2020

# Detecting phone-related pedestrian distracted behaviours via a two-branch convolutional neural network

Humberto Saenz
*The University of Texas Rio Grande Valley*

Huiming Sun

Lingtao Wu

Xuesong Zhou

Hongkai Yu



## Recommended Citation

Saenz, H, Sun, H, Wu, L, Zhou, X, Yu, H. Detecting phone-related pedestrian distracted behaviours via a two-branch convolutional neural network. IET Intell Transp Sy. 2020; 1– 12. https://doi.org/10.1049/itr2.12012

Received: 29 April 2020 | Revised: 14 September 2020 | Accepted: 1 October 2020

DOI: 10.1049/itr2.12012

ORIGINAL RESEARCH PAPER

The Institution of Engineering and Technology WILEY

# Detecting phone-related pedestrian distracted behaviours via a two-branch convolutional neural network

**Humberto Saenz**[1,2] | **Huiming Sun**[1] | **Lingtao Wu**[3] | **Xuesong Zhou**[4] | **Hongkai Yu**[1]

[1] Department of Electrical Engineering and Computer Science, Cleveland State University, Cleveland, Ohio, USA

[2] Department of Computer Science, University of Texas-Rio Grande Valley, Edinburg, Texas, USA

[3] Texas A&M Transportation Institute, College Station, Texas, USA

[4] School of Sustainable Engineering and the Built Environment, Arizona State University, Tempe, Arizona, USA

**Correspondence**
Hongkai Yu, Department of Electrical Engineering and Computer Science, Cleveland State University, Cleveland, OH 44115, USA.
Email: h.yu19@csuohio.edu

**Funding information**
NVIDIA GPU Grant; Dwight David Eisenhower Transportation Fellowship Program; Amazon Web Services (AWS) Cloud Credits for Research Award

**Abstract**

The distracted phone-use behaviours among pedestrians, like *Texting*, *Game Playing* and *Phone Calls*, have caused increasing fatalities and injuries. However, the research of phone-related distracted behaviour by pedestrians has not been systemically studied. It is desired to improve both the driving and pedestrian safety by automatically discovering the phone-related pedestrian distracted behaviours. Herein, a new computer vision-based method is proposed to detect the phone-related pedestrian distracted behaviours from a view of intelligent and autonomous driving. Specifically, the first end-to-end deep learning based Two-Branch Convolutional Neural Network (CNN) is designed for this task. Taking one synchronised image pair by two front on-car GoPro cameras as the inputs, the proposed two-branch CNN will extract features for each camera, fuse the extracted features and perform a robust classification. This method can also be easily extended to video-based classification by confidence accumulation and voting. A new benchmark dataset of 448 synchronised video pairs of 53,760 images collected on a vehicle is proposed for this research. The experimental results show that using two synchronised cameras obtained better performance than using one single camera. Finally, the proposed method achieved an overall best classification accuracy of 84.3% on the new benchmark when compared to other methods.

## 1 | INTRODUCTION

Pedestrians are among the most vulnerable road users, and both the number of pedestrian fatalities and percentage of pedestrian fatalities have been continuously increasing in recent years according to the Fatality Analysis Reporting System (FARS). [1] Taking Texas as an example, Texas has quite high numbers of pedestrian fatalities from 2016 to 2018, that is, 676, 605, and 622 pedestrian fatalities in 2016, 2017, and 2018, respectively, which are the three highest counts of the past decade in Texas. Although there is a decline in 2017, the overall trend of pedestrian fatalities in Texas in recent decade is increasing at an average annual rate of 7.7%. [2] Likewise, the total pedestrian fatalities in the United States are also increasing in the past decade, reaching the highest count of 6,283 in 2018. [3] The detailed trends are shown in Figure 1.

Compared to other collisions, the consequence of a pedestrian-related crash tends to be more severe for the people involved. One major reason for pedestrian-related collisions is phone-related distractions: pedestrians are distracted and engaged with their mobile phones or devices (texting, video watching, map viewing, game playing, phone calls, etc.), which will increase the probability of accidents. [4, 5] Studies have shown that the percentage of pedestrians fatalities while using mobile phones has risen from less than 1% in 2004 to more than 3.5% in 2010, and the number of pedestrians injured while engaged with their mobile phones has more than doubled since 2005. [4, 5] Reducing distracted behaviours is one of the emphasis areas in the Strategic Highway Safety Plan (SHSP) in Texas. [6]

Safety analysts from interdisciplinary fields (e.g. engineering, psychology, public health, computer science) have made

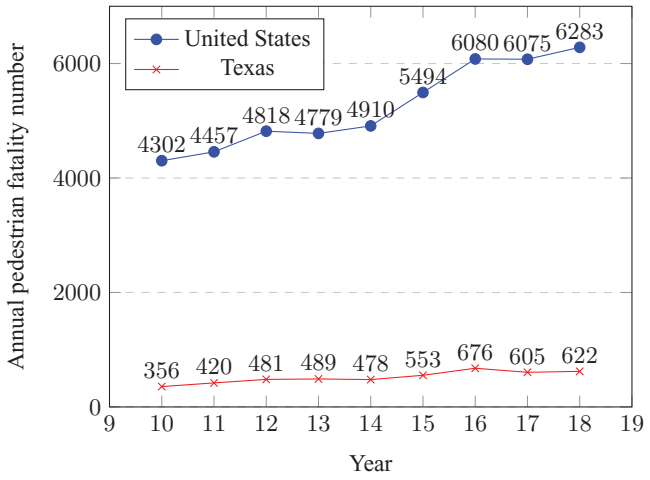


**FIGURE 1** Annual pedestrian fatality number in Texas and United States in recent years from 2010 to 2018

efforts to understand and reduce distracted behaviours. However, nearly all the focus has been placed on driver distractions, [7–12] and the research of phone-related pedestrian distractions has not been systemically studied. Previous studies [13, 14] show that distracted walking with cell phones or other handheld devices will cause severe pedestrian safety problems and some interventions should be applied to improve the pedestrian safety. It is the goal of this research to improve both the driving and pedestrian safety by progressing technology capable of automatically discovering the phone-related distracted pedestrian behaviours. By accurately monitoring the phone-related distracted pedestrian behaviours, interventions can be applied to improve traffic safety if a dangerous event is detected, such as sending warning messages or signals to the driver (e.g. alarm) and the pedestrian (e.g. Bluetooth communication). Considering the low costs of cameras installed on vehicles and the popular availability of mobile Bluetooth communication, the implementation of the proposed system in this article is not very difficult.

The pioneering work in phone-related pedestrian distraction detection using computer vision is proposed by Rangesh et al., [4, 5] where phone-related pedestrian distractions are detected by designing a combined machine learning framework involving phone location, pose estimation and pattern recognition. However, the proposed framework in [4, 5] containing multiple cues is not an end-to-end learning system and it utilises one single image as input. End-to-end learning system means to learn/optimise the system parameters by only considering the inputs and outputs using back propagation, while [4, 5] include many hand-crafted separate steps. Modern machine learning and deep learning techniques prefer end-to-end learning frameworks because the end-to-end framework is robust as an entirely unified network, easy to be re-realised, and convenient to be extended to other deep learning networks. In this article, we made significant progress compared to [4, 5] based on the following key observations: (1). Deep learning-based methods like Convolutional Neural Network (CNN) [15–18] show robust and advanced performance in computer vision tasks and can be learned in an end-to-end manner; (2). Multiple-camera input for computer vision tasks is usually better and more robust than single camera input. [19–21] Based on these key observations, we propose a new computer vision based method to detect the phone-related pedestrian distracted behaviours. With the detection result of the proposed computer vision based method, we could implement safety interventions by sending warning messages to the driver (e.g. alarm) and pedestrian (e.g. Bluetooth communication) to improve the traffic safety. The whole framework leads to a proposed Artificial Intelligence (AI) system, which is shown in Figure 2. Because the technologies in safety interventions are well known, this article is focused on the research of designing a computer vision based method to detect phone-related pedestrian distracted behaviours for a better and accurate intervention. Specifically, we design a new end-to-end deep learning based Two-branch CNN for this task. Taking one synchronised image pair by two front on-car GoPro cameras as input instead of one single image as input in [4, 5], the proposed Two-Branch CNN will extract features for each camera, fuse the extracted features and perform a robust classification. The proposed method can also be easily extended to video-based classification by confidence accumulation and voting.

In this research, we define three classes for the phone-related pedestrian distracted behaviours: Class 1: No Engagement, Class 2: Eye Engagement, and Class 3: Phone Call Engagement. To test this research, we collected a new benchmark dataset of 448 synchronised video pairs of 53,760 images on a real intelligent car, which is named as *PPDB* Benchmark for detecting *Phone-related Pedestrian Distracted Behaviors*. The experimental results show that using two synchronised cameras obtained better performance than using one single camera. Finally, the proposed method achieved an overall best classification accuracy of 84.3% on the collected benchmark compared to other comparison methods.

In summary, our main contributions in this article are as follows: 1. We propose the first end-to-end deep learning framework for phone-related pedestrian distraction detection. 2. We propose a new Two-branch CNN that takes synchronised image pair of two cameras as input to detect phone-related pedestrian distractions. 3. We propose a new large dataset of PPDB Benchmark for this research problem.

## 2 | PREVIOUS WORK

### 2.1 | Driver distraction

Many previous works to improve driving safety are about driver distractions. [7–11] Strayer et al. [7] found that using the cell phone during the simulated driving slowed reaction speed by approximately 18% and increased the risk of collision. Liang et al. [8] used eye movements and driving performance data to evaluate the driver distractions. Pettitt et al. [9] tried to define driver distractions. Brodsky et al. [10] explored the effects of driver music engagement to the driving safety. Przybyla et al. [11] estimated the risk effects of distracted driving, by incorporating a dynamic, data-driven car-following model in an

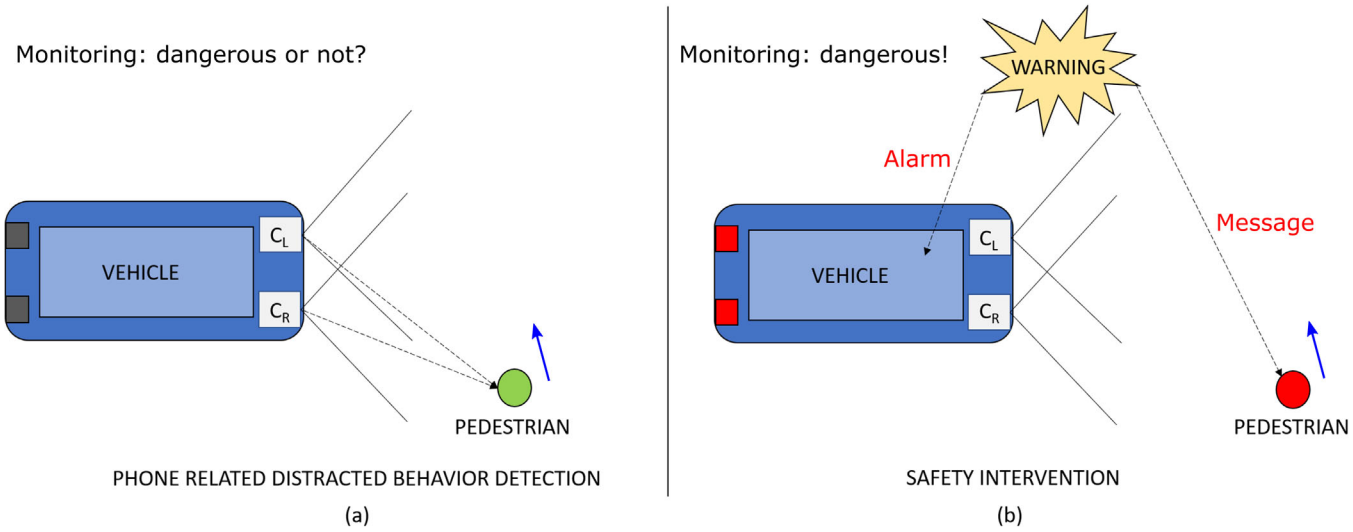


**FIGURE 2** The proposed artificial intelligence (AI) system to improve the traffic safety: (a) a computer vision based method to detect phone-related pedestrian distracted behaviours by two cameras installed in the front-car left and right sides, (b) safety intervention by sending warning messages to the driver (e.g. alarm) and pedestrian (e.g. Bluetooth communication) based on the result of (a). This article is focused on the research task of (a), and the research task of (b) is the future work of this article

algorithmic framework, which also incorporates the probabilities of driver distraction. These previous research problems are different with the proposed research in this article that is focused on phone-related pedestrian distractions, not the driver distractions.

## 2.2 | Pedestrian distraction

Distracted walking with cell phones or other hand-held devices will cause severe pedestrian safety problems and some interventions should be applied to improve the pedestrian safety. [13, 14] Recent studies [22, 23] also show that many traffic accidents are caused by pedestrian distracted behaviours, where 35% of distractions reported are cell phone related. [23] Some solutions to improve pedestrian safety are to install applications on cell phones [24] and detect wearable-device motion, [25] but these kinds of applications or wearable device solutions are still not widely used due to many constraints in the real world. Many techniques are used to analyse the pedestrian behaviours, such as gait analysis, [26] intention prediction, [27] path prediction, [28, 29] activity recognition, [30–32] however they do not focus on the pedestrian distracted behaviours. Because many cameras are already installed on intelligent and autonomous vehicles, computer vision techniques can be used to solve this problem. The pioneering work in phone-related pedestrian distraction detection using computer vision is proposed by Rangesh et al., [4, 5] where phone-related pedestrian distractions are detected by designing a combined machine learning framework involving phone location, pose estimation and pattern recognition with one single image as input. However, the proposed method in [4, 5] containing hand-crafted separate steps is not an end-to-end learning system. The proposed work in this article is an end-to-end deep learning method using synchronised image/video pair as input.

## 2.3 | Benchmark dataset

Many current computer vision benchmarks have been proposed for human related factors, such as human detection, [33] human pose estimation, [34] human action recognition, [19, 35] human tracking, [36, 37] and so forth. However, their problem definitions are different with phone-related pedestrian distracted behaviour recognition. The only related benchmark dataset (not publicised) for this research problem is by, [4, 5] however their dataset contains only several thousand images, which is not large enough for deep learning based methods. Furthermore, [4, 5] use one single image as input, so the synchronised information from different cameras are missed in their dataset. Based on these observations, we propose a new large benchmark dataset for this research problem. The proposed benchmark dataset in this article contains 448 synchronised video pairs of 53,760 images on an intelligent vehicle in Texas. To the best of our knowledge, the proposed benchmark dataset is the largest dataset for this research problem and the first dataset including multiple synchronised cameras for this research problem.

# 3 | METHODS

As we discussed above, we define three classes for the phone-related pedestrian distracted behaviours: Class 1: No Engagement, Class 2: Eye Engagement, Class 3: Phone Call Engagement. Class 1 means that the pedestrian exhibits no distracted behaviour. Class 2 means that the pedestrian is distracted by eye engagement with phone (texting, video watching, map viewing, game playing, etc.). Class 3 means that the pedestrian is distracted by phone call. Our research goal is to accurately detect these three classes of phone-related pedestrian distracted behaviours.

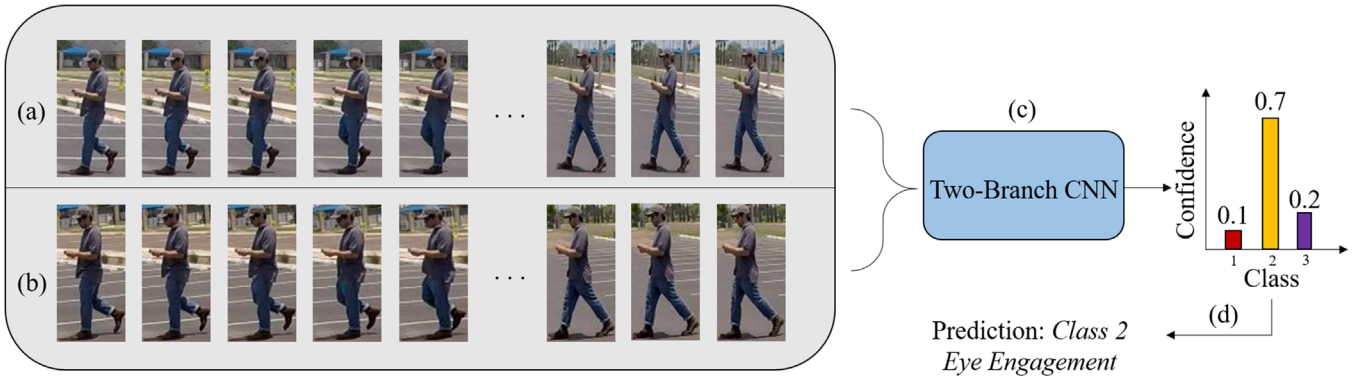


**FIGURE 3** The proposed end-to-end deep learning framework: the synchronised image pairs of (a) left camera and (b) right camera are the inputs into the (c) Two-branch CNN, followed by (d) voting based confidence score analysis. Note that the overall proposed framework's input is a synchronised video pair, that is, a video from the left camera and a synchronised video from the right camera

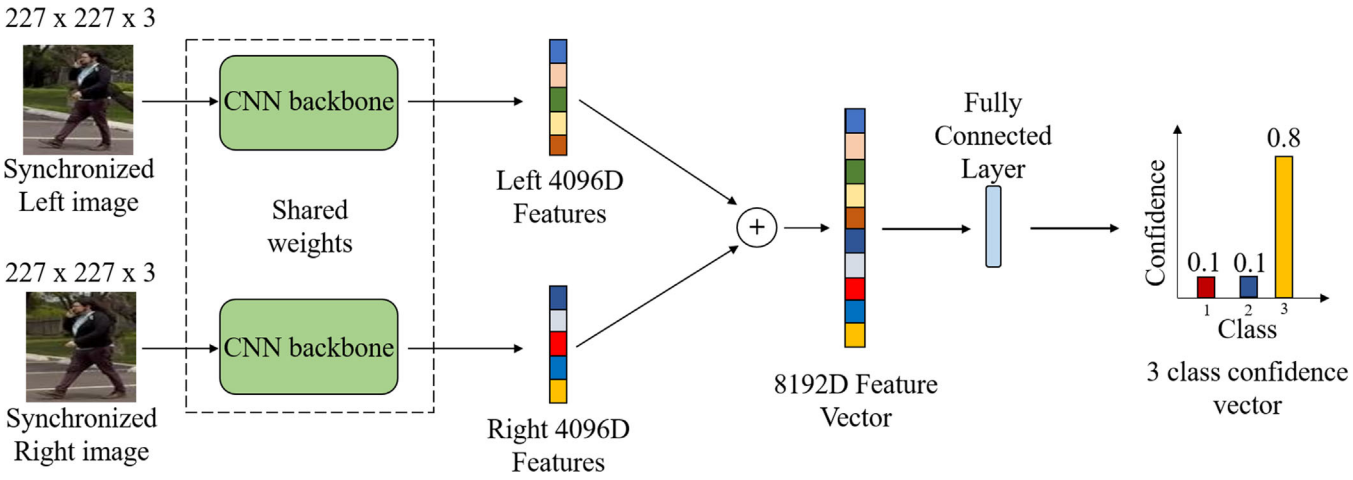


**FIGURE 4** The detailed network structure of the proposed Two-Branch CNN. Note that the proposed Two-Branch CNN's input is a synchronised image pair, that is, an image from the left camera and a synchronised image from the right camera

## 3.1 | System design and problem definition

On an intelligent or autonomous vehicle, we install two GoPro cameras in the front of the vehicle. One GoPro camera is installed on the left side and another GoPro camera is installed on the right side of the windshield on the vehicle. Nowadays, deep learning based CNN methods, like Faster R-CNN, [38] YOLO [39] and Mask R-CNN, [40] have shown accurate and efficient performance in human detection, so it is easy to extract the location of pedestrians in the image. Taking one synchronised image pair of the same pedestrian by the two front on-car GoPro cameras as the inputs, defined as synchronised left image and synchronised right image, we propose a Two-branch CNN to learn to output the three classes of phone-related pedestrian distracted behaviours. Because the classification on one single image pair is not accurate enough, we propose a voting based confidence score analysis that uses a short video (dozens of images) for the final classification as shown in Figure 3.

## 3.2 | Two-branch CNN framework

The two-branch framework is distinct from a standard CNN that takes one single image as input. In this article, our input is an image pair of the same pedestrian: synchronised left image by the left camera and synchronised right image by the right camera. Therefore, our input is actually two images. Because the camera parameters (angle, location and scale) are fixed, we expect the two-branch CNN to capture the camera-related features from two perspectives. The proposed Two-Branch CNN will extract features for each camera by a CNN backbone, concatenate the extracted features and finally perform a robust classification, as displayed in Figure 4. This is the reason for using two branches in the proposed method. The network weights are shared for the two branches, but they have different inputs so the two-branch CNN could fit to learn features from the left and right cameras. Sharing weights could reduce the network parameter size but also keep the idea of inputting left and right camera images. The term "Two-branch CNN" is widely used in

the area of computer vision and deep learning to represent the feature learning of two different inputs even when the weights are shared, for example, similar to that in [41]. We will introduce the details of each key component in the following sections.

### 3.3 | CNN backbone for feature extraction

The CNN backbone is used for feature extraction. Many CNN architectures can be used as the backbone for feature extraction such as AlexNet, [15] VGG, [16] ResNet, [17] GoogLeNet, [42] and DenseNet [18] because of their discriminative feature representation of images. To reduce the network parameters and keep the consistent feature extraction of two cameras, we use shared weights of the CNN backbone for the two branches. In our research, we use.

The proposed method is compatible to use with most of the existing deep Convolutional Neural Networks as the CNN backbone, such as AlexNet (8 layers), VGG16 (16 layers), GoogLeNet (22 layers), ResNet (152 layers) and so on. Since there are many possible CNN backbones, it is hard to show the experiments using every existing CNN network as the backbone. To show the feasibility of the proposed method and make the experiments consistent, we use AlexNet [15] or VGG16 [16] as the CNN backbone for our algorithm development and experiments on the two-branch model. In our preliminary experiments, we used ResNet as the CNN backbone before, it could improve the performance a little bit. However, ResNet needs much more time to train the proposed Two-branch CNN on our collected large dataset because ResNet (152 layers) has much more parameters and needs more GPU memory. Due to the much longer training time and higher GPU memory requirement of other CNN models, we decided to use a modified AlexNet or modified VGG16 as the CNN backbone for the experiments in this article. Both AlexNet and VGG16 are pretrained on the large ImageNet dataset [15] and are put through additional transfer learning on our training data.

#### 3.3.1 | Modified AlexNet

The input size is 227×227×3 (a colour image with size 227x227). The original AlexNet architecture includes five convolutional layers, used for creating a feature map that indicates locations and strengths of detected features in the input, three max pooling layers for down sampling the data size and three fully-connected layers which hold the composite information from all layers before it. The final three fully connected layers of the original AlexNet are removed to extract features. The extracted feature of this CNN backbone based on the modified AlexNet is one 4096-dimensional vector.

#### 3.3.2 | Modified VGG16

The standard VGG16 architecture is an accurate image classifier that is considered deeper than AlexNet. The VGG16 architecture has 13 convolutional layers and 5 max pooling layers and 3 fully-connected layers. Just as we did to AlexNet, the final 3 fully-connected layers of the original VGG16 are removed to extract features. The extracted feature of this CNN backbone based on the modified VGG16 is also one 4096-Dimensional vector.

### 3.4 | Feature concatenation

The extracted feature vectors (4096D) of the left and right images passed through the CNN backbone respectively are concatenated to create the final feature vector, that is, one 8192-Dimensional vector. This final concatenated feature vector is then passed into an extra fully-connected layer to reduce the output dimension to one 3-Dimensional vector which is related to the 3 classes of phone-related pedestrian distracted behaviours.

### 3.5 | Classification

The output of the network is a value that represents how confident the network is that it belongs to this particular class. The larger the value, the more confident the network is in that classification. To normalise these values, the softmax function is applied to the input (synchronised image pair) $x$ in Equation (1).

$$F(x_i) = \frac{e^{x_i}}{\sum_{i=1}^{k} e^{x_i}}, \tag{1}$$

where $k$ represents the number of classes, and $i = 1, 2, \ldots, k$. Here, $x_i$ represents the output value of class $i$ for the input $x$ and $F(x_i)$ is the predicted confidence for class $i$ of the input $x$. In this research, $k = 3$. This softmax function will ensure that the confidences returned are normalised to the range between 0 and 1 with the summation as 1. The confidence scores seen in Figure 4 have been normalised. The classification of the input (synchronised image pair) $x$ will therefore be the class with the highest normalised confidence as defined in Equation (2).

$$P_x = \max(F(x_i)), \tag{2}$$

where $i = 1, 2, \ldots, k$.

### 3.6 | Loss function for training

We use cross-entropy loss to train the proposed Two-Branch CNN network. We have ground truth labels for our training data. The ground truth classification label $y$ for the input $x$ is a one-hot $k$-Dimensional vector, where $y_i = 1$ if the ground truth class is $i$. The detailed cross-entropy loss is defined in Equation (3).

$$L(y, x) = -\sum_{i=1}^{k} y_i \log(P_x). \tag{3}$$

Because $P_x$ is a confidence between 0 and 1, minimising this loss function will encourage the larger confidence of $P_x$ for the correct class in the CNN training. Cross-entropy loss is a widely used loss function in deep learning, which can be efficiently minimised by gradient decent and back propagation.

### 3.7 | From image recognition to video recognition

After training the CNN, it can be directly applied to test new inputs. During testing, the class with highest confidence is given as the prediction for the input. This is the image-level recognition taking synchronised image pair as input, which can be easily extended to the video-level recognition taking synchronised video pair as input. One short video contains dozens of images. We use a voting based confidence score analysis to determine the video-level recognition.

#### 3.7.1 | Voting based confidence score analysis for video-level recognition

Let us define $F(x_{i,j})$ as the predicted confidence for the class $i$ of the $j$-th input (synchronised image pair). $N$ represents the number of images in a short video and $N$ is equal to or less than 60 in our experiment, so $j$ is from 1 to $N$. Once all the images in a video are given a confidence score, the average confidence for each class is computed by Equation (4):

$$Z_i = \frac{\sum_{j=1}^{N} F(x_{i,j})}{N}, \tag{4}$$

where $Z_i$ represents the averaged confidence score of class $i$ (where $i = 1, 2, \ldots, k$) for one video input. Once the average of each class is found, the class with the highest averaged confidence score is assigned to that video input. Inspired by the idea of sparse coding that favours one-hot prediction, we make an exception when all average confidence scores are within a similar range (standard deviation is smaller than a threshold $T = 0.2$) to predict the video input as Class 1 (No Engagement).

## 4 | EXPERIMENTS

In this section, we will introduce the collision study, collected benchmark dataset, experimental setting and results.

### 4.1 | Collision study

First of all, we conducted a simple collision study to better understand the severity of a distraction itself. In this experiment, five pedestrian participants were moving from one predetermined starting point to another predetermined ending point

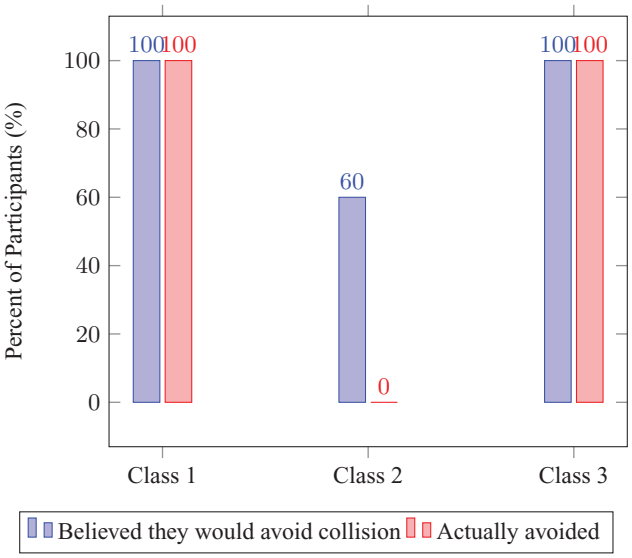


**FIGURE 5** Result of the collision study on five pedestrian participants. It significantly shows that Class 2 (eye engagement) is a serious distraction for traffic safety. Note that classes 1, 2, 3 indicate No Engagement, Eye Engagement, Phone Call Engagement, respectively

(about 100 meters) by performing the behaviours of the three classes described above. In the motion path of each participant, we randomly put a stationary obstacle object (e.g. a chair), whose location is unknown to the participant. Each participant was asked before completing the motions if they believed they would be able to avoid obstacles placed in their path secretively.

Among the responses, 100% believed they would be able to avoid obstacles when they were free walking (class 1: no engagement) and talking on the phone (class 3: phone call engagement), while 60% believed they would be able to while viewing their phone (class 2: eye engagement). We found that each participant was able to avoid all possible collisions laid before them when completing a motion that fits into classes 1 and 3. When performing a motion that fits class 2 however, each participant had at least 1 collision. These results can be seen on Figure 5. It significantly shows that class 2 (eye engagement) is a serious distraction for traffic safety. Also, 100% of the five participants agreed that class 3 might distract the pedestrian more in complex urban environments. This collision study shows that it is important to detect the phone-related pedestrian distracted behaviours to improve the traffic safety.

### 4.2 | PPDB benchmark dataset

Because there are no publicised datasets for our research problem, we collected our own custom dataset in the campus of University of Texas-Rio Grande Valley (UTRGV) on multiple days between the times of 11:00 am and 4:00 pm. The data was gathered using two GoPro cameras mounted on the front windshield of an intelligent vehicle. One camera is installed on the left side and another one is on the right side. The two cameras recorded video at 720p resolution and 30 frames per


**FIGURE 6** Synchronised videos by the Left and Right cameras (60 frames for each video); (a) Class 1: No Engagement, (b) Class 2: Eye Engagement, (c) Class 3: Phone Call Engagement

second. The cameras were angled to be front facing and were placed at equal distances from the edge of the windshield, approximately 8 inches or 20.32 centimetres. The resulting view provides a nearly 180-degree view ahead of the vehicle using the two perspectives. Therefore, even when the pedestrian is on the road curb, both cameras could clearly see that pedestrian. The gathered data is staged with 8 different participants acting as pedestrians, each completing all staged motions within 1 hour. The activity of each participant belongs to one of three classes. Each activity was performed by passing nearly perpendicular to the front of the vehicle at different distances. It is worth mentioning that the pedestrians in most images captured in the benchmark dataset are not exactly in front of the vehicle but moving nearby the intelligent vehicle to simulate the road crossing. Within each class, different activities were captured such as texting with the left hand or right hand or both hands, walking with or without phones in hand. In total, each participant performed 10 different activities twice: from left to right and from right to left. We only kept the synchronised left and right images/videos of the same pedestrians and discarded the non-pedestrian areas in the collected images. Each image is resized to a height of 128 pixels and a width of 64 pixels. Each video has 60 images/frames, which is just 2 s. The example images/videos are displayed in Figure 6.

In total, we collected 448 synchronised video pairs of 53,760 images. 24% of the dataset was used for testing (12,960 images or 6480 pairs) and the rest were used for training (40,800 images or 20,400 pairs). There are 340 video pairs for training and 108 video pairs for testing. Among the 340 training video pairs, 190 of them are captured while the vehicle remains static and the remained are captured while the vehicle is in motion (dynamically). There is no overlap between participants in training and testing. The details about the PPDB benchmark are summarised in Table 1.

## 4.3 | Experimental setting

The proposed network is implemented by PyTorch and our training and testing were completed on a workstation with 4.0 GHz CPU and 16GB memory with a NVIDIA TITAN Xp GPU. We used the pre-trained AlexNet or VGG16 model to initialise the CNN backbone and randomly initialise other layers of our model, and then we fine-tune the initialised model using our training set with the following setting: batch size of 1, an initial learning rate of 0.0001, and a momentum of 0.9. Stochastic gradient descent (SGD) is used for optimisation during the back propagation and the training epoch is 20. The validation set is identical to the training set.

### 4.3.1 | Comparison methods

We implement several baseline methods as comparisons. For all the comparison methods based on AlexNet [15] and VGG16, [16] we modify the corresponding final fully-connected layer to be adaptive to output three classes. Several baseline methods are defined for this research in the following: **AlexNet**$_{\text{Left}}$: an AlexNet network pre-trained on ImageNet using only the Left camera's data of PPDB training set for training/fine-tuning; **AlexNet**$_{\text{Right}}$: an AlexNet network pre-trained on ImageNet using only the Right camera's data of PPDB training set for training/fine-tuning; **VGG16**$_{\text{Left}}$: a VGG16 network pre-trained on ImageNet using only the Left camera's data of PPDB training set for training/fine-tuning; **VGG16**$_{\text{Right}}$: a VGG16 network pre-trained on ImageNet using only the Right camera's data of PPDB training set for training/fine-tuning. **AlexNet**$_{\text{All}}$: an AlexNet network pre-trained on ImageNet using all the PPDB training set for training/fine-tuning. For synchronised image pair, predict the Left image and Right image by the trained AlexNet$_{\text{All}}$ independently and then pick the class with the highest confidence; **VGG16**$_{\text{All}}$: a VGG16 network pre-trained on ImageNet using all the PPDB training set for training/fine-tuning. For synchronised image pair, predict the Left image and Right image by the trained VGG16$_{\text{All}}$ independently and then pick the class with the highest confidence. **AlexNet**$_{\text{Joined}}$: for synchronised image pair, predict the Left image by the trained AlexNet$_{\text{Left}}$ and predict the Right image by the trained AlexNet$_{\text{Right}}$. We pick the class with highest confidence by AlexNet$_{\text{Left}}$ and AlexNet$_{\text{Right}}$ as the result for the synchronised image pair; **VGG16**$_{\text{Joined}}$: for synchronised image pair, predict the Left image by the trained VGG16$_{\text{Left}}$ and predict the Right image by the trained VGG16$_{\text{Right}}$. We pick the class with highest confidence by VGG16$_{\text{Left}}$ and VGG16$_{\text{Right}}$ as the result for the synchronised image pair; **Proposed**$_{\text{AlexNet}}$: proposed

**TABLE 1** Summary of PPDB benchmark dataset

| Set | Class | No. of synchronised video pairs | No. of videos | No. of synchronised image pairs | No. of frames |
|---|---|---|---|---|---|
| Training | 1 | 110 | 220 | 6600 | 13200 |
| | 2 | 120 | 240 | 7200 | 14400 |
| | 3 | 110 | 220 | 6600 | 13200 |
| | **Total:** | **340** | **680** | **20,400** | **40800** |
| Testing | 1 | 38 | 76 | 2280 | 4560 |
| | 2 | 40 | 80 | 2400 | 4800 |
| | 3 | 30 | 60 | 1800 | 3600 |
| | **Total:** | **108** | **216** | **6480** | **12960** |

**TABLE 2** Results for Task 1: single image classification on the PPDB Benchmark. Note that classes 1, 2, 3 indicate No Engagement, Eye Engagement, Phone Call Engagement, respectively

| Baseline methods | Tested on | Class | Accuracy | Baseline methods | Tested on | Class | Accuracy |
|---|---|---|---|---|---|---|---|
| $AlexNet_{Left}$ | Left images | 1 | 36.9% | $VGG16_{Left}$ | Left images | 1 | 51.4% |
| | | 2 | 61.3% | | | 2 | 76.5% |
| | | 3 | 65.7% | | | 3 | 54.1% |
| | | Average: | 54.6% | | | Average: | 60.7% |
| | Right images | 1 | 45.1% | | Right images | 1 | 71.4% |
| | | 2 | 68.7% | | | 2 | 77.7% |
| | | 3 | 53.9% | | | 3 | 38.5% |
| | | Average: | **55.9%** | | | Average: | **62.5%** |
| $AlexNet_{Right}$ | Left images | 1 | 11.2% | $VGG16_{Right}$ | Left images | 1 | 75.4% |
| | | 2 | 63.3% | | | 2 | 81.9% |
| | | 3 | 68.2% | | | 3 | 58.1% |
| | | Average: | 47.6% | | | Average: | **71.8%** |
| | Right images | 1 | 34.5% | | Right images | 1 | 73.5% |
| | | 2 | 74.3% | | | 2 | 75.6% |
| | | 3 | 52.3% | | | 3 | 50.4% |
| | | Average: | **53.7%** | | | Average: | 66.5% |
| $AlexNet_{All}$ | Left images | 1 | 55.0% | $VGG16_{All}$ | Left images | 1 | 80.1% |
| | | 2 | 82.8% | | | 2 | 90.0% |
| | | 3 | 44.2% | | | 3 | 55.8% |
| | | Average: | 60.7% | | | Average: | **75.3%** |
| | Right images | 1 | 56.2% | | Right images | 1 | 83.7% |
| | | 2 | 87.5% | | | 2 | 88.8% |
| | | 3 | 41.5% | | | 3 | 43.7% |
| | | Average: | **61.7%** | | | Average: | 72.0% |

Two-Branch CNN using AlexNet as the CNN backbone; **Proposed**$_{VGG16}$: proposed Two-Branch CNN using VGG16 as the CNN backbone.

We evaluate three tasks for this research problem. We could make the classification based on one single image (Task 1), one synchronised image pair (Task 2), and one synchronised video pair (Task 3). Task 3, as an extension of Task 2, uses a voting algorithm to deal with multiple images of the video pair. To make a comprehensive study, we evaluate the performance in these three different settings. Task 1: Single image classification by $AlexNet_{Left}$, $AlexNet_{Right}$, $VGG16_{Left}$, $VGG16_{Right}$, $AlexNet_{All}$ and $VGG16_{All}$. Task 2: Synchronised image-pair classification by $AlexNet_{All}$, $AlexNet_{Joined}$, $VGG16_{All}$, $VGG16_{Joined}$ and the proposed

**TABLE 3** Results for Task 2: synchronised image pair classification on the PPDB Benchmark. Note that classes 1, 2, 3 indicate No Engagement, Eye Engagement, Phone Call Engagement, respectively

| Method | Class | Accuracy | Method | Class | Accuracy |
|---|---|---|---|---|---|
| $AlexNet_{Joined}$ | 1 | 33.4% | $VGG16_{Joined}$ | 1 | 68.2% |
| | 2 | 72.1% | | 2 | 81.6% |
| | 3 | 60.2% | | 3 | 59.1% |
| | Average: | 55.2% | | Average: | 69.6% |
| $AlexNet_{All}$ | 1 | 55.6% | $VGG16_{All}$ | 1 | 81.9% |
| | 2 | 85.1% | | 2 | 89.4% |
| | 3 | 42.9% | | 3 | 49.7% |
| | Average: | 61.2% | | Average: | 73.7% |
| $Proposed_{AlexNet}$ | 1 | 42.7% | $Proposed_{VGG16}$ | 1 | 67.9% |
| | 2 | 76.8% | | 2 | 86.5% |
| | 3 | 82.5% | | 3 | 83.4% |
| | Average: | **67.3%** | | Average: | **79.3%** |

**TABLE 4** Results of Task 3: Synchronised video pair classification on the PPDB Benchmark. The best performance is shown in the red colour. Note that classes 1, 2, 3 indicate No Engagement, Eye Engagement, Phone Call Engagement, respectively

| Name of network | Class | Accuracy | Name of network | Class | Accuracy |
|---|---|---|---|---|---|
| $AlexNet_{Joined}$ | 1 | 34.2% | $VGG16_{Joined}$ | 1 | 68.4% |
| | 2 | 70.0% | | 2 | 80.0% |
| | 3 | 63.3% | | 3 | 53.3% |
| | Average: | 55.8% | | Average: | 67.2% |
| $AlexNet_{All}$ | 1 | 65.8% | $VGG16_{All}$ | 1 | 84.2% |
| | 2 | 95.0% | | 2 | 97.5% |
| | 3 | 43.3% | | 3 | 60.0% |
| | Average: | 68.0% | | Average: | 80.6% |
| $Proposed_{AlexNet}$ | 1 | 50.0% | $Proposed_{VGG16}$ | 1 | 71.1% |
| | 2 | 82.5% | | 2 | 95.0% |
| | 3 | 83.3% | | 3 | 86.7% |
| | Average: | **71.9%** | | Average: | **84.3%** |

method. Task 3: Synchronised video-pair classification by $AlexNet_{All}$, $AlexNet_{Joined}$, $VGG16_{All}$, $VGG16_{Joined}$ and the proposed method. Each of the baseline methods $AlexNet_{All}$, $AlexNet_{Joined}$, $VGG16_{All}$, and $VGG16_{Joined}$ can also be extended to synchronised video-pair classification following the same voting based confidence score analysis for video-level recognition by Equation (4) and the standard deviation check same as the proposed method. We use the classification accuracy of each of three classes and their average classification accuracy to evaluate the trained model.

## 4.4 | Experimental results

For the Task 1: Single image classification, the results are shown in Table 2. VGG16 result is better than AlexNet result because VGG16 has more convolution layers in CNN structure, leading to more advanced performance. In most cases, training a CNN model on the training data of one camera might not be consistent on the testing data of another camera. This is because of the view-angle difference between the Left camera and the Right camera. Note that the proposed method needs a synchronised image pair as input, so the proposed method cannot be evaluated in Task 1. We can clearly see that using one single camera for phone-related pedestrian distracted behaviour classification is not robust enough (75.3% for the best average classification accuracy). Intuitively, the model trained on one single camera should perform better on the same camera, but $VGG16_{Left}$ performed better on the Right images (62.5%) than the Left images (60.7%), and $AlexNet_{Left}$ performed better on the Right images (55.9%) than the Left images (54.6%). Although the performance gap is very small, this result might be confusing. The possible reason is the inconsistency of the human detector. Because we first detect pedestrians and then crop the image patch and then resize it to a uniform size (64-pixel width and 128-pixel height), the bounding box of the detected pedestrian has some location and scale variances, leading to the above confusion. However, the proposed Two-branch CNN method does not have this problem, which uses the synchronised image pair as the input.

For the Task 2: Synchronised image pair classification, the results are shown in Table 3. We can see that $Proposed_{VGG16}$ obtained 79.3% for the best average classification accuracy. In the experiments of Task 2, the proposed Two-Branch CNN with AlexNet as backbone is better than the baseline methods using AlexNet, and the proposed Two-Branch CNN with VGG16 as backbone is better than the baseline methods using VGG16. Compared to Task 1, we find that using two synchronised cameras for phone-related pedestrian distracted behaviour classification is better than using one single camera.

For the Task 3: Synchronised video-pair classification, the results are shown in Table 4. We can see that $Proposed_{VGG16}$ obtained 84.3% for the best average classification accuracy: 71.1%, 95.0% and 86.7% for Class 1, Class 2 and Class 3 respectively. In the experiments of Task 3, the proposed Two-Branch CNN with AlexNet as backbone is better than the baseline methods using AlexNet, and the proposed Two-Branch CNN with VGG16 as backbone is better than the baseline methods using VGG16. Compared to Task 2, we find that using a short synchronised video pair for phone-related pedestrian distracted behaviour classification can be more accurate and robust than using one synchronised image pair. In order to better show the comprehensive result, we also display the confusion matrices of the proposed methods for the Task 3 in Figure 7. From the confusion matrices, we can see that $Proposed_{VGG16}$ obtains higher classification accuracy than $Proposed_{AlexNet}$. Taking $Proposed_{VGG16}$ as the example, 95% of Class 2 sequences are correctly classified, while 26.3% of Class 1 sequences are misclassified as Class 3, while 13.3%

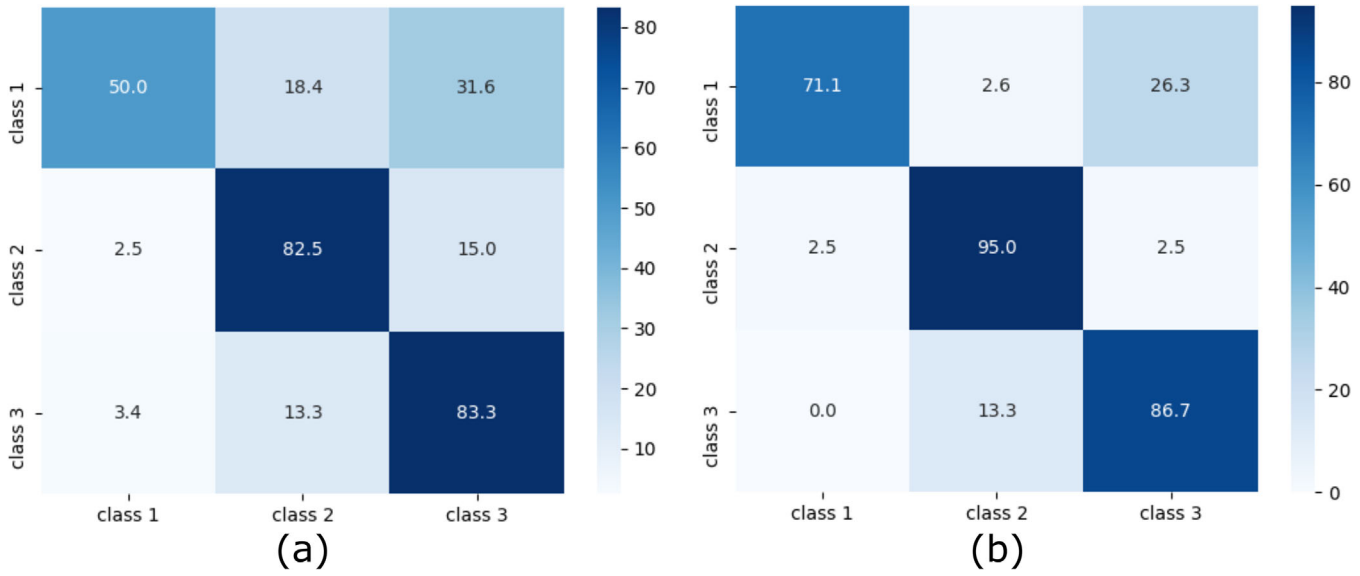


**FIGURE 7** Confusion matrices of the proposed methods on the final synchronised video pair based classification by (a) $\text{Proposed}_{\text{AlexNet}}$, (b) $\text{Proposed}_{\text{VGG16}}$. Note that classes 1, 2, 3 indicate No Engagement, Eye Engagement, Phone Call Engagement, respectively

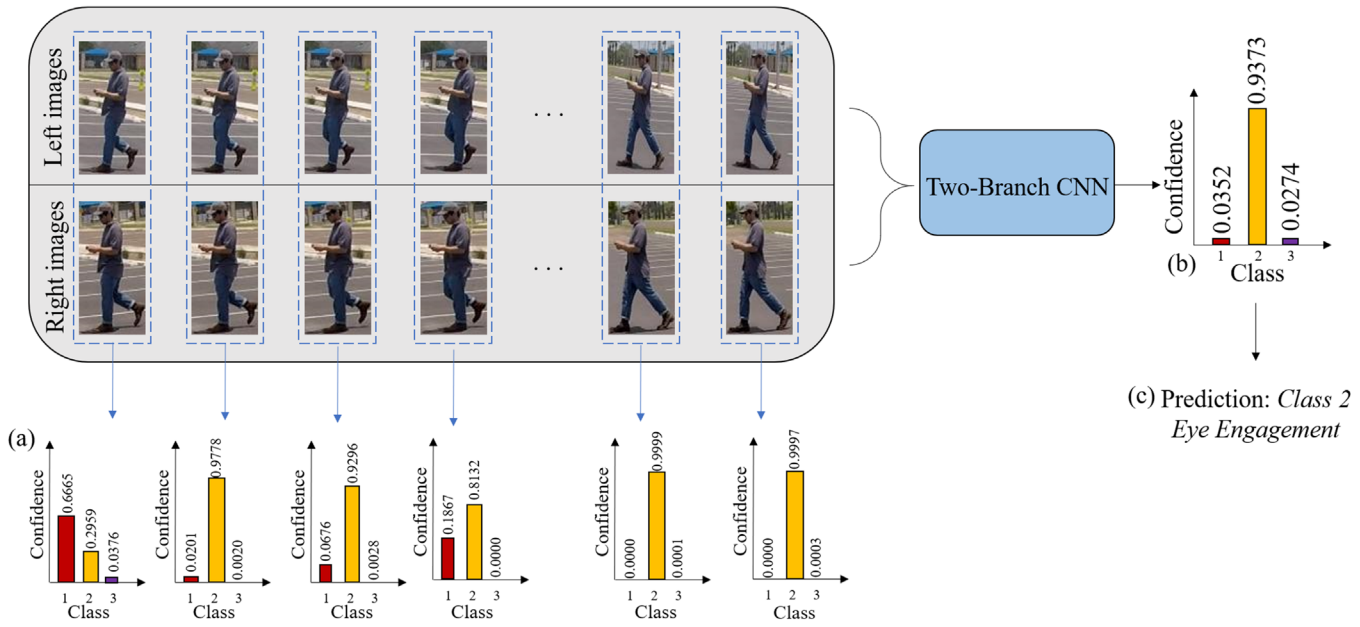


**FIGURE 8** Recognition for one example synchronised video pair of PPDB Benchmark

of Class 3 sequences are misclassified as Class 2. The overall performance is good and reasonable. The Class 1 (no engagement) sometimes has visually similar pose to Class 3 (phone call engagement), so the similar pose leads to this recognition difficulty to Class 1. This is the reason why Class 1 has the lowest accuracy (71.1%) for $\text{Proposed}_{\text{VGG16}}$.

## 5 | DISCUSSION

In Figure 8, we show the actual classification for one synchronised video pair by the proposed Two-Branch CNN with VGG16 as backbone. The majority of image pairs are correctly classified as Class 2: Eye Engagement. If we only consider one single synchronised image pair as input, like the first image pair in Figure 8, it might not be accurate enough with some biases. Therefore, the voting based confidence score analysis in the image sequence is able to improve video-level recognition.

The Figure 9 shows the resulting average video-level classification accuracy of voting based confidence score analysis when only the first $N$ images are considered in Equation (4). For example, when $N = 10$, only the first 10 image/frame pairs are used for the voting based confidence score analysis. We

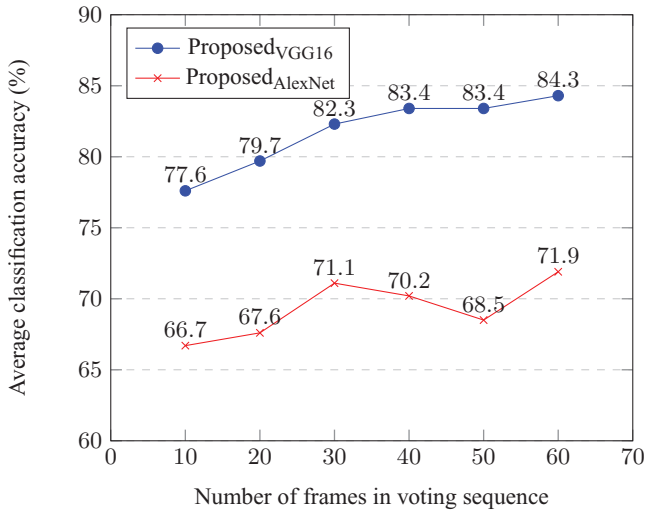


**FIGURE 9** Average video-level classification accuracy of the proposed method (using VGG16 and AlexNet as the backbone) with different images for voting on the PPDB Benchmark

find that using more images lead to better average classification accuracy.

## 6 | CONCLUSIONS

In this article, we propose the first end-to-end deep learning framework (Two-Branch CNN) to detect the phone-related distracted behaviours for intelligent and autonomous vehicles. We also propose and will publicise a new large dataset of PPDB Benchmark for this research problem. In the experiments, we find that two synchronised cameras could obtain better performance than using one single camera, and using a short synchronised video pair could achieve improved results than using single synchronised image pair. The proposed method finally gets 84.3% as the average classification accuracy. In the future, we will involve human pose estimation and spatio-temporal analysis into the proposed Two-Branch CNN to improve the recognition.

## ACKNOWLEDGEMENTS

This work is supported by NVIDIA GPU Grant, Dwight David Eisenhower Transportation Fellowship Program, and Amazon Web Services (AWS) Cloud Credits for Research Award.

## REFERENCES

1. 2017 Fatal motor vehicle crashes: overview, National Highway Traffic Safety Administration, US Department of Transportation (2017). https://crashstats.nhtsa.dot.gov/Api/Public/ViewPublication/812603. Accessed 25 July 2020.
2. Texas motor vehicle crash statistics 2019, Texas Department of Transportation (2019). https://www.txdot.gov/government/enforcement/annual-summary.html. Accessed 25 July 2020
3. Traffic safety facts 2018, National Highway Traffic Safety Administration, US Department of Transportation (2018). https://crashstats.nhtsa.dot.gov/Api/Public/ViewPublication/812850. Accessed 25 July 2020
4. Rangesh, A., et al.: Pedestrians and their phones-detecting phone-based activities of pedestrians for autonomous vehicles. In: IEEE International Conference on Intelligent Transportation Systems (ITSC), Rio de Janeiro, November 2016, pp. 1882–1887. IEEE, Piscataway (2016)
5. Rangesh, A., Trivedi, M.M.: When vehicles see pedestrians with phones: A multicue framework for recognizing phone-based activities of pedestrians. IEEE Trans. Intell. Veh. 3(2), 218–227 (2018)
6. Texas Strategic Highway Safety Plan, T.A.T. Institute (2019). https://www.texasshsp.com/emphasis-areas/distracted-driving/. Accessed 25 July 2020
7. Strayer, D.L., Drew, F.A.: Profiles in driver distraction: Effects of cell phone conversations on younger and older drivers. Hum. Factors 46(4), 640–649 (2004)
8. Liang, Y., et al.: Real-time detection of driver cognitive distraction using support vector machines. IEEE Transactions on Intelligent Transportation Systems 8(2), 340–350 (2007)
9. Pettitt, M., et al.: Defining driver distraction. In: 12th World Congress on Intelligent Transport Systems, San Francisco, 6–10 November 2005
10. Brodsky, W.: A performance analysis of in-car music engagement as an indication of driver distraction and risk. Transp. Res. Part F: Traffic Psychol. Behav. 55, 210–218 (2018)
11. Przybyla, J., et al.: Estimating risk effects of driving distraction: A dynamic errorable car-following model. Transp. Res. Part C: Emerging Technol. 50, 117–129 (2015)
12. Przybyla, J., Zhou, X.: Cell phone use while driving: a literature review and recommendations. ResearchGate 13, 1–25 (2008)
13. Russo, B.J., et al.: Pedestrian behavior at signalized intersection crosswalks: observational study of factors associated with distracted walking, pedestrian violations, and walking speed. Transp. Res. Rec. 2672(35), 1–12 (2018)
14. Mwakalonge, J., et al.: Distracted walking: examining the extent to pedestrian safety problems. J. Traffic Transp. Eng. 2(5), 327–337 (2015)
15. Krizhevsky, A., et al.: Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems, Conference on Neural Information Processing Systems, pp. 1097–1105 (2012)
16. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
17. He, K., et al.: Deep residual learning for image recognition. In: IEEE conference on Computer Vision and Pattern Recognition, Las Vegas, June 2016, pp. 770–778. IEEE, Piscataway (2016)
18. Huang, G., et al.: Densely connected convolutional networks. In: IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, June 2016, pp. 4700–4708. IEEE, Piscataway (2017)
19. Zheng, K., et al.: Learning view-invariant features for person identification in temporally synchronized videos taken by wearable cameras. In: IEEE International Conference on Computer Vision, Honolulu, November 2017, pp. 2858–2866. IEEE, Piscataway (2017)
20. Lin, Y., et al.: Co-interest person detection from multiple wearable camera videos. In: IEEE International Conference on Computer Vision, Santiago, December 2015, pp. 4426–4434. IEEE, Piscataway (2015)
21. Fu, H., et al.: Cluster-based co-saliency detection. IEEE Trans. Image Process. 22(10), 3766–3778 (2013)
22. Thompson, L.L., et al.: Impact of social and technological distraction on pedestrian crossing behaviour: an observational study. Inj. Prev. 19(4), 232–237 (2013)
23. Le, B., et al.: Determining the incidence of distraction among trauma patients in all modes of transportation. J. Trauma Acute Care Surg. 87(1), 87–91 (2019)
24. Wang, T., et al.: WalkSafe: a pedestrian safety app for mobile phone users who walk and talk while crossing roads. In: 12th Workshop on Mobile Computing Systems & Applications, Vol. 5. ACM, New York (2012)
25. Vinayaga-Sureshkanth, N., et al.: A practical framework for preventing distracted pedestrian-related incidents using wrist wearables. IEEE Access 6, 78016–78030 (2018)
26. Zaki, M.H., Sayed, T.: Exploring walking gait features for the automated recognition of distracted pedestrians. IET Intel. Transport Syst. 10(2), 106–113 (2016)

27. Schulz, A.T., Stiefelhagen, R.: Pedestrian intention recognition using latent-dynamic conditional random fields. In: IEEE Intelligent Vehicles Symposium (IV), Seoul, August 2015, pp. 622–627. IEEE, Piscataway (2015)
28. Keller, C.G., Gavrila, D.M.: Will the pedestrian cross? A study on pedestrian path prediction. IEEE Trans. Intell. Transp. Syst. 15(2), 494–506 (2013)
29. Kooij, J.F.P., et al.: Context-based pedestrian path prediction. In: European Conference on Computer Vision, Zurich, September 2014, pp. 618–633. Springer, New York (2014)
30. Kataoka, H., et al.: Fine-grained walking activity recognition via driving recorder dataset. In: IEEE 18th International Conference on Intelligent Transportation Systems, Las Palmas, November 2015, pp. 620–625. IEEE, Piscataway (2015)
31. Quintero, R., et al.: Pedestrian path prediction based on body language and action classification. In: 17th International IEEE Conference on Intelligent Transportation Systems (ITSC), Qingdao, November 2014, pp. 679–684. IEEE, Piscataway (2014)
32. Choi, W., et al.: What are they doing?: Collective activity classification using spatio-temporal relationship among people. In: 2009 IEEE 12th International Conference on Computer Vision Workshops, Kyoto, May 2009, pp. 1282–1289. IEEE, Piscataway (2009)
33. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: IEEE conference on Computer Vision and Pattern Recognition, San Diego, June 2005, pp. 886–893. IEEE, Piscataway (2005)
34. Andriluka, M., et al.: 2D human pose estimation: New benchmark and state of the art analysis. In: IEEE Conference on computer Vision and Pattern Recognition, Columbus, September 2014, pp. 3686–3693. IEEE, Piscataway (2014)
35. Soomro, K., et al.: UCF101: A dataset of 101 human actions classes from videos in the wild. arXiv preprint arXiv:1212.0402 (2012)
36. Leal-Taixé, L., et al.: Motchallenge 2015: Towards a benchmark for multi-target tracking. arXiv preprint arXiv:1504.01942 (2015)
37. Yu, H., et al.: Multiple human tracking in wearable camera videos with informationless intervals. Pattern Recognit. Lett. 112, 104–110 (2018)
38. Ren, S., et al.: Faster r-cnn: Towards real-time object detection with region proposal networks. In: Advances in Neural Information Processing Systems, Conference on Neural Information Processing Systems, pp. 91–99 (2015)
39. Redmon, J., Farhadi, A.: YOLO9000: better, faster, stronger. In: IEEE conference on Computer Vision and Pattern Recognition, Honolulu, November 2017, pp. 7263–7271. IEEE, Piscataway (2017)
40. He, K., et al.: Mask R-CNN. In: IEEE International Conference on Computer Vision, Honolulu, November 2017, pp. 2961–2969. IEEE, Piscataway (2017)
41. Guo, H., et al.: Visual attention consistency under image transforms for multi-label image classification. In: IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, June 2019, pp. 729–739. IEEE, Piscataway (2019)
42. Szegedy, C., et al.: Going deeper with convolutions. In: IEEE conference on computer vision and pattern recognition, Boston, June 2015, pp. 1–9. IEEE, Piscataway (2015)

**How to cite this article:** Saenz H, Sun H, Wu L, Zhou X, Yu H. Detecting phone-related pedestrian distracted behaviours via a two-branch convolutional neural network. *IET Intell Transp Sy*. 2020;1–12. https://doi.org/10.1049/itr2.12012