622	0	3	0	0	0	0	0	0	0	0	2	0	0	0	72	0	1	0	0	7	0	0	0	0	0	0	0
c7	0	0	0	0	0	0	7	43	3	0	1	0	0	1	0	0	0	0	1	0	0	4	0	0	0	0	0	0	0	0	0	0	0	0
c8	0	0	1	0	1	0	3	0	71	0	0	0	0	13	0	36	0	0	1	0	0	12	0	0	7	2	0	0	0	0	0	0	1	0
c9	0	0	0	0	1	0	1	0	8	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	4	0	0	0	0	0	0	0	0	0
c10	0	0	0	0	0	0	1	0	4	0	20	12	0	1	0	0	0	1	0	0	0	9	2	0	1	1	0	0	0	0	0	0	0	0
c11	0	0	0	0	0	0	0	0	1	0	1	77	0	0	0	0	1	0	1	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0
c12	0	0	0	0	0	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	2	0	0	1	1	0	0	0	0	0	0	0	0
c13	0	0	0	0	1	0	1	0	65	0	0	2	0	29	0	1	0	0	0	0	0	9	0	0	3	7	0	0	0	0	0	0	4	0
c14	0	0	0	0	1	0	0	0	8	0	0	0	0	0	2	0	0	0	1	0	0	5	0	1	0	0	0	0	0	0	0	0	0	0
c15	0	0	0	0	1	0	0	0	36	0	1	0	0	0	2	12	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
c16	0	0	0	0	0	0	2	0	1	0	0	0	0	0	0	0	7	0	3	0	0	16	1	0	1	1	0	0	0	0	0	0	0	0
c17	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	7	0	0	2	12	0	0	0	0	1	0	0	0	0	0	0	0
c18	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	8	0	0	11	0	0	0	0	0	0	0	0	0	0	0	0
c19	0	0	0	0	0	0	4	3	9	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
c20	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	4	10	0	0	0	0	0	0	0	0	0	0	0	0
c21	0	0	0	0	0	0	12	1	3	0	7	0	0	0	0	0	0	2	13	0	2	182	0	0	1	6	3	3	0	0	0	0	0	0
c22	0	0	0	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0	2	0	0	2	0	1	0	0	0	0	0	0	0	0	0	0
c23	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	2	1	0	0	1	0	0	0	0	0	0	0
c24	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	127	4	0	19	0	0	0	0	0	0
c25	0	0	0	0	0	0	11	0	4	0	0	2	0	0	0	0	0	0	4	0	0	10	0	0	0	19	0	3	0	0	0	0	0	0
c26	0	0	0	0	0	0	44	3	5	0	0	0	0	0	1	0	0	0	0	0	0	13	0	0	0	0	19	0	0	0	0	0	0	0
c27	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	7	2	0	104	0	0	0	0	0	0
c28	0	0	0	0	0	0	0	0	15	0	0	0	0	0	1	5	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0
c29	1	0	0	0	0	0	0	0	4	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0
c30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	7	0	0
c31	0	0	0	0	0	0	0	0	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
c32	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
c33	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

with TML. The confusion matrix describes the performance of a classification model by counting how many samples of each category are predicted as each category. It can be observed that, on both datasets, the number of correctly predicted samples increases.

Overall, the proposed method outperforms the baselines considering all perspectives.

B. COMPARISON WITH STATE-OF-THE-ART STUDIES

In this subsection, we compare our work with state-of-the-art studies on the AUC Distracted Driver Dataset [13] and the Drive and Act Dataset [16]. We find that, after the training, the accuracy can be further improved if we fuse the prediction scores respectively obtained from the raw input and positive sample. Thus, in this subsection, we both present the accuracy obtained only by raw input images (X) and the accuracy obtained by fusing the prediction scores respectively obtained from the raw input and positive sample ($X + X_{pos}$).

Table 7 shows the comparison results obtained with prior studies on the AUC Distracted Driver Dataset [13], and Table 8 shows the comparison results obtained with prior studies on the Drive and Act Dataset [16].

TABLE 7. Comparison with state-of-the-art studies on AUC Distracted Driver Dataset [13].

InceptionV3-LSTM [53]	89.8%
C-SLSTM [53]	92.7%
GA-Weighted ensemble of Inception-V3 [54]	90.1%
Densent + Latent Pose [55]	94.2%
NasNet mobile [56]	94.7%
Multi-stream fusion [36]	95.6%
Decreasing-HOG convolutional neural network [37]	95.6%
Major voting ensemble [10]	95.8%
GA weighted ensemble [10]	95.9%
VGG with regularization [57]	96.3%
TML (X)	**96.0%**
TML (X+X_{pos})	**96.3%**

Figure 8 shows some typical failure cases of the 3.7% wrongly-predicted images. Most failure cases are actually the beginning frame of a sequence of frames that composes a certain distracted driving behavior. In other words, in those failure cases, the driver is about to start a certain distracted driving behavior. Thus, those images actually look very

TABLE 8. Comparison with state-of-the-art studies on Drive and Act Dataset [16].

Pose [16]	44.36%
Interior [16]	40.30%
2-Stream [58]	45.39%
3-Stream [59]	46.95%
C3D [48]	57.1%
P3D [60]	45.32%
I3D [61]	63.64%
TML (X)	**66.8%**
TML $(X+X_{pos})$	**66.9%**

FIGURE 8. Some typical example images that are wrongly classified by TML on the AUC Distracted Driver Dataset [13]. The category name below each image is its ground truth, but all the images are predicted as "Drive Safe."

similar to "Drive Safe," and are very difficult even for humans to recognize.

Overall, our best result achieves the state-of-the-art performance observed for both datasets. Moreover, after the training, our approach does not require any more extra computational cost. In comparison, Abouelnaga *et al.* [10] required multiple streams of network backbones to gather the information obtained from different regions (e.g., head, hand, body, etc.) during the testing procedure. Martin *et al.* [59] required three-stream network backbones to gather the information obtained from RGB frames, optical flows, and skeletons. Thus, our approach not only has better performance but also requires less computation cost during the testing procedure.

C. LIMITATIONS OF THE PROPOSED FRAMEWORK

Currently, the best accuracy of the proposed study is achieved by averaging the log its respectively outputted with the raw input and positive sample. It would be better if the best accuracy can be achieved with only the raw input image. Besides, the current model requires around 90 million parameters, which is less than the previous state-of-the-art model [57] (160 million parameters), but still, it would be better for real-world application if we can make it less.

VI. CONCLUSION

In this paper, we propose a triple-wise multi-task learning (TML) framework to improve the accuracy of distracted driver recognition tasks. CNNs have been proven to exhibit bias towards local features, which sometimes causes the models to fail to focus on semantically meaningful regions for finding clues. Our framework firstly generates positive and negative samples of the given inputs. Then our framework trains the network backbone with different tasks that include (a) recognizing the raw input and positive sample as the given ground truth and recognizing the negative sample as an extra "meaningless" label, and (b) pulling closer the distance between the features obtained from the raw input and positive sample while pushing away the distance between the features obtained from the raw input and negative sample. Those tasks force the CNNs to improve their awareness of global spatial structure by requiring the CNNs to explore the commonalities and differences between the raw images and positive/negative samples. The experimental results show that the proposed framework helps the model to learn more accurate clues from the videos. For our future plans, we mainly plan to: (a) collect data by ourselves and test the generalization ability of the model; (b) decrease the parameter size while keeping the performance so that our work will be more suitable for real-world application; and (c) compare our work with graph neural networks [62], which have similar function to our work of understanding the relationship between different regions.

REFERENCES

[1] *Overview of the 2019 Crash Investigation Sampling System*, Traffic Safety Facts: Research Note, National Highway Traffic Safety Administration, Washington, DC, USA, 2020, pp. 1–14.
[2] M. Gjoreski, M. Z. Gams, M. Lustrek, P. Genc, J.-U. Garbas, and T. Hassan, "Machine learning and End-to-End deep learning for monitoring driver distractions from physiological and visual signals," *IEEE Access*, vol. 8, pp. 70590–70603, 2020.
[3] Y. Liao, G. Li, S. E. Li, B. Cheng, and P. Green, "Understanding driver response patterns to mental workload increase in typical driving scenarios," *IEEE Access*, vol. 6, pp. 35890–35900, 2018.
[4] J. M. Ramirez, M. D. Rodriguez, A. G. Andrade, L. A. Castro, J. Beltran, and J. S. Armenta, "Inferring drivers' visual focus attention through head-mounted inertial sensors," *IEEE Access*, vol. 7, pp. 185422–185432, 2019.
[5] A. Plebe and M. D. Lio, "On the road with 16 neurons: Towards interpretable and manipulable latent representations for visual predictions in driving scenarios," *IEEE Access*, vol. 8, pp. 179716–179734, 2020.
[6] A. Zhu, S. Cao, H. Yao, M. Jadliwala, and J. He, "Can wearable devices facilitate a driver's brake response time in a classic car-following task?" *IEEE Access*, vol. 8, pp. 40081–40087, 2020.
[7] B. Lv, R. Yue, and Y. Zhang, "The influence of different factors on right-turn distracted driving behavior at intersections using naturalistic driving study data," *IEEE Access*, vol. 7, pp. 137241–137250, 2019.
[8] *2015 Motor Vehicle Crashes: Overview*, Traffic Safety Facts: Research Note, National Highway Traffic Safety Administration, Washington, DC, USA, 2016, pp. 1–9.
[9] S. N. Resalat and V. Saba, "A practical method for driver sleepiness detection by processing the EEG signals stimulated with external flickering light," *Signal, Image Video Process.*, vol. 9, no. 8, pp. 1751–1757, Nov. 2015.
[10] Y. Abouelnaga, H. M. Eraqi, and M. N. Moustafa, "Real-time distracted driver posture classification," 2017, *arXiv:1706.09498*. [Online]. Available: http://arxiv.org/abs/1706.09498
[11] Y. Zhang, Y. Chen, and C. Gao, "Deep unsupervised multi-modal fusion network for detecting driver distraction," *Neurocomputing*, vol. 421, pp. 26–38, Jan. 2021.
[12] G. Lechner, M. Fellmann, A. Festl, C. Kaiser, T. E. Kalayci, M. Spitzer, and A. Stocker, "A lightweight framework for multi-device integration and multi-sensor fusion to explore driver distraction," in *Proc. Int. Conf. Adv. Inf. Syst. Eng.* Rome, Italy: Springer, 2019, pp. 80–95.

[13] M. Alotaibi and B. Alotaibi, "Distracted driver classification using deep learning," *Signal, Image Video Process.*, vol. 14, no. 3, pp. 617–624, 2019.
[14] Y. Aflalo, A. Noy, M. Lin, I. Friedman, and L. Zelnik, "Knapsack pruning with inner distillation," 2020, *arXiv:2002.08258*. [Online]. Available: https://arxiv.org/abs/2002.08258
[15] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2921–2929.
[16] M. Martin, A. Roitberg, M. Haurilet, M. Horne, S. ReiB, M. Voit, and R. Stiefelhagen, "Drive&act: A multi-modal dataset for fine-grained driver behavior recognition in autonomous vehicles," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 2801–2810.
[17] A. Kashevnik, R. Shchedrin, C. Kaiser, and A. Stocker, "Driver distraction detection methods: A literature review and framework," *IEEE Access*, vol. 9, pp. 60063–60076, 2021.
[18] C. Yan, F. Coenen, and B. Zhang, "Driving posture recognition by convolutional neural networks," *IET Comput. Vis.*, vol. 10, no. 2, pp. 103–114, 2016.
[19] R. Geirhos, P. Rubisch, C. Michaelis, M. Bethge, F. AWichmann, and W. Brendel, "ImageNet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness," in *Proc. Int. Conf. Learn. Represent.*, 2019, pp. 1–22.
[20] M. A. Islam, M. Kowal, P. Esser, S. Jia, B. Ommer, K. G. Derpanis, and N. Bruce, "Shape or texture: Understanding discriminative features in CNNs," 2021, *arXiv:2101.11604*. [Online]. Available: http://arxiv.org/abs/2101.11604
[21] A. Doshi and M. M. Trivedi, "On the roles of eye gaze and head dynamics in predicting driver's intent to change lanes," *IEEE Trans. Intell. Transp. Syst.*, vol. 10, no. 3, pp. 453–462, Sep. 2009.
[22] I. Teyeb, O. Jemai, M. Zaied, and C. Ben Amar, "A novel approach for drowsy driver detection using head posture estimation and eyes recognition system based on wavelet network," in *Proc. 5th Int. Conf. Inf., Intell., Syst. Appl. (IISA)*, Jul. 2014, pp. 379–384.
[23] Y. Yao, X. Zhao, X. Feng, and J. Rong, "Assessment of secondary tasks based on drivers' eye-movement features," *IEEE Access*, vol. 8, pp. 136108–136118, 2020.
[24] P. Watta, S. Lakshmanan, and Y. Hou, "Nonparametric approaches for estimating driver pose," *IEEE Trans. Veh. Technol.*, vol. 56, no. 4, pp. 2028–2041, Jul. 2007.
[25] E. Murphy-Chutorian and M. M. Trivedi, "Head pose estimation and augmented reality tracking: An integrated system and evaluation for monitoring driver awareness," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 2, pp. 300–311, Jun. 2010.
[26] I. Teyeb, O. Jemai, M. Zaied, and C. B. Amar, "A drowsy driver detection system based on a new method of head posture estimation," in *Proc. Int. Conf. Intell. Data Eng. Automated Learn.*, 2014, pp. 362–369.
[27] L. M. Bergasa, J. Nuevo, M. A. Sotelo, R. Barea, and M. E. Lopez, "Real-time system for monitoring driver vigilance," *IEEE Trans. Intell. Transp. Syst.*, vol. 7, no. 1, pp. 63–77, Mar. 2006.
[28] O. Jemai, I. Teyeb, and T. Bouchrika, "A novel approach for drowsy driver detection using eyes recognition system based on wavelet network," *Int. J. Recent Contrib. Eng., Sci. IT*, vol. 1, no. 1, pp. 46–52, 2013.
[29] J. Lei, Q. Han, L. Chen, Z. Lai, L. Zeng, and X. Liu, "A novel side face contour extraction algorithm for driving fatigue statue recognition," *IEEE Access*, vol. 5, pp. 5723–5730, 2017.
[30] S. Y. Cheng, S. Park, and M. M. Trivedi, "Multi-spectral and multi-perspective video arrays for driver body tracking and activity analysis," *Comput. Vis. Image Understand.*, vol. 106, nos. 2–3, pp. 245–257, 2007.
[31] C. Tran, A. Doshi, and M. M. Trivedi, "Modeling and prediction of driver behavior by foot gesture analysis," *Comput. Vis. Image Understand.*, vol. 116, no. 3, pp. 435–445, 2012.
[32] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, vol. 25. Stateline, NV, USA, Dec. 2012, pp. 1097–1105.
[33] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2818–2826.
[34] M. R. Arefin, F. Makhmudkhujaev, O. Chae, and J. Kim, "Aggregating CNN and HOG features for real-time distracted driver detection," in *Proc. IEEE Int. Conf. Consum. Electron. (ICCE)*, Jan. 2019, pp. 1–3.
[35] Y. Hu, M. Lu, and X. Lu, "Feature refinement for image-based driver action recognition via multi-scale attention convolutional neural network," *Signal Process., Image Commun.*, vol. 81, Feb. 2020, Art. no. 115697.
[36] A. Behera, Z. Wharton, A. Keidel, and B. Debnath, "Deep CNN, body pose and body-object interaction features for Drivers' activity monitoring," *IEEE Trans. Intell. Transp. Syst.*, early access, Oct. 12, 2020, doi: 10.1109/TITS.2020.3027240.
[37] B. Qin, J. Qian, Y. Xin, B. Liu, and Y. Dong, "Distracted driver detection based on a CNN with decreasing filter size," *IEEE Trans. Intell. Transp. Syst.*, early access, Mar. 17, 2021, doi: 10.1109/TITS.2021.3063521.
[38] W. Brendel and M. Bethge, "Approximating CNNs with bag-of-local-features models works surprisingly well on ImageNet," 2019, *arXiv:1904.00760*. [Online]. Available: https://arxiv.org/abs/1904.00760
[39] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.
[40] P. Li, Y. Xu, Y. Wei, and Y. Yang, "Self-correction for human parsing," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 12, pp. 1–12, 2020.
[41] X. Chen, R. Mottaghi, X. Liu, S. Fidler, R. Urtasun, and A. Yuille, "Detect what you can: Detecting and representing objects using holistic models and body parts," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 1971–1978.
[42] W. Y. J. L. C. Wei and W. Wang, "Deep retinex decomposition for low-light enhancement," in *Proc. Brit. Mach. Vis. Conf.*, 2018, pp. 1–12.
[43] X. Guo, Y. Li, and H. Ling, "LIME: Low-light image enhancement via illumination map estimation," *IEEE Trans. Image Process.*, vol. 26, no. 2, pp. 982–993, Feb. 2017.
[44] M. Loncaric. (2018). *Handling 'Background' Classes in Machine Learning*. Accessed: Jan. 2021. [Online]. Available: https://thehive.ai/insights/handling-background-classes-in-machine-lear%ning
[45] M. McCoyd and D. Wagner, "Background class defense against adversarial examples," in *Proc. IEEE Secur. Privacy Workshops (SPW)*, May 2018, pp. 96–102.
[46] F. S. Saleh, M. S. Aliakbarian, M. Salzmann, L. Petersson, and J. M. Alvarez, "Bringing background into the foreground: Making all classes equal in weakly-supervised video semantic segmentation," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2125–2135.
[47] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
[48] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3D convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 4489–4497.
[49] K. Hara, H. Kataoka, and Y. Satoh, "Can spatiotemporal 3D CNNs retrace the history of 2D CNNs and ImageNet?" in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6546–6555.
[50] I. Loshchilov and F. Hutter, "SGDR: Stochastic gradient descent with warm restarts," in *Proc. 5th Int. Conf. Learn. Represent.*, 2017, pp. 1–16.
[51] B. Baheti, S. Talbar, and S. Gajre, "Towards computationally efficient and realtime distracted driver detection with MobileVGG network," *IEEE Trans. Intell. Vehicles*, vol. 5, no. 4, pp. 565–574, Dec. 2020.
[52] 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. 109335–109349, 2020.
[53] J. M. Mase, P. Chapman, G. P. Figueredo, and M. Torres Torres, "A hybrid deep learning approach for driver distraction detection," in *Proc. Int. Conf. Inf. Commun. Technol. Converg. (ICTC)*, Oct. 2020, pp. 1–6.
[54] H. M. Eraqi, Y. Abouelnaga, M. H. Saad, and M. N. Moustafa, "Driver distraction identification with an ensemble of convolutional neural networks," *J. Adv. Transp.*, vol. 2019, pp. 1–12, Feb. 2019.
[55] A. Behera and A. H. Keidel, "Latent body-pose guided DenseNet for recognizing driver's fine-grained secondary activities," in *Proc. 15th IEEE Int. Conf. Adv. Video Signal Based Surveill. (AVSS)*, Nov. 2018, pp. 1–6.
[56] F. Saxen, P. Werner, S. Handrich, E. Othman, L. Dinges, and A. Al-Hamadi, "Face attribute detection with MobileNetV2 and NasNet-mobile," in *Proc. 11th Int. Symp. Image Signal Process. Anal. (ISPA)*, Sep. 2019, pp. 176–180.
[57] B. Baheti, S. Gajre, and S. Talbar, "Detection of distracted driver using convolutional neural network," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2018, pp. 1032–1038.
[58] H. Wang and L. Wang, "Modeling temporal dynamics and spatial configurations of actions using two-stream recurrent neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 499–508.
[59] M. Martin, J. Popp, M. Anneken, M. Voit, and R. Stiefelhagen, "Body pose and context information for driver secondary task detection," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2018, pp. 2015–2021.

[60] Z. Qiu, T. Yao, and T. Mei, "Learning spatio-temporal representation with pseudo-3D residual networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 5533–5541.
[61] J. Carreira and A. Zisserman, "Quo vadis, action recognition? A new model and the kinetics dataset," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6299–6308.
[62] J. Zhou, G. Cui, S. Hu, Z. Zhang, C. Yang, Z. Liu, L. Wang, C. Li, and M. Sun, "Graph neural networks: A review of methods and applications," *AI Open*, vol. 1, pp. 57–81, 2020.

DICHAO LIU (Member, IEEE) received the M.S. degree in information science from Nagoya University, in 2018. He is currently pursuing the Ph.D. degree with the Graduate School of Informatics, Nagoya University. His current research interests include fine-grained image classification and fine-grained human action recognition.

TOSHIHIKO YAMASAKI (Member, IEEE) received the Ph.D. degree from The University of Tokyo, in 2004. He is currently an Associate Professor with the Department of Information and Communication Engineering, Graduate School of Information Science and Technology, The University of Tokyo. He was a JSPS Fellow of Research Abroad and a Visiting Scientist with Cornell University, from February 2011 to February 2013. His current research interests include attractiveness computing based on multimedia big data analysis, computer vision, pattern recognition, and machine learning.

YU WANG (Member, IEEE) received the M.S. degree in information science and the Ph.D. degree in engineering from Nagoya University, in 2010 and 2013, respectively. He is currently an Assistant Professor with the College of Information Science and Engineering, Ritsumeikan University.

KENJI MASE (Senior Member, IEEE) received the B.E. degree in electrical engineering and the M.E. and Ph.D. degrees in information engineering from Nagoya University, in 1979, 1981, and 1992, respectively. He joined the Nippon Telegraph and Telephone Corporation NTT, in 1981, and he had been with NTT Human Interface Laboratories. He was a Visiting Researcher with the Media Laboratory, MIT, from 1988 to 1989. He was with the Advanced Telecommunications Research Institute (ATR), from 1995 to 2002. He became a Professor with Nagoya University, in August 2002. He has been a Research Supervisor of JST CREST on Symbiotic Interactions, since 2017. He is currently with the Graduate School of Informatics, Nagoya University. His research interests include gesture recognition, computer graphics, artificial intelligence and their applications for computer-aided communications, wearable/ubiquitous computers, and lifelog. He is a fellow of the Institutes of Electronics, Information and Communication Engineers (IEICE) of Japan, and a member of the Information Processing Society of Japan (IPSJ), Japan Society of Artificial Intelligence (JSAI), Virtual Reality Society of Japan, Human Interface Society of Japan, and ACM, and a Senior Member of IEEE Computer Society. He was the Section Chair of IEEE Nagoya Section, from 2014 to 2015. He is the 24th and 25th Associate Member of Science Council of Japan.

JIEN KATO (Senior Member, IEEE) received the M.E. and Ph.D. degrees in information engineering from Nagoya University, in 1990 and 1993, respectively. In 1999, she was a Visiting Researcher with the University of Oxford, for one year. Then, she became an Assistant Professor with Toyama University, and an Associate Professor with the Graduate School of Engineering, Nagoya University, in 2000. She has been a Professor with the College of Information Science and Engineering, Ritsumeikan University, since 2018. Her research interests include object recognition, visual event recognition, and machine learning. She is a member of IEICE, IPSJ, and JSAI.