

Cost-Effective Smartphone-Camera-Based Scheme to Distinguish Drivers from Passengers for Texting-While-Driving*

SHIH-YU HUANG AND CHIA-HUI WANG⁺

Department of Computer Science and Information Engineering

Ming Chuan University

Taoyuan, 333 Taiwan

E-mail: {syhuang; wangch⁺}@mail.mcu.edu.tw

The method proposed in this paper utilizes the front and rear cameras of a user's smartphone to distinguish drivers from passengers to prevent texting-while-driving. Since texting behaviors of drivers differ from those of passengers, the images captured by the front and rear cameras on a smartphone have distinguishable characteristics. These features can be cost-effectively employed to detect whether the smartphone is being used by the driver or a passenger. Our experimental results show that the accuracy of the proposed scheme is over 92% among most of the testing cases, which makes the proposed system a cost-effective tool for automotive safety via smartphone cameras.

Keywords: smartphone cameras, texting-while-driving, pattern recognition, SIFT, SURF

1. INTRODUCTION

Talking-and-texting while driving is a widely recognized dangerous behavior which dramatically increases the risk of traffic accidents. Competent authorities and studies have documented the link between distracted driving of texting (DDT) on smartphone and significantly diminished safety. The National Highway Traffic Safety Administration of USA reported [1] that 13 percent of all fatal distraction-affected crashes in 2019 involved cell phone use and texting while driving. Since 2012, Taiwan drivers caught using smartphones without the hands-free kit while operating a vehicle could face a fine of NT\$3,000 [2].

Many researches have been focused on helping the driver and his/her passengers stay away from DDT. These related contributions can be classified into off-line or on-line preventions from DDT. For off-line DDT prevention, [3] proposed a game-based, multi-player, online simulated training application with an integrated hazard warning system for further improving young and inexperienced drivers' hazard perception skills. As for on-line DDT prevention, [4] developed a mobile application coupled with an on-board embedded system to monitor the mobile usage of a driver to acquire data from the vehicle to identify driver's behavior with respect to phone usage, sudden lane changes, and abrupt breaking/speeding. All information was used in mobile application to control the driver's mobile usage as well as to report driving behavior while driving. To automatically recognize driver's misbehaviors more than DDT, such as hands off the wheel, hand reaching behind, operating the radio, smoke, drinking and *etc.*, [5] employed deep learning techni-

Received March 16, 2021; revised August 31 & December 4, 2021; accepted January 27, 2022.

Communicated by Chun-Rong Huang.

⁺ Corresponding author.

* This research was partially funded by Ministry of Science and Technology, Taiwan, Grant No. 110-2221-E-130-001.

ques to tackle the task. The multi-column CNN (convolutional neural network) was utilized with feature fusion techniques to extract multi-scale features under different receptive fields in large-amount drivers' recorded image datasets. With high computational power and large dataset, their off-line work achieved high classification accuracy in diversified driver behaviors.

In on-line DDT preventions research, using automated techniques for real-time preventing talking and texting on smartphone while driving has also become a significant topic. In the work of [6], drivers must first register their own smartphones with the mobile operator as the master smartphone. When the owner is moving, the base station will obtain the continuous position of the master smartphone and use this information to calculate the current speed. When the speed is below a certain limit during operation of the master smartphone, it is considered safe, otherwise it would be deemed dangerous and the mobile operator sends an alarm to warn the driver. The main problem with this method, beyond the reluctance of drivers to register their smartphone, is in determining whether the user is a driver or a passenger (DDP).

In our paper, the proposed DDP scheme also requires the user's own smartphone to determine whether the user is a driver or a passenger, but unlike most of the studies, cameras on the smartphone are the only major sensors utilized in our proposed scheme. Since behaviors of driver or passenger are different while using phone in a vehicle, images captured by the front and rear smartphone cameras will have different recognition features. Based on these two features, two heuristics are proposed to cost-effectively perform DDP. The proposed DDP techniques can be cooperated with other related studies such as [7-10] to improve the overall system performance for on-line DDT preventions.

The remainder of this paper is organized as follows. Details of related DDP works are discussed briefly in Section 2. The proposed DDP scheme is described in Section 3, and experimental results and evaluations from the proposed DDP scheme are shown in Section 4. Conclusions and future work are given in Section 5.

2. RELATED WORKS

To automatically perform DDP, the study of [11] proposed a scheme based on the relative distance of sound reflection by sending a Bluetooth signal to the vehicle's stereo sound system to instigate high frequency sounds. The smartphone then analyzes time differences from the left, right, front and rear speakers to determine if the user is a passenger or the driver.

In the study of iLOC [12], four smartphones are placed in the driver's seat, the front passenger's seat, the rear of the driver's seat, and the rear of the passenger's seat, respectively. These smartphones will generate different accelerometer and gyroscope information under various road conditions. The iLOC system uses these information to detect whether the smartphone user is the driver or a passenger.

In the study of [13], additional cameras were installed in front of the driver's seat to determine whether a user in the vehicle is a passenger or driver. This paper designs an activity recognition algorithm based on image processing technologies. This method can accurately detect whether the driver is texting-while-driving, but it requires additional cameras deployed in vehicle.

In these aforementioned studies, additional equipment are required to detect whether the smartphone is being used by the driver or a passenger. To remedy this, Liu *et al.* propose a scheme by using only their own smartphone [7]. That scheme uses some of the special characteristics of phone use while driving, including reduced speed, hesitation while turning, and the phone being held upright. These behaviors generate special patterns in the gyroscopes, accelerometers, and GPS sensors of the smartphone, and these patterns can be used to conduct DDP. It is based on three special patterns when a driver is using the phone: reduced speed, hesitation when turning, and the phone being held upright. First, data captured by the gyroscope sensor can be used to discern the touchstroke-induced rotation change of the smartphone. To validate the existence of first pattern, GPS sensor data is used to estimate the speed of the vehicle. If the speed before keystrokes is greater than that after keystrokes, then the first pattern is detected. To identify the second pattern, their scheme uses the GPS sensor data to detect turning behavior during keystrokes. Finally, accelerometer information is used to distinguish the phone orientation for detecting the third pattern.

Similarly, in 2017, Bo *et al.* proposed a system named TEXIVE for DDP by recognizing rich micro-movements of smartphone [8]. TEXIVE first exploits unique patterns extracted from inertial sensors in smartphones to detect whether a user is entering a vehicle or not, from which side of the vehicle, and sitting in front or rear seats. Since typo data are different in focusing and distracted condition, TEXIVE then collects the typo texting pattern by monitoring the backspace key. Based on the above information, a machine learning technique is employed to perform DDP.

In 2018, Ahmad *et al.* use the smartphone inertial measurements and door's signal for DDP [9]. In this study, a user in a car is classified into four types: driver, front passenger, rear passenger nearside, and rear passenger offside. A driver and a front passenger can be classified by the behavior whilst entering the vehicle by turning clockwise or counterclockwise. The doors signal is employed to distinguish the present of front or rear vehicle users. A probabilistic model utilizing salient relevant features of smartphone sensory data and doors signal is proposed for final driver/passenger classification.

Mariakakis *et al.* propose a WatchUDrive system which identifies whether the wearer of smartwatch is the driver or a passenger in a vehicle by the accelerometer and the camera [10]. As a driver steer the vehicle, he/she must keep hold of the steering wheel for long periods of time. The wrist orientation of a driver is different to that of a passenger. This work utilizes the roll and pitch of the smartwatch's accelerometer to classify the orientation of the user's arm. Similarly, the surroundings captured by the camera of smartwatch between driver and passenger are different. They employ GIST features from [14] for DDP. Finally, a prediction aggregation is proposed to combine the above predictions.

In 2020, Khurana *et al.* proposed a light-weight and software-only solution [15] using smartphone cameras to first observe the car's interior geometry to find smartphone's position and orientation. Then, no matter where the smartphone is docked in car or held in hand, the obtained information can be applied by machine learning technique to detect the smartphone's owner for DDP. Features from the detected lines in car's interior geometry are used in random forest classifier with default parameter and 10 trees. Their real-time DDP classifier can achieve overall accuracy of 94% when the smartphone is held in hand and 92.2% when the smartphone is docked. They also claimed that if the smartphone IMU data is further applied, the overall accuracy can be pushed to 99.8%.

3. PROPOSED DDP SCHEME

The proposed scheme uses images captured by the front and rear cameras on a user's smartphone to determine whether the user is a driver or passenger (DDP). This direct method only requires cameras which are build-in sensors in almost every smartphone on current market. Images captured by cameras can easily be processed with Scale Invariant Feature Transform (SIFT) [16] and Speeded Up Robust Features (SURF) [17] to classify the user's role in a vehicle, but these algorithms require high computational power which is beyond the capacity of most smartphones. We therefore proposed two heuristics to furnish low-complexity in the DDP scheme.

3.1 Heuristic 1

Fig. 1 shows images of users using a smartphone in the vehicle. Fig. 1 (a) shows a side-view of the driver, and Fig. 1 (b) shows the passenger sitting in the front seat, Fig. 1 (c) shows the passenger on one side of the rear seat, and Fig. 1 (d) shows a passenger sitting in the center of the rear seat. When a driver uses a smartphone while driving, his hand usually shakes and lifts slightly to facilitate seeing the content on screen of the smartphone as well as the road conditions. This causes the rear camera of the smartphone to capture other equipment in the vehicle (as shown in Figs. 2 (a) and (b)) or views outside the vehicle (as shown in Figs. 2 (c) and (d)). The differences between two successive images captured by the rear camera of the smartphone is obvious due to the movement of phone or vehicle.

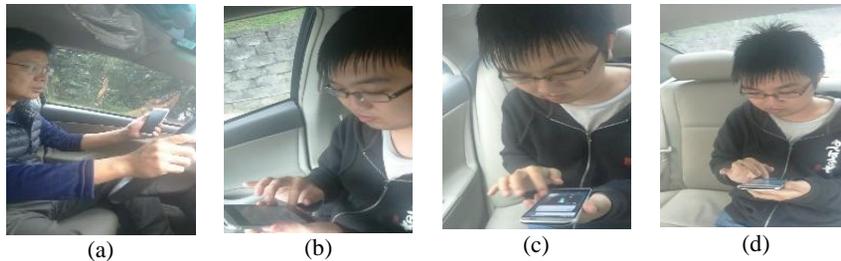


Fig. 1. Different images of users using the mobile phone in the vehicle: (a) the side-view image of the driver; (b) the image of the passenger sitting in the front seat; (c) the image of the passenger sitting in the side of the rear seat; (d) the image of the passenger sitting in the center of the rear seat.

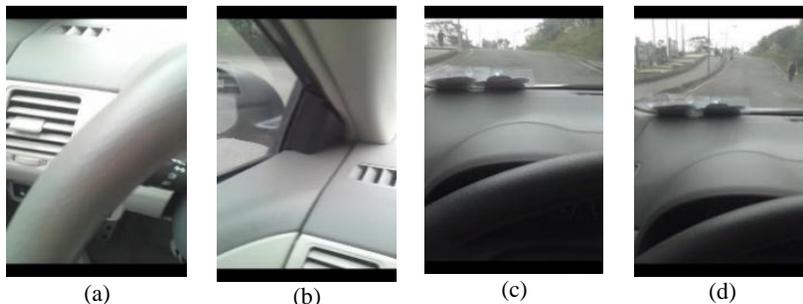


Fig. 2. Images captured by the rear camera of the mobile phone for the driver texting while driving: (a) Image 1; (b) Image 2; (c) Image 3; (d) Image 4.

In contrast, when a passenger uses a smartphone, he/she usually holds the smartphone close to their chest and looks down at the screen, so the rear camera captures images of the vehicle floor or the user’s legs. Figs. 3-5 show successive images taken by a user sitting in the front passenger seat of the vehicle, in a side rear seat, and in the center of the rear seat, respectively. Differences between these images are very slight. Therefore, successive images captured by the rear camera of a smartphone can be utilized to discern whether the user is the driver or a passenger. If there are only slight differences between two successive images, the user is considered a passenger; otherwise, the user is considered a driver. This is the first heuristic used to determine whether the user is the driver or a passenger.

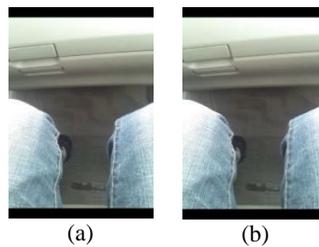


Fig. 3. Images captured by the rear camera of user’s smartphone and the user was sitting in the front seat of the vehicle: (a) Image 1; (b) Image 2.

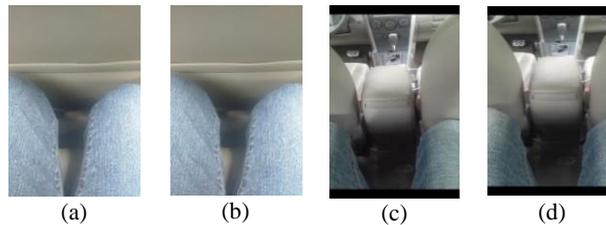


Fig. 4. Images captured by the rear camera of user’s smartphone and the user was sitting the rear seat: (a) Image 1 from first side; (b) Image 2 from 2nd side; (c) Image 3 from center; (d) Image 4 from center.

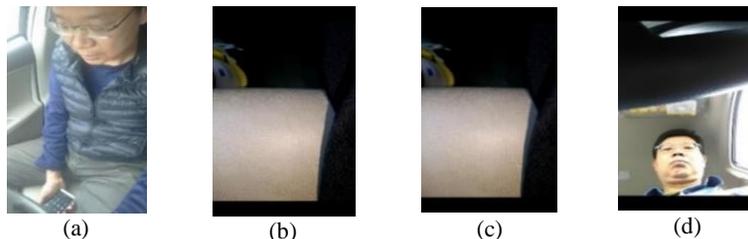


Fig. 5. Images of the driver using the mobile phone under steering wheel: (a) Side-view image for driver; (b) Image 1 captured by the rear camera; (c) Image 2 captured by the rear camera; (d) Image 3 captured by the front camera.

3.2 Heuristic 2

In most cases, Heuristic 1 can correctly distinguish whether the user is a driver or a passenger, but it fails when the driver uses the smartphone facing downwards, as shown in Fig. 5 (a). In this position, the rear camera would capture images of the vehicle floor or the

user's legs, as shown in Figs. 5 (b) and (c). The difference between these successive images is so slight that Heuristic 1 cannot work correctly.

In this situation, the driver tends to hold the smartphone more away from his/her chest in order to maintain a clear line of sight, so the image of the user's face in the smartphone's front camera is smaller, as shown in Fig. 5 (d). Fig. 6 (c) shows a corresponding image of a passenger in the front seat of the vehicle. Heuristic 2 uses this feature to distinguish whether the user is a driver or a passenger.

To reduce the calculation load, Heuristic 2 counts the number of skin pixels instead of computing the face size in image. For this intention, images captured by the front camera while users are texting are stored and processed to obtain the number of skin pixels, which is then used as thresholds in Heuristic 2. If the number of skin pixels captured by the front camera is within the range of these thresholds, the user is a passenger; otherwise, the user is a driver.

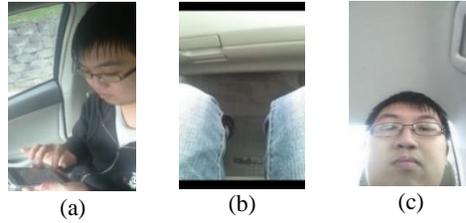


Fig. 6. Images of the passenger sitting in the front seat of the vehicle for Heuristic 2: (a) Side-view image of the driver; (b) Image captured by the rear camera; (c) Image captured by the front camera.

There are three phases in the proposed algorithm: the first phase is the color transform for preprocessing, while the second and third phases are Heuristics 1 and 2, respectively. In the first phase, pixels in the RGB color space are transformed into the YC_bC_r color space to overcome the light problem. Eq. (1) gives the corresponding transform function.

$$\begin{bmatrix} Y \\ C_b \\ C_r \end{bmatrix} = \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix} + \begin{bmatrix} 65.481 & 128.533 & 24.966 \\ -37.797 & -74.203 & 112 \\ 122 & -93.786 & -84.214 \end{bmatrix} \cdot \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (1)$$

In the second phase, the differences between successive frames are computed to do background subtraction [16], and only Y values of pixels in frames are taken into account to reduce computation load. With Y_j^i as the Y value in the i th frame of the j th pixel, differences of the i th frame are defined as

$$D^i = \sum_{j=1}^n |Y_j^i - Y_j^{i-1}| / n, \quad (2)$$

where n is the number of pixels in each frame. If $D^i > TH_1$, the second phase judges the user to be a driver. If not, the proposed algorithm performs the third phase. In this paper, TH_1 is set to 10 which is based on results conducted previously. That is, if the difference in Y value per pixel between successive frames is greater than 10, the user of the smartphone is judged as a driver.

The third phase uses skin detection to determine the face area of a user's image [20].

A pixel is classified as a skin pixel if the corresponding Y value is between 60 and 255, the C_b value is between 97 and 142, and the C_r value is between 134 and 176. With S^i as the number of skin pixels in the i th frame, if $S^i > TH_2$ and $S^i < TH_3$, then the third phase judges the user to be a passenger. Otherwise, the user is judged to be a driver. To determine the values of TH_2 and TH_3 , the user is asked to take a picture using the front camera of the smartphone in a normal texting pose when the proposed system is installed. With the number of skin pixels in the recorded picture as m , then TH_2 and TH_3 are set be to $0.9 \times m$ and $1.1 \times m$, respectively. It should be noted that different skin colors affect the skin detection algorithm. Therefore, different YC_bC_r models of skin colors are required for different skin colors. Fig. 7 shows the flowchart of the proposed system.

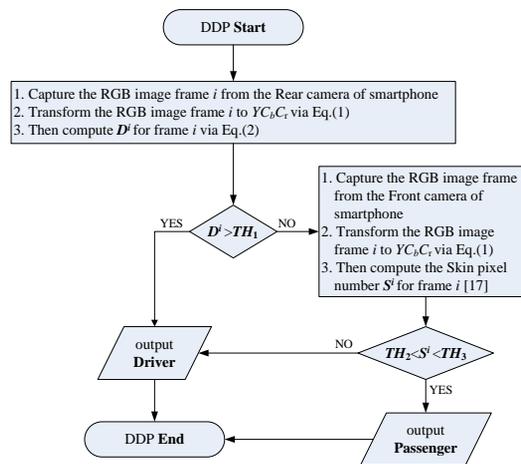


Fig. 7. Flowchart of the proposed DDP algorithm using Heuristics 1 and 2.

3.3 Storage Consumption and Computation Complexity of Proposed DDP

As shown in Fig. 7, there are three phases in the proposed algorithm. The first phase is the color transformation of the captured image frame from the rear camera on a smartphone. It requires 9 multipliers and 9 adders to transfer RGB color space into YC_bC_r color space for each pixel in a frame. If there are n pixels in a frame, the first phase costs $9 \times n$ multipliers and $9 \times n$ adders/subtractors.

The second phase first computes the differences between successive Y -type frames. It requires one subtractor and one adder for each pixel in a frame. For a frame with n pixels, it requires $2 \times n$ adders/subtractors. Without considering the storages for temporarily processing image frames, two storages are necessary to keep D^i and TH_1 . For each frame, one comparator for D^i and TH_1 is then employed to judges the user to be a driver or not.

With the same color transformation as the first phase, the third phase uses skin detection to determine the face area of a user’s image from the front camera on a smartphone. It also spends computation of $9 \times n$ multipliers and $9 \times n$ adders/subtractors in color transformation. A pixel classified as a skin pixel requires 6 comparators. Three storages are necessary to keep S^i , TH_2 and TH_3 . For each frame, two comparators are then used to judges the user to be a driver or a passenger. For a frame with n pixels, it requires 3 additional

storages and $9 \times n$ multipliers, $9 \times n$ adders/subtractors, $6 \times n + 3$ comparators.

Therefore, DDP will require constant storage consumption of 5 additional storages and $O(n)$ computation complexity in $20 \times n$ multipliers, $20 \times n$ adders/subtractors and $6 \times n + 4$ comparators. We believe that proposed DDP scheme can help resource-constraint smartphone to cost-effectively distinguish drivers from passengers for texting-while-driving.

4. EXPERIMENTAL RESULTS AND EVALUATIONS

4.1 Preliminary Experimental Setting

In this paper, all experiments are conducted on a Sony Xperia C3 smartphone with a Qualcomm Snapdragon S4 Pro APQ8064 processor, 2GB memory and the Android 4.4.4 operating system. Because the proposed DDP system must run in real time, the camera image resolution is set to 176×144 pixels so that Sony Xperia C3 can handle up to 13 images per second (*i.e.* video frame rate is 13) for each camera. To verify the DDP detection ratio of the proposed algorithm, we used 10 testing videos filmed in different situations. Six videos (Videos 1-6) test whether the proposed algorithm can correctly judge whether the user of the video is a driver and 4 videos (Videos 7-10) determine if the user of the video is a passenger. There are two men in a car in these videos. The driver aged 50 drives the car and the passenger aged 26 has a seat in different locations of the car in different videos.

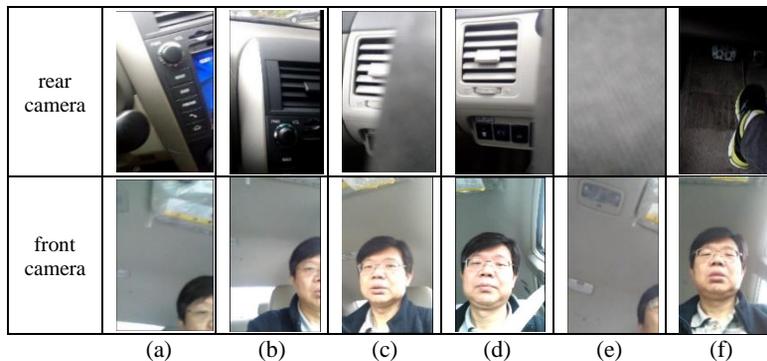


Fig. 8. Representative images for Videos 1-6: (a) Video 1; (b) Video 2; (c) Video 3; (d) Video 4; (e) Video 5; (f) Video 6.

4.2 Preliminary Experimental Results

Fig. 8 shows representative images captured by the rear camera and the front camera from Videos 1-6. Table 1 gives the detection rate of the proposed algorithm. There are 241 frames in Video 1 where a driver holds a smartphone with his right hand in front of him, and his hand movement causes large difference between successive images. Heuristic 1 can correctly determine the user to be a driver for most frames. However, since this paper uses a fixed value of TH_1 , it is not suitable for all testing images. In Video 1, there are 7 frames can't be correctly judged as a driver. From the viewpoint of individual frames, the

accuracy of Video 1 is 97%. However, from the viewpoint of the video, Video 1 can be regarded as a video with a driver. The behavior in Video 2 is similar to that of Video 1, with the driver's hand shaking and moving. From the viewpoint of individual frames, the accuracy of Video 2 is 94%, and it also can be considered as a video with a driver from the viewpoint of the video.

Table 1. Detection rates of the proposed algorithm for Videos 1-6.

| | Frames | Driver Frame | Passenger Frame | Frame Accuracy | Classified Type |
|---------|--------|--------------|-----------------|----------------|-----------------|
| Video 1 | 241 | 234 | 7 | 97% | <i>Driver</i> |
| Video 2 | 125 | 117 | 8 | 94% | <i>Driver</i> |
| Video 3 | 233 | 224 | 9 | 96% | <i>Driver</i> |
| Video 4 | 117 | 109 | 8 | 93% | <i>Driver</i> |
| Video 5 | 250 | 240 | 10 | 96% | <i>Driver</i> |
| Video 6 | 106 | 97 | 9 | 92% | <i>Driver</i> |

In Video 3, a driver holds a smartphone in front of him with his left hand, and part of the steering wheel is captured by the rear camera in this video. The movements of the steering wheel cause a lot of image changes. Heuristic 1 successfully determined that the user is a driver. The accuracy of the proposed algorithm for Video 3 is 96%. In Video 4, a driver holds a smartphone with his right hand, and part of the steering wheel is also captured by the rear camera in this video. Similar to Video 3, a lot of image changes are caused by the movement of the steering wheel. The accuracy of the proposed algorithm for this video is 93%, so Videos 3 and 4 can be considered as videos with a driver.

In Video 5, the driver holds a smartphone with his right hand resting on an object, so there are fewer differences in successive images captured by the rear camera and Heuristic 1 misjudges the user to be a passenger. However, Heuristic 2 determines that the user is a driver since only a part of the faces is captured by the front camera in Video 5. The accuracy of the proposed algorithm for Video 5 is 97% in the view of individual frames. Video 5 can be considered as a video with a driver.

In Video 6, the smartphone is held directly in front of the driver. Since the rear camera is static now and there are few differences between successive images, Heuristic 1 does not work correctly. However, Heuristic 2 can effectively determine that the user is a driver since the number of skin pixels is not within the proposed threshold range. Of the 106 frames for this video, 97 are correct, so the accuracy of the proposed algorithm for Video 6 is 92% according to individual frames. Video 6 can also be considered as a video with a driver.

Fig. 9 shows representative images captured by the rear camera and the front camera for Videos 7-10. Table 2 gives the corresponding detection rates of the proposed algorithm. In Video 7 a passenger is sitting in the front seat, while in Videos 8 and 9 a passenger is sitting on the left and on the right sides of the rear seat, respectively. In Video 10, there is a passenger sitting in the center of the rear seat. The images in Videos 7, 8 and 9 are very stable, with only slight differences between successive images. Heuristic 1 and Heuristic 2 can correctly determine that the user of these videos is a passenger. The accuracy of the proposed algorithm for Videos 7, 8 and 9 is 100%. In Video 10, some images captured by the rear camera have large vibrations incurred by the vehicle accelerating or braking, which causes an incorrect detection by the proposed algorithm. Of the 754 frames in Video 10,

46 frames incorrectly determine that the user is a driver, for an error rate of 6%. Nevertheless, the overall accuracy rate of the proposed algorithm is still 94%. In the view of individual frames, the above detection rates are all greater than 90%. These videos can be considered as videos with a passenger.

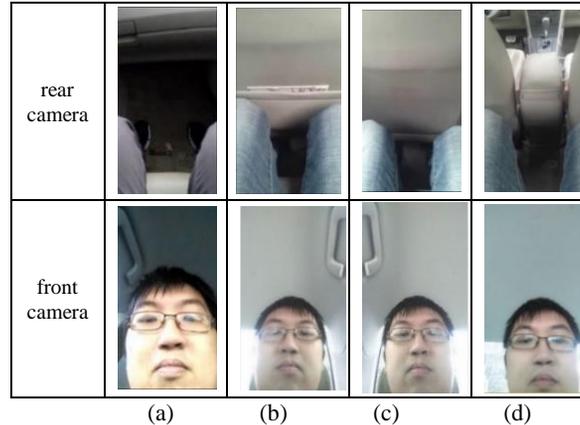


Fig. 9. Representative images for Videos 7-10: (a) Video 7; (b) Video 8; (c) Video 9; (d) Video 10.

Table 2. Detection rate of the proposed algorithm for Videos 7-10.

| | Frames | Driver Frame | Passenger Frame | Frame Accuracy | Classified Type |
|----------|--------|--------------|-----------------|----------------|------------------|
| Video 7 | 1007 | 0 | 1007 | 100% | <i>Passenger</i> |
| Video 8 | 758 | 0 | 758 | 100% | <i>Passenger</i> |
| Video 9 | 773 | 0 | 773 | 100% | <i>Passenger</i> |
| Video 10 | 754 | 46 | 708 | 94% | <i>Passenger</i> |

Table 3. Evaluation of Heuristic 1 vs. Heuristic 2 (*D* indicates driver video, *P* indicates passenger video).

| | Video 1 (<i>D</i>) | Video 2 (<i>D</i>) | Video 3 (<i>D</i>) | Video 4 (<i>D</i>) | Video 5 (<i>D</i>) | Video 6 (<i>D</i>) | Video 7 (<i>P</i>) | Video 8 (<i>P</i>) | Video 9 (<i>P</i>) | Video 10 (<i>P</i>) |
|-----------|-------------------------|-------------------------|-------------------------|-------------------------|-------------------------|-------------------------|-------------------------|-------------------------|-------------------------|--------------------------|
| H.1 | <i>D</i> | <i>D</i> | <i>D</i> | <i>D</i> | <i>P</i> | <i>P</i> | <i>P</i> | <i>P</i> | <i>P</i> | <i>P</i> |
| H.2 | <i>P</i> | <i>P</i> | <i>P</i> | <i>P</i> | <i>D</i> | <i>D</i> | <i>P</i> | <i>P</i> | <i>P</i> | <i>P</i> |
| H.1 & H.2 | <i>D</i> | <i>D</i> | <i>D</i> | <i>D</i> | <i>D</i> | <i>D</i> | <i>P</i> | <i>P</i> | <i>P</i> | <i>P</i> |

Now we look at the effectiveness of the proposed schemes for Videos 1-10 when Heuristic 1, Heuristic 2, and both are applied. Table 3 gives the corresponding results from the viewpoint of the video. Heuristic 1 can correctly determine the user to be a driver for Videos 1-4 and the user to be a passenger for Videos 7-10. It misjudges the user to be a passenger for Videos 5 and 6. The total detection rate is 80% for Heuristic 1. For Heuristic 2, it misjudges the user to be a passenger for Videos 1 and 4. Heuristic 2 correctly determines the user to be a driver for Videos 5 and 6 and the user to be a passenger for Videos 7-10. The total detection rate is 60% for Heuristic 2. Therefore, Heuristic 1 and Heuristic 2 cannot be utilized individually. The proposed algorithm combines Heuristic 1 and Heuristic 2 to achieve 100% detection rate.

4.3 Evaluations with More Experiments in Different Metrics

From Videos 1-6, different drivers' use cases are tested, and use cases of passengers in different position in a vehicle are examined in Videos 7-10. Only users' roles and their ways of using smartphone are taken into account. In the following subsections, we further look into the effect of various conditions to the system performance: user's wearing, car vibrating, in-car luminance, position of the phone, weather and vehicle types. Then, we further compare the performance of our DDP scheme with other DDP schemes using smartphone cameras only including [15] and using deep neural network. Finally, we discuss the limitations of our work.

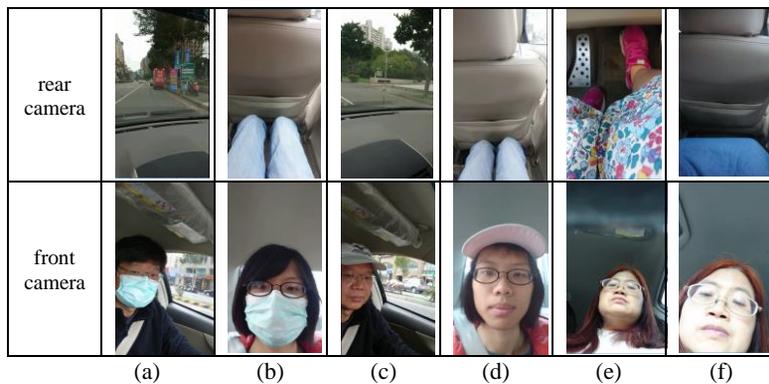


Fig. 10. Representative images captured by the rear camera and the front camera for Videos 11-16: (a) Video 11(D); (b) Video 12(P); (c) Video 13(D); (d) Video 14(P); (e) Video 15(D); (f) Video 16(P).

Table 4. Detection rates of proposed DDP algorithm for Videos 11-16.

| | Frames | Driver Frame | Passenger Frame | Frame Accuracy | Classified Type |
|--------------|--------|--------------|-----------------|----------------|-----------------|
| Video 11 (D) | 245 | 245 | 0 | 100% | Driver |
| Video 12 (P) | 280 | 280 | 0 | 0% | Driver |
| Video 13 (D) | 260 | 251 | 9 | 96% | Driver |
| Video 14 (P) | 195 | 39 | 156 | 80% | Passenger |
| Video 15 (D) | 260 | 252 | 8 | 97% | Driver |
| Video 16 (P) | 224 | 5 | 219 | 97% | Passenger |

(A) Performance from different user's wearing

The effect caused by different wearing is first analyzed in Fig. 10. Videos 11 and 12 show a man wearing a mask driving a car and a passenger wearing a mask sitting in a backseat of a car, respectively. Similarly, Videos 13 and 14 are videos of a man wearing a hat. Videos 15 and 16 show a woman wearing a skirt driving a car and sitting in the backseat of a car.

Fig. 10 shows representative images captured by the rear camera and the front camera for Videos 11-16. Table 4 gives the frame detection rates of the proposed algorithm for these videos. Results show that the detection rates are greater than 90% except Videos 12

and 14. For Video 11, the proposed algorithm can correctly determine it as a video with a driver since the differences of the successive frames captured by the rear camera are obvious. However, the proposed algorithm misjudges Video 12 as a video with a driver. In this video, the mask covers the user's face, resulting the number of skin pixels is not within the proposed threshold range.

For Videos 13 and 14, the proposed algorithm classifies them into the right types. It should be noted that the proposed algorithm would make a wrong decision for Video 14 if the hat covers more user's face like Video 12 with a mask. For Video 15, the rear camera captures images of the skirt. Since the skirt is static, differences between the above successive images are few. Heuristic 1 does not work correctly. However, Heuristic 2 can effectively determine that the user is a driver since the number of skin pixels captured by the front camera is not in the threshold range. The proposed algorithm correctly determines that the user is a passenger in Video 16 as well.

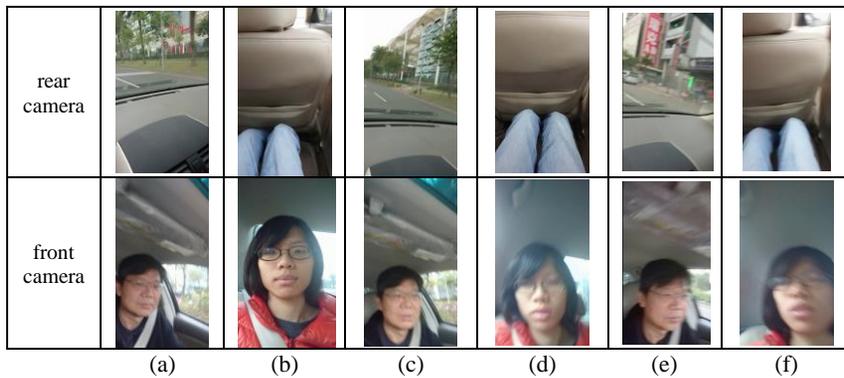


Fig. 11. Representative images captured by the rear camera and the front camera for Videos 17-22: (a) Video 17(*D*); (b) Video 18(*P*); (c) Video 19(*D*); (d) Video 20(*P*); (e) Video 21(*D*); (f) Video 22(*P*).

Table 5. Detection rates of proposed DDP algorithm for Videos 17-22.

| | Frames | Driver Frame | Passenger Frame | Frame Accuracy | Classified Type |
|-----------------------|--------|--------------|-----------------|----------------|------------------|
| Video 17 (<i>D</i>) | 240 | 236 | 4 | 98% | <i>Driver</i> |
| Video 18 (<i>P</i>) | 220 | 13 | 207 | 94% | <i>Passenger</i> |
| Video 19 (<i>D</i>) | 235 | 223 | 12 | 95% | <i>Driver</i> |
| Video 20 (<i>P</i>) | 224 | 43 | 181 | 81% | <i>Passenger</i> |
| Video 21 (<i>D</i>) | 240 | 240 | 0 | 100% | <i>Driver</i> |
| Video 22 (<i>P</i>) | 240 | 240 | 0 | 0% | <i>Driver</i> |

(B) Performance from car vibration

In this subsection, the effect to the proposed system caused by different vibrating is discussed. Videos 17 and 18 are videos in which the vibrating is the result of uneven roads. Videos 19 and 20 are videos in which the user touching the screen of the smartphone frequently. Videos 21 and 22 are videos where the user shakes the smartphone on purpose. Fig. 11 shows representative images captured by the rear camera and the front camera for

Videos 17-22. Table 5 gives the frame detection rates of the proposed algorithm for these videos.

Performance results show that the detection rates are also greater than 80% except Video 22. For Videos 17-20, the proposed algorithm can correctly classify them into corresponding types since the vibration caused by the uneven roads and the user touching the screen is not clear, although there is some distortion in images captured by cameras. However, the vibration caused by shaking the smartphone on purpose is significant in Videos 21 and 22. Such vibration makes more differences between successive images captured by the rear camera, resulting that the proposed algorithm misjudges in Video 22. The proposed algorithm determines them as videos with a driver.

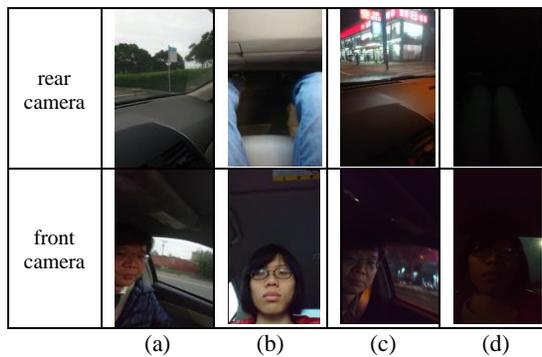


Fig. 12. Representative images captured by the rear camera and the front camera for Videos 23-26: (a) Video 23(*D*); (b) Video 24(*P*); (c) Video 25(*D*); (d) Video 26(*P*).

Table 6. Detection rates of proposed DDP algorithm for Videos 23-26.

| | Frames | Driver Frame | Passenger Frame | Frame Accuracy | Classified Type |
|-----------------------|--------|--------------|-----------------|----------------|------------------|
| Video 23 (<i>D</i>) | 224 | 213 | 11 | 95% | <i>Driver</i> |
| Video 24 (<i>P</i>) | 220 | 42 | 178 | 81% | <i>Passenger</i> |
| Video 25 (<i>D</i>) | 242 | 242 | 0 | 100% | <i>Driver</i> |
| Video 26 (<i>P</i>) | 250 | 250 | 0 | 0% | <i>Driver</i> |

(C) Performance from different in-car luminance

This paragraph analyzes the effect to the proposed system caused by different light. Videos 23 and 24 are testing videos for driving at dusk and Videos 25 and 26 are testing videos for driving in a night. Fig. 12 shows representative images captured by the rear camera and the front camera for Videos 23 to 26. Table 6 gives the frame detection rates of the proposed algorithm for these videos. Results show that the detection rate are greater than 80% except Video 26. The proposed algorithm can correctly classify them into corresponding types for Videos 23 and 24. This indicates that the proposed algorithm works well under the condition of driving at dusk. However, it makes some mistakes under the condition of driving in a night.

For Video 25, the proposed algorithm can correctly determine it as a video with a driver since the in-car luminance in the nights makes the differences between successive images captured by the rear camera greater than the threshold. However, the proposed al-

gorithm misjudges Video 26 as a video with a driver because that the number of skin pixels captured by the front camera is too few in the night, resulting that Heuristic 2 does not work correctly.

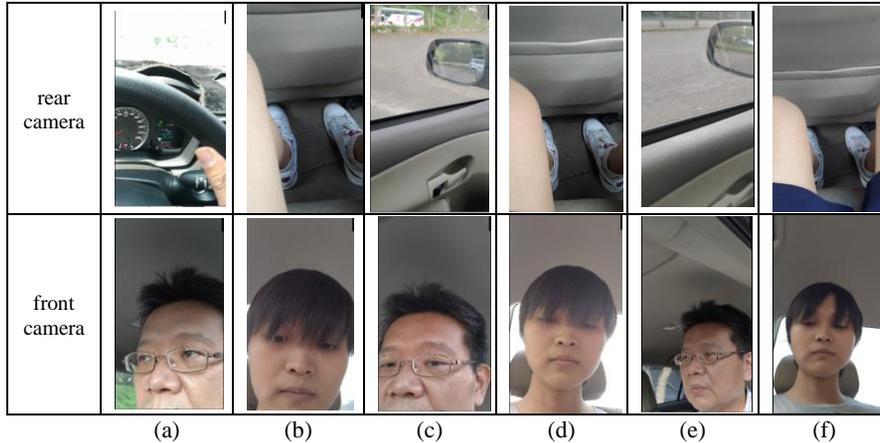


Fig. 13. Representative images captured by the rear camera and the front camera for Videos 27-32: (a) Video 27(D); (b) Video 28(P); (c) Video 29(D); (d) Video 30(P); (e) Video 31(D); (f) Video 32(P).

Table 7. Detection rates of proposed DDP algorithm for Videos 27-32.

| | Frames | Driver Frame | Passenger Frame | Frame Accuracy | Classified Type |
|--------------|--------|--------------|-----------------|----------------|-----------------|
| Video 27 (D) | 224 | 207 | 17 | 92% | Driver |
| Video 28 (P) | 224 | 5 | 219 | 98% | Passenger |
| Video 29 (D) | 224 | 215 | 9 | 96% | Driver |
| Video 30 (P) | 224 | 4 | 220 | 98% | Passenger |
| Video 31 (D) | 224 | 222 | 2 | 99% | Driver |
| Video 32 (P) | 224 | 0 | 224 | 100% | Passenger |

(D) Performance from different position of the phone

In this subsection, the effect to the proposed system caused by different position of the phone is discussed. Three modes are defined to evaluate the performance: the near, moderate, and far modes. The near denotes that the distance between the phone and the face is about 20 centimeters. The corresponding distances for moderate and far modes are 30 and 40 centimeters, respectively. Fig. 13 shows representative images captured by the rear camera and the front camera for these modes in Videos 27 to 32. Table 7 gives the detection rates of the proposed algorithm for these videos. Videos of drivers can be successfully judged by the proposed scheme as *Driver* since images captured by the rear camera are almost the contents outside of cars in which the differences between the corresponding images are significant. The corresponding accuracies are 92%, 96%, and 99% for the near, moderate, and far modes in Videos 27, 29 and 31, respectively. Note that, the detection rate for Video 27 (the near mode) is the lowest among them. This is the results that the phone is so close to the face that there is a great possibility to capture the steering

wheel by the rear camera. Steering wheel is more stable, resulting in the lower differences between frames and lower accuracy. For Videos 26, 30 and 32, the proposed scheme can successfully judge them as *Passenger* since the differences between the corresponding images captured by the rear camera are tiny. The detection rates are in a range from 98% to 100%.

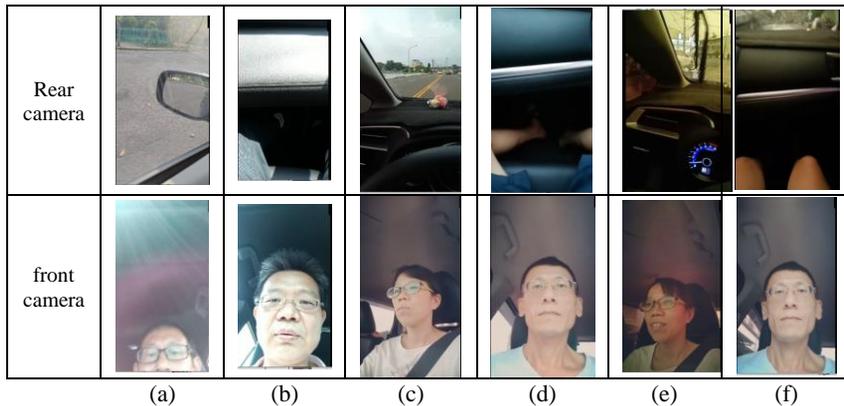


Fig. 14. Representative images captured by the rear camera and the front camera for Videos 33-38: (a) Video 33(D); (b) Video 34(P); (c) Video 35(D); (d) Video 36(P); (e) Video 37(D); (f) Video 38(P).

Table 8. Detection rates of proposed DDP algorithm for Videos 33-38.

| | Frames | Driver Frame | Passenger Frame | Frame Accuracy | Classified Type |
|--------------|--------|--------------|-----------------|----------------|------------------|
| Video 33 (D) | 224 | 222 | 2 | 99% | <i>Driver</i> |
| Video 34 (P) | 224 | 138 | 86 | 38% | <i>Driver</i> |
| Video 35 (D) | 224 | 216 | 8 | 96% | <i>Driver</i> |
| Video 36 (P) | 224 | 7 | 217 | 97% | <i>Passenger</i> |
| Video 37 (D) | 224 | 223 | 1 | 99% | <i>Driver</i> |
| Video 38 (P) | 224 | 83 | 141 | 63% | <i>Passenger</i> |

(E) Performance from different weather conditions

This paragraph analyzes the effect to the proposed system caused by different weather conditions. Videos 33 and 34 are testing videos for driving in a sunny day. Videos 35 and 38 are testing videos for driving in a rainy day. Fig. 14 shows representative images captured by the rear camera and the front camera for videos 33 to 38. Table 8 gives the frame detection rates of the proposed algorithm for these videos. The proposed algorithm can correctly classify Video 33 into *Driver* type. The corresponding accuracy is 99%. This indicates that the proposed algorithm works well under the condition of a driver texting in a sunny day. However, the proposed algorithm misjudges in Video 34 where a passenger is texting in the front seat in a sunny day. The ratio of accuracy is 38%. Thus, Video 34 is classified into *Driver* type too. The wrong reason is that the front seat is directly lightened in a sunny day resulting the differences between some images captured by the rear camera are larger than the threshold in *Heuristic 1*. Videos 35 and 36 are testing videos for driving

in a drizzle day. The experimental results indicate that the proposed algorithm works well under the condition of driving. Videos 35 and 36 are successfully classified into *Driver* and *Passenger* types. The ratios of accuracies are 96% and 97%, respectively. Videos 37 and 38 are testing videos for driving in a heavy rainy day. For Video 37, the proposed algorithm can successfully determine it as a video with a driver since the heavy rains makes the differences between successive images captured by the rear camera are significant. The corresponding ratio of accuracy is 99%. For Video 38, the ratio of accuracy to detect a passenger drops to 63%. The reason is that the luminance is low in a rainy day, resulting differences between some successive images captured by the rear camera greater than the threshold in *Heuristic 1*.

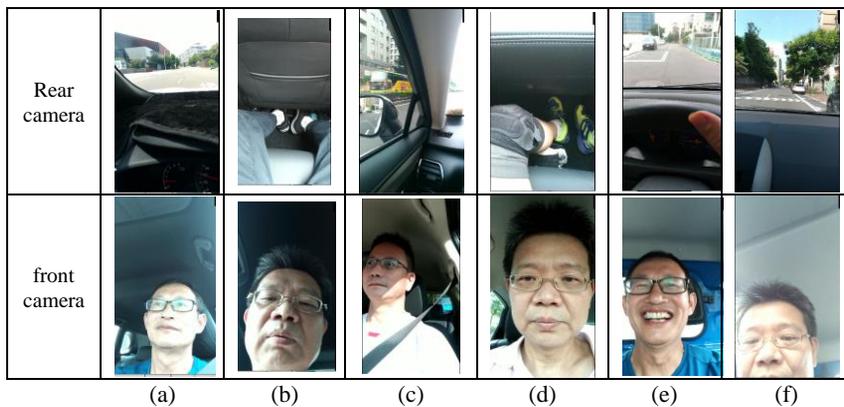


Fig. 15. Representative images captured by the rear camera and the front camera for Videos 39-44: (a) Video 39(*D*); (b) Video 40(*P*); (c) Video 41(*D*); (d) Video 42(*P*); (e) Video 43(*D*); (f) Video 44(*P*).

Table 9. Detection rates of proposed DDP algorithm for Videos 39-44.

| | Frames | Driver Frame | Passenger Frame | Frame Accuracy | Classified Type |
|-----------------------|--------|--------------|-----------------|----------------|------------------|
| Video 39 (<i>D</i>) | 224 | 217 | 7 | 97% | <i>Driver</i> |
| Video 40 (<i>P</i>) | 224 | 2 | 222 | 99% | <i>Passenger</i> |
| Video 41 (<i>D</i>) | 224 | 220 | 4 | 98% | <i>Driver</i> |
| Video 42 (<i>P</i>) | 224 | 3 | 221 | 99% | <i>Passenger</i> |
| Video 43 (<i>D</i>) | 224 | 222 | 2 | 99% | <i>Driver</i> |
| Video 44 (<i>P</i>) | 224 | 214 | 10 | 5% | <i>Driver</i> |

(F) Performance from different vehicle types

The above experiments are performed in ordinary cars. To test the performance of the proposed scheme different vehicle types, a four-wheel drive, a van, and a truck are chosen to evaluate the effect of various heights of seats in these vehicles. In the experiment, the truck comes with the highest seats and the van with the lowest seats. Fig. 15 shows representative images captured by the rear camera and the front camera for Videos 39-44. Table 9 gives the frame detection rates of the proposed algorithm for these videos. Videos of drivers can be successfully judged by the proposed scheme as *Driver* since the differences

between the corresponding images captured by the rear camera are significant too. The ratios of accuracy are 97%, 98%, and 99% for videos in a four-wheel drive, a van, and a truck (Videos 39, 41 and 43), respectively. For videos of passengers in a four-wheel drive and a van, they are also successfully judged by the proposed scheme as *Passenger*. The ratio of accuracy is 99% in Videos 40 and 42. However, the proposed scheme makes a wrong decision for the video of a truck (Video 44). It judges it as *Driver*. The ratio of accuracy is only 5%. This phenomenon is the results that the seat height of a truck is so high that the front seat of a truck is directly lightened and part of the scenes outside the truck are captured by the rear camera of the passenger, resulting in a larger difference and a lower accuracy ratio.

(G) Comparison with related DDP works

This paragraph illustrates the comparison with related DDP works. The major difference between the proposed scheme and the other works is the utilized sensors to distinguish drivers from passengers in texting-while-driving. Table 10 gives the comparison table including the model size of the number of different sensors utilized in each work. In researches of [7-13], sensors such as gyroscopes, accelerometers, or GPS are usually utilized for their DDP scheme. In [10], they use smartwatch's accelerometer sensor and camera to accomplish DDP. Particularly in [11], the Bluetooth in smartphone and car's stereo system are coupled to perform DDP. Our proposed DDP and research of [15] both utilized cameras of user's smartphone only. Therefore, Table 11 is illustrated for the comparison table between our proposed DDP and [15] of using smartphone cameras only. The well-known machine learning model of random forest is utilized in [15]. Their required resource consumption is medium and the detection accuracy is 94%, 92.2% and 99.8% for smartphone held in hand, docked and with IMU, respectively.

Our proposed DPP can further apply two kinds of DDPs using smartphone cameras only. The first one is based on heuristics mentioned above. The required resource consumption is low. The detection accuracy is 83.7% and 96.2% with and without noises, respectively. The second one is based on Mobilenetv2 model [20] of deep neural network. Then, the required resource consumption is medium and the detection accuracy is higher than 99%. We believe that the proposed DDP scheme is a good alternative for resource-constraint smartphone to cost-effectively distinguish drivers from passengers in texting-while-driving.

Table 10. Comparison table between proposed DDP scheme and other related works.

| DDP Schemes | Different Utilized Sensors in Model | Model Size |
|-----------------------|-----------------------------------------------------------------|------------|
| T&D [7] | accelerometer, gyroscope and GPS sensors of smartphone | 3 |
| TEXIVE [8, 9] | accelerometer and gyroscope sensors of smartphone | 2 |
| WatchUDrive [10] | accelerometer sensor and camera of smartwatch | 2 |
| [11] | stereo sound speakers in car and Bluetooth of smartphone | 2 |
| iLOC [12] | accelerometer and gyroscope sensors of 4 smartphones on 4 seats | 2×4 |
| CarSafe [13] | cameras, accelerometers, gyroscopes and GPS of smartphone | 4 |
| Proposed DDP and [15] | cameras of smartphone only | 1 |

Table 11. Comparison table of smartphone DDP schemes using cameras only.

| Camera-only DDP | Detection Accuracy | Classifier Model | Resource Consumption |
|-----------------------------------|------------------------------------------------------|------------------|----------------------|
| Eyes on the Road [15] | 94% (held in hand), 92.2%(docked), 99.8(with IMU) | Random Forest | Medium |
| Proposed DDP | 83.7% 96.2% (without noises) | Heuristics | Low |
| Proposed DDP using Mobilenetv2 | 99% above | Shallow DNN | Medium |

Mobilenets is a light-weight deep neural network for mobile and embedded vision applications based on a streamlined architecture that uses depth-wise separable convolutions. It uses two global hyper-parameters that efficiently trade-off between latency and accuracy to allow model builders to choose the right-sized model for their applications. Mobilenets demonstrates the effectiveness across a wide range of applications and use cases including object detection, fine-grained classification, face attributes and large-scale geo-localization.

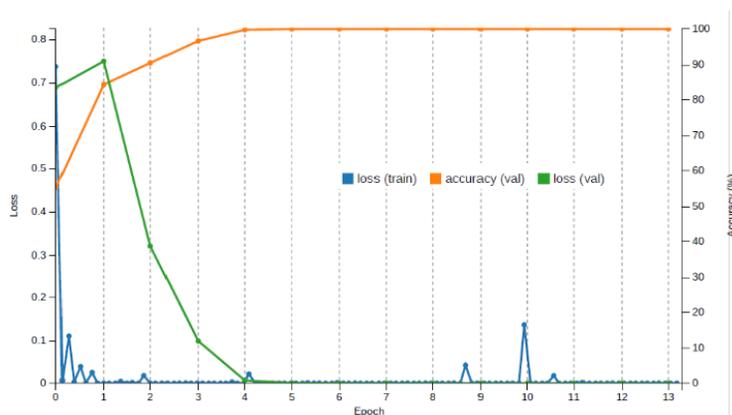


Fig. 16. Mobilenetv2 deep learning performance metrics for DDP.

Table 12. Confusion matrix of detection rates from Mobilenetv2 [21].

| | Driver | Passenger | Per-class Accuracy |
|-----------|--------|-----------|--------------------|
| Driver | 701 | 4 | 99.43% |
| Passenger | 1 | 809 | 99.88% |

Therefore, to validate detection rates for determining whether the user is a driver or a passenger via deep learning, approximate 4500 frames from aforementioned Videos 1-10 are applied in the on-device deep learning models of Mobilenetv2 [21]. The Mobilenetv2 performance metrics for DDP are shown in Fig. 16 and the confusion matrix is shown in Table 12. The detection rates for the driver and the passenger are 99.43% and 99.88%, respectively. These results indicate that the Mobilenetv2 can help our proposed heuristic DDP scheme to perform better accuracy, but the storage consumption and computation complexity of Mobilenetv2 are larger than that of the proposed heuristic DDP scheme.



Fig. 17. Demonstration examples of the fixed smartphone via hand-free kit in a vehicle.

(H) Limitations of this work

Furthermore, there is a trend that hand-free kits are used to hold the smartphone in a vehicle and some use cases are shown in Fig. 17. In such a scenario, the rear camera of smartphone usually captures the outside images. Thus, the difference between the successive captured images is obvious and proposed DDP algorithm will detect the user of the video as a driver. It results in that the texting is not allowed in vehicle. However, it's one of failure cases of the proposed DDP algorithm, if the outside images captured from rear camera keep changing. This problem cannot be overcome by images captured from cameras of a smartphone. If the smartphone charger is not available in car, power consumption is another problem to capture images from camera of a smartphone. One ideal solution is to combine the proposed scheme with other techniques utilizing the information of the 3-axes accelerometer of a smartphone.

5. CONCLUSIONS AND FUTURE WORK

In this paper we proposed an intelligent DDP scheme to prevent driver from using mobile phone while driving. Based on information captured by the front and rear cameras of user's smartphone, the proposed three-phase image processing scheme identifies whether the user is a driver or a passenger (DDP). Experimental results showed that the low-complexity DDP system has greater than 92% accuracy in about 80% of the testing cases, which makes it an efficient alternative to distinguish drivers from passengers using mobile phone in a moving vehicle. The proposed heuristics are camera-based approaches, so the cameras are the only sensors on the smart phone being used. Therefore, our proposed technique can be cooperated with other techniques using different sensors on smartphone to further improve the overall system performance for distracted driving of texting (DDT).

As mentioned in previous section, the DDP prototype implemented on the smartphone could only process up to 13 images per second, which incurs processing images from the front and rear cameras separately. To improve our proposed DDP system for practical DDT prevention in real-time is one of our near future works. A deep learning approach is one of the possible solutions to improve the accuracy of differentiating driver/passenger. However, a huge dataset is necessary for deep learning. To cost-effectively implement such an on-device deep learning system for resource-constrained smartphone DDP is also our future work.

REFERENCES

1. "Distracted driving 2019," *Traffic Safety Facts Research Note*, NHTSA, U.S. Department of Transportation, 2021, <https://crashstats.nhtsa.dot.gov/Api/Public/ViewPublication/813111>.

2. "Law targets all use of mobiles while driving," *Taipei Times*, 2012, <https://www.taipeitimes.com/News/front/archives/2012/04/06/2003529622>.
3. A. Arslanyilmaz, "Hazard warning systems to improve young distracted drivers' hazard perception skills," *Safety*, Vol. 1, 2020, p. e6010012.
4. A. Khandakar, *et al.*, "Portable system for monitoring and controlling driver behavior and the use of a mobile phone while driving," *Sensors*, Vol. 19, 2019, p. e19071563.
5. Y. Hu, M. Lu, and X. Lu, "Driving behaviour recognition from still images by using multi-stream fusion CNN," *Machine Vision and Applications*, Vol. 30, 2019, pp. 851-865.
6. O. Jr. Robert, "System for preventing text messaging while driving," US Patent 8,295,854, 2012.
7. X. Liu, J. Cao, S. Tang, Z. He, and J. Wen, "Drive now, text later: Nonintrusive texting-while-driving detection using smartphones," *IEEE Transactions on Mobile Computing*, Vol. 16, 2017, pp. 73-86.
8. C. Bo *et al.*, "Detecting driver's smartphone usage via nonintrusively sensing driving dynamics," *IEEE Internet of Things Journal*, Vol. 4, 2017, pp. 340-350.
9. B. I. Ahmad *et al.*, "Driver and passenger identification from smartphone data," *IEEE Transactions on Intelligent Transportation Systems*, Vol. 20, 2019, pp. 1278-1288.
10. A. Mariakakis, V. Srinivasan, K. Rachuri, and A. Mukherji, "Watchdrive: Differentiating drivers and passengers using smartwatches," in *Proceedings of IEEE International Conference on Pervasive Computing and Communication Workshops*, 2016, pp. 1-4.
11. J. Yang, S. Sidhom, G. Chandrasekaran, *et al.*, "Detecting driver phone use leveraging car speakers," in *Proceedings of the 17th Annual International Conference on Mobile Computing and Networking*, 2011, pp. 97-108.
12. Z. He, J. Cao, X. Liu, and S. Tang, "Who sits where? Infrastructure-free in-vehicle cooperative positioning via smartphones," *Sensors*, Vol. 14, 2014, pp. 11605-11628.
13. C.-W. You, M. Montes-de Oca, T. J. Bao, *et al.*, "CarSafe demo: Supporting driver safety using dual-cameras on smartphones," in *Proceedings of ACM Conference on Ubiquitous Computing*, 2012, pp. 547-547.
14. A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *International Journal of Computer Vision*, Vol. 42, 2001, pp. 145-175.
15. K. Rushil and M. Goel, "Eyes on the road: Detecting phone usage by drivers using on-device cameras," in *Proceedings of CHI Conference on Human Factors in Computing Systems*, 2020, pp. 1-11.
16. D. G. Lowe, "Object recognition from local scale-invariant features," in *Proceedings of the 7th IEEE International Conference on Computer Vision*, Vol. 2, 1999, pp. 1150-1157.
17. H. Bay, T. Tuytelaars, and L. van Gool, "Surf: Speeded up robust features," in *Proceedings of European Conference on Computer Vision*, 2006, pp. 404-417.
18. M. Piccardi, "Background subtraction techniques: a review," in *Proceedings of IEEE International Conference on Systems, Man and Cybernetics*, Vol. 4, 2004, pp. 3099-3104.

19. L. Phung, A. Bouzerdoum, and D. Chai, "Skin segmentation using color pixel classification: analysis and comparison," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 27, 2005, pp. 148-154.
20. A. G. Howard, *et al.* "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv Preprint*, 2017, arXiv:1704.04861.
21. M. Sandler, *et al.*, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4510-4520.



Shih-Yu Huang received the BS degree in Information Engineering from the Tatung Institute of Technology, Taipei, Taiwan, in 1988, and the MS and Ph.D. degrees from the Department of Computer Sciences, National Tsing Hua University, Taiwan, in 1990 and 1995, respectively. From 1995 to 1999, he was with the Telecommunication laboratory, Chungwa Telecom Co., Ltd., Taiwan. In 1999, he joined the Department of Computer Science and Information Engineering, Ming Chuan University, Taiwan. His current research interests are video processing and steganography.



Chia-Hui Wang received BS degree in Computer Science from Tamkang University in 1986 and received MS degree in Computer Science from New Jersey Institute of Technology, USA, in 1991. Then, he received Ph.D. degree in Computer Science and Information Engineering from National Taiwan University in 2002. His industry experience included four years as a software developer in high-tech industry. Now, he is an Associate Professor of Department of Computer Science and Information Engineering, Ming Chuan University. His research interests include multimedia communications, multimedia security and embedded systems. He is a member of IEEE and ACM.