2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 25-29, 2020, Las Vegas, NV, USA (Virtual)

Real-time Detection of Distracted Driving using Dual Cameras

Duy Tran, Ha Manh Do, Jiaxing Lu, Weihua Sheng

***Abstract*— Distracted driving is one of the main contributors to traffic accidents. This paper proposes a deep learning approach to detecting multiple distracted driving behaviors. In order to obtain more accurate detection results, a synchronized image recognition system based on two cameras is designed, by which the body movements and face of the driver are monitored respectively. The images captured from driver's body and face areas are fed to two Convolutional Neural Networks (CNNs) simultaneously to ensure the performance of classification. The data collection and validation processes of the proposed distraction detection approach were conducted on a laboratory-based assisted driving testbed to provide near-realistic driving experiences. Our dataset includes distracted and safe driving images of the drivers. Furthermore, we developed a meaningful and practical application of a voice-alert system that alerts the distracted driver to focus on the driving task. We evaluated VGG-16, ResNet, and MobileNet-v2 networks for the proposed approach. Experimental results show that by using two cameras and VGG-16 networks, we can achieve a recognition accuracy of 96.7% with a computation speed of 8 fps.**

***Index Terms*— Distracted Driving, Deep Learning, Transportation Safety**

I. INTRODUCTION

A. Motivation

Vehicle driving requires the combination of cognitive skills, physical fitness, coordination and concentration of the driver. However, drivers may get involved in activities that divert their full attention from driving, which is called distracted driving. According to a World Health Organization survey, 1.3 million people worldwide die, and 20 to 50 million people are injured in traffic accidents each year [1]. In the U.S., 391,000 injuries and 3,477 fatalities were caused by distracted driving in 2015 [2]. Every day, nine people in the U.S. are killed in accidents caused by distracted drivers [3]. In 2015, cell phone usage contributed to 14% of distracted driving related deaths. It was also reported that eating, drinking, or reading while driving has 80% more chance to cause an accident [3]. Therefore, we believe that a distraction detection system may help reduce traffic accidents. This paper aims to develop an driver assistance system that can classify distracted driving behaviors and alert drivers when needed. Due to budget and safety concerns, our research was conducted on an assisted-driving testbed, which provides near-realistic driving experiences and preliminary studies.

Duy Tran, Jiaxing Lu and Dr. Weihua Sheng (corresponding author) are with the School of Electrical and Computer Engineering, Oklahoma State University, Stillwater, OK 74078, USA, email: weihua.sheng@okstate.edu. Ha Manh Do is with the Communities to Build Active STEM Engagement (CBASE) and the Department of Engineering, Colorado State University-Pueblo, Pueblo, CO 81001, USA (e-mail: ha.do@okstate.edu).

The main contributions of this paper are: 1) proposing and realizing a real-time driver distraction detection system based on a two-Convolutional Neural Network (CNN) scheme; and 2) evaluating the proposed system using an assisted-driving testbed.

The rest of this paper is organized as follows. The related works are presented in the remainder of this section, Section II describes the overall methodology of distraction detection. Section III presents the experimental setup and results on the assisted-driving testbed. The conclusion and some potential future research directions are discussed in the final section.

B. Related works

Driver distraction detection has been studied in recent years and different approaches have been developed. The simplest one is thresholding in which the distraction is determined by comparing extracted feature values and pre-desired thresholds. In the study of Tabrizi *et al.*, the driver drowsiness is determined based on the comparison between the PERCLOS value and a threshold [4]. Other studies applied traditional machine learning methods such as Hidden Markov Model (HMM) [5], Support Vector Machine (SVM) [6]–[8], Bayesian networks [9], [10] and neural networks [11], [12] to detect driver distraction.

With the computational power of embedded computers growing continuously, deep learning has been adopted in distracted driving detection research. In 2016, the State Farm launched a distraction detection competition using dashboard images as input, which has attracted many participants to develop different approaches [13]. The approaches that implemented Convolutional Neural Networks (CNNs) achieved top performance in this competition [13]. A VGG-16 model was adopted by Colbran *et al.* to achieve an accuracy of 80% [14]. Satisfactory performance was achieved when AlexNet models were used in [15], [16]. Lőrincz *et al.* proposed an approach that applies CNNs to certain regions including head, hands, and the object held by the driver, which are utilized to detect the distracted driving behaviors [17]. Abouelnag *et al.* presented a real-time distraction detection scheme that combines five GoogleNet and five AlexNet models for skin, face, and hand features [16]. The results from the ten CNNs are merged using a weighted genetic algorithm (GA) scheme. Using their own dataset, the authors achieved an accuracy of 95.98%. However, this approach is not feasible on embedded systems because it requires a significant amount of computation resources to handle the ten CNNs in real time. Therefore, we propose a computational efficiency system that is more meaningful for practical applications. Choi *et al.* proposed a real-time distraction detection system based

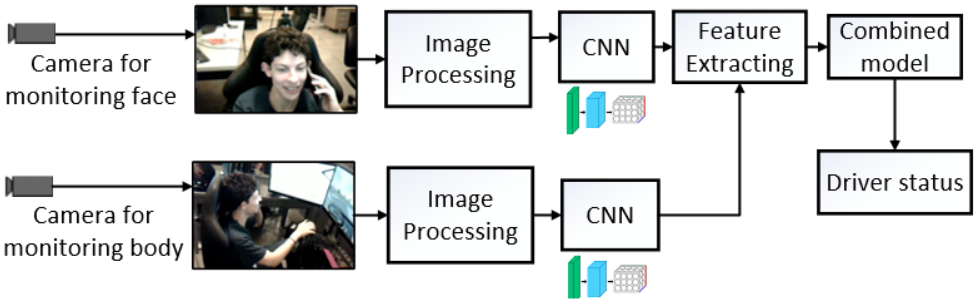
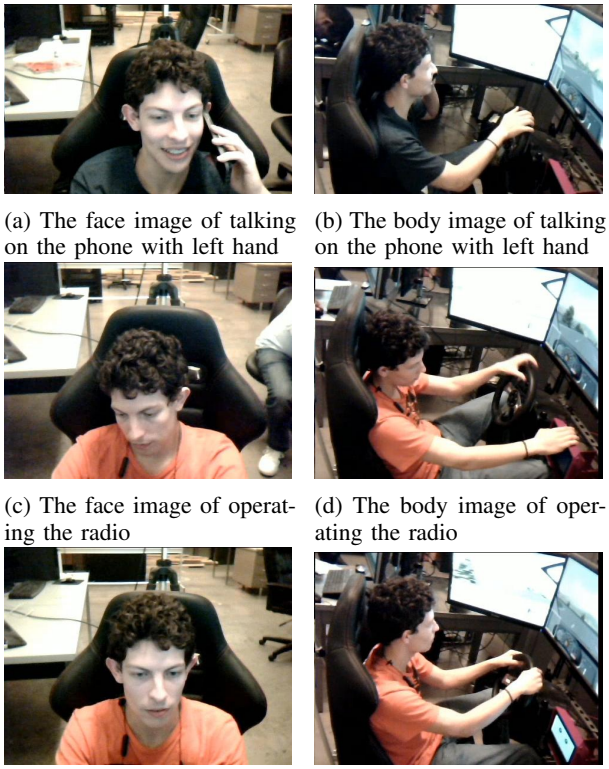


Fig. 1: The overall framework for the dual-camera based driver distraction detection.

on eye-gaze [18]. The driver's face is detected by using a Haar-feature face detector and a MOSSE (Minimizing the Output Sum of Square Error) tracker. Then nine gaze zones can be classified by using an AlexNet model. Deep learning techniques are also utilized to estimate the driver's head pose which is an important feature of distraction detection. In the study of Venturelli *et al.*, head pose is estimated from a depth camera by using a simple CNN [19]. From the image, the drive's head region is extracted by removing the background using a linear interpolation algorithm. Then the driver's head pose is estimated by using a CNN of five Convolution layers and three Fully-connected (FC) layers. This approach utilizes image input with dimensions of 64×64 which is rather small comparing with other CNN architectures. Unlike the approaches that uses a frontal camera to monitor the driver face [18], [19], we propose a system which monitors both the driver's face and body activities. Hssayeni *et al.* compared the performances of a traditional classification approach which is SVM using Histogram of Gradient features versus deep learning ones. The authors also proposed an approach that utilizes both traditional classification and deep learning algorithms. The SVM classifier takes features extracted from the convolutional layers as inputs.

Recently, Lin *et al.* proposed a bilinear CNN framework of two pre-trained CNNs which are fine-tuned on fine-grained image classification datasets [20]. Their approach is similar to ours in terms of model capacity. However, while both of their networks are trained for the same classification task, our dual-source system includes two explicitly different types of features, i.e., face and body, for the distraction classification. The two-stream network approaches have been utilized to improve video classification [21], [22]. Multi-view CNN schemes were introduced for 3D shape recognition [23], [24]. 2D images surrounding the objects are rendered based on 3D point cloud or shapes, and then multiple CNNs are utilized to classify the object. This approach has achieved dominating performance on shape classification and retrieval tasks [25]. However, due to the limitation in computer resources, running multiple CNNs may not satisfy the real-time requirements. In this study, we aim to develop a distracted-driving detection system that not only utilizes the multiple-CNN approach but also runs in real time.



(a) The face image of talking on the phone with left hand

(b) The body image of talking on the phone with left hand

(c) The face image of operating the radio

(d) The body image of operating the radio

(e) The face image of safe driving

(f) The body image of safe driving

Fig. 2: Sample images from the two cameras that monitor the driver's body and face.

II. METHODOLOGY

In tackling the problem of distracted driving, a deep learning-based scheme is adopted by using two cameras. The first camera is mounted in front of the driver and takes real-time videos of the driver's face while the driver is in action. The second camera is mounted on the right side of the driver and observes the whole of the driver's body.

The overall framework of the proposed system is shown in Figure 1. Images of the driver's face and body are fed to the VGG-16 models for face and body respectively. Then, their features are concatenated into a total feature vector which is fed to the Fully-connected (FC) layers for classification.

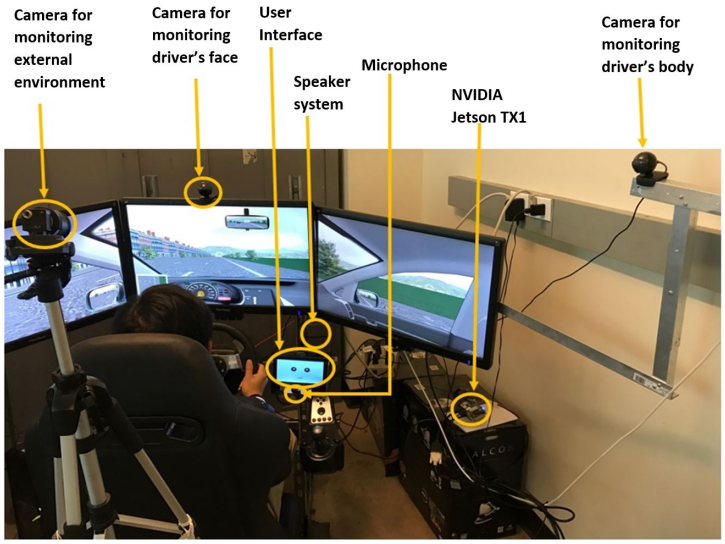


Fig. 3: The assisted-driving testbed

Figure 2 shows sample input images from the face and body cameras.

In real world, there are many distracted driving behaviors. In this study, we only consider the 10 most significant distracted driving behaviors and 1 normal behavior for comparison and validation. The 11 driving behaviors are listed below:

- s0: Safe driving
- s1: Texting on the phone using the right hand
- s2: Talking on the phone using the right hand
- s3: Texting on the phone using the left hand
- s4: Talking on the phone using the left hand
- s5: Operating the radio
- s6: Drinking
- s7: Reaching behind
- s8: Doing hair and makeup
- s9: Talking to passenger
- s10: Drowsy

A. VGG-16 model

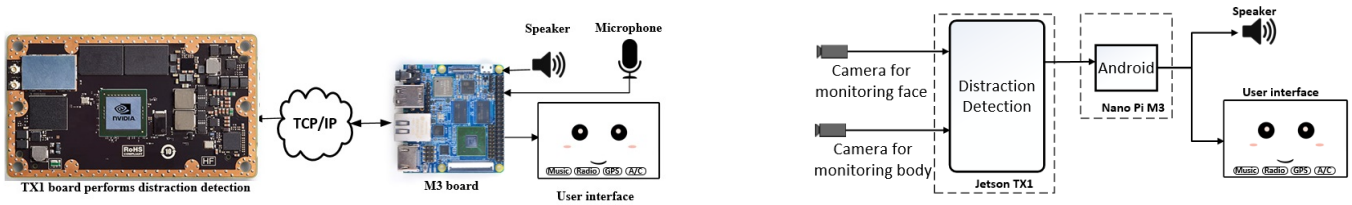
VGG-16 proposed by Simon *et al.* is one of most famous submitted model in 2014 ILSVRC (ImageNet large-scale visual recognition challenge) [26]. To this day is it still considered to be an excellent vision model. It accept 224 $\times$ 224 input images. The input image is fed through 3 $\times$ 3 convolutional layers, and max pooling layers for feature extraction. In our previous work using a single-camera setup [27], it is shown that VGG-16 has the best processing time among the four CNN architectures (VGG-16, AlexNet, GoogleNet and ResNet-152). With the proposed dual-camera setup, the computation time increases, which prompts us to select VGG-16 as the main component of the algorithm.

We adopted the transfer learning technique in which we reuse the pre-trained network. Therefore, the last Fully-connected (FC) layers is replaced by a custom-designed Multi-layer perceptron (MLP) classifier which is a type of conventional neural networks and includes several FC layers. The MLP classifier is a combination of FC layers so that the model does not miss meaningful features coming out from the convolutional layers. The classifier consists of 1024-neuron and 2048-neuron activation layers. The last FC layer has 11 neurons with corresponding to 11 classes. Then, the probabilities for each class is computed by a SoftMax function. We freeze the learning process of the beginning layers and use the pre-trained weights of these layer. We only need to train several last layers. Therefore this technique can save a significant amount of training time [27].

B. Dataset and Data Augmentation

In this project, the data was collected from twelve subjects using the driving simulator from the two cameras simultaneously. For each activity (class), the drivers were required to perform this activity only, and a six-minute video for this activity was recorded. Therefore each subject has 11 videos, i.e., all the videos were already labeled. For each video, we extract images with one second apart to avoid the situation when two consecutive frame consists of the same posture. So, we have around 77,000 body images and the same number of face images simultaneously.

In general, there is a variation of drivers' possible poses due to the differences in their heights and driving habits, i.e., some drivers lean more toward the seat while others do not. Therefore, data augmentation is necessary to avoid overfitting. We added variation to the dataset, through data augmentation, by zooming-in, rotating, shearing, and height-shifting the original images. By doing so, we obtained around 473,000 body images and the same number of face images simultaneously. The data of two random drivers (20% of the whole dataset) was selected as the validation set. Similarly the images from two other drivers were chosen as the test set. Thus, the dataset is divided into training, validation, and test set at the ratio of 6:2:2. So, of the total of 1,576,390 images (after augmentation), 945,834 images were used to train and 315,278 images to validate and 315,278 images to



(a) (b)

Fig. 4: The hardware (a) and software (b) setups of the distraction detection on the embedded system.

test our proposed model. Totally, the images were extracted to eleven classes.

C. Assisted-driving testbed

An assisted-driving testbed was developed to validate and evaluate the proposed distraction detection system. This testbed, which was developed based on a commercial driving simulator, is utilized for preliminary studies and reliability tests before conducting experiments in real traffic. The assisted-driving testbed is shown in Figure 3. It consists of a Carnetsoft driving simulator [28] and an embedded computer system for the implementation of the real-time distraction detection and alert system.

It is not suitable to adopt a powerful computer in a car due to power, cost, and size impediments. Embedded systems are more preferred because they are generally more power sufficient, cheaper, and smaller. The distraction detection algorithm is implemented to operate on an NVIDIA Jetson TX1 with a built-in GPU which helps accelerate the computation when the images are fed through the CNN layers, thus decreasing the detection time. Then the results are transferred to an NanoPi M3 board which displays a message on the touch screen or a voice alert to warn the distracted driver. The hardware and software setup of the embedded computer system is shown in Figure 4. The input RGB images are preprocessed by resizing to the required input size (224 $\times$ 224 for VGG-16 models). Then the resized images are fed to the CNN models for driver activity classification whose result is sent to a Java program running on the M3 board. Normally, the user interface acts similar to the control panel in a conventional vehicle. However, when the driver is detected to be distracted, the user interface provides a voice alert to the driver to ensure safe driving. The details of the development of assisted-driving testbed is presented in [27].

III. EXPERIMENTS & RESULTS

We conducted experiments on the assisted-driving testbed as shown in Figure 3 to validate and evaluate the proposed distraction detection system. In this paper, we utilize the side and front cameras to monitor the driver's body and face activities.

For the dual-camera approach, we trained the VGG-16 and MobileNet-v2 [29] models to compare the performance. Note that, for each type of the model, we trained it for both face and body images collectively. As a comparison to the single-camera approaches, our approach is compared with those using ResNet model [30], which has the best accuracy, and VGG-16, which has the least computational cost [27]. Single-camera models were trained with body images only since the body images are more recognizable and distinguishable than face images. The models were trained on a computer with the following specifications: Intel i7 4790 CPU, 16 GB RAM and NVIDIA GeForce GTX 970 GPU. To reduce the overfitting problem, we applied the following strategies for the training phrase: using data augmentation; using the dropout layers [31] with a probability of 0.5 to the first two FC layers; using batch normalization with a size of 128; and then applying Stochastic Gradient Descent Learning (SGD) [32] with learning rate (α), momentum (μ) and weight decay (γ). Table I shows the training parameter values corresponding to each model.

TABLE I: List of parameters used to train the dual-source approaches.

Model	Learning rate	Weight decay	Momentum
MobileNet-v2	3×10^{-5}	6×10^{-5}	9×10^{-1}
VGG-16	4×10^{-5}	7×10^{-1}	9×10^{-1}

TABLE II: Test accuracy and maximum computation time (fps) of the dual-camera approach when using different models.

Model	Accuracy	Max fps
MobileNet-v2	83.5%	25 fps
VGG-16	96.5%	8 fps

The validation results in terms of accuracy are illustrated in Figure 5. The VGG-16 approach (96.5%) out-performed the MobileNet-v2 (83.5%). This is because some information is lost during the squeezing process of the 1 $\times$ 1 convolution layers in the MobileNet-v2 models.

The trained models were trained to evaluate the computation time. Table II shows the test accuracy and the maximum processing speed of each model. The MobileNet-v2 models process much faster than the VGG-16 models (25 fps vs. 8 fps). Once again this is also from the squeezing process of the MobileNet-v2 models which reduces their number of parameters and thus increase the processing speed. However, since the VGG-16 models is much more accurate than the MobileNet-v2 ones ($\approx$ 13% difference), it is worthy

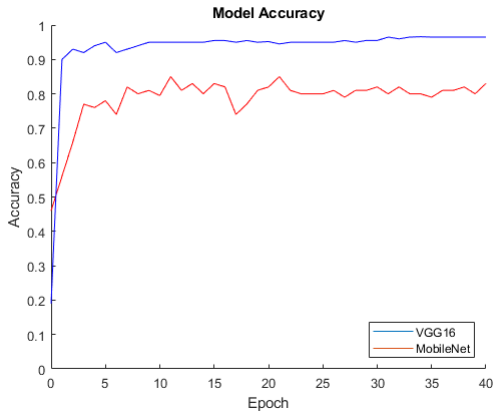


Fig. 5: The validation results of the dual-camera approach when using the VGG16 models and when using the MobileNet-v2 models.

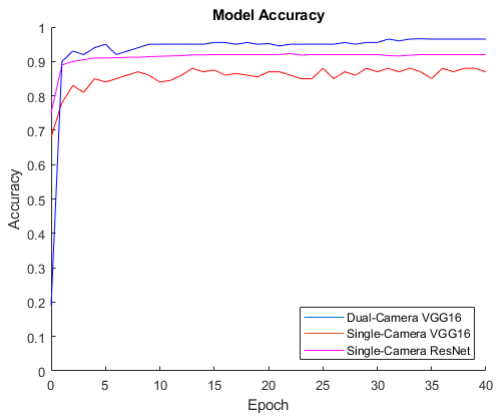


Fig. 6: The validation results when using a single camera and dual cameras.

to sacrifice the processing speed to achieve a much better accuracy. Furthermore, the features provided by the frames within one second are very similar. For VGG-16, the system can process 8 frames per second. Therefore dropping a few frames with the same features within one second is not an issue for our system. Hence, the VGG-16 model is more suitable than MobileNet-v2 for our assisted-driving testbed.

With regard to the single-camera approach, Figure 6 shows the validation results in terms of accuracy. According to the results, the VGG-16 model using dual-cameras yields a significantly better performance than the VGG-16 with a single camera. The Resnet model has lower accuracy than the dual VGG-16 model as well, while they have similar computational cost. We executed these three trained models to evaluate the computation time. The test accuracy and the maximum processing speed of each model are shown in Table III. While the dual VGG-16 approach achieves the best performance, its process speed is lower than the others, since it has notably more image to analyze. Nonetheless, the difference of process time between the dual VGG-16 model and the ResNet is only 1 fps, which is apparently small and

TABLE III: Test accuracy and maximum computation time (fps) of the VGG-16 approach using either single or dual cameras, and the Resnet approach using single camera.

Approach	Accuracy	Max fps
Dual VGG-16	96.5%	8 fps
Single VGG-16	88.9%	14 fps
Single ResNet	92.0%	9 fps

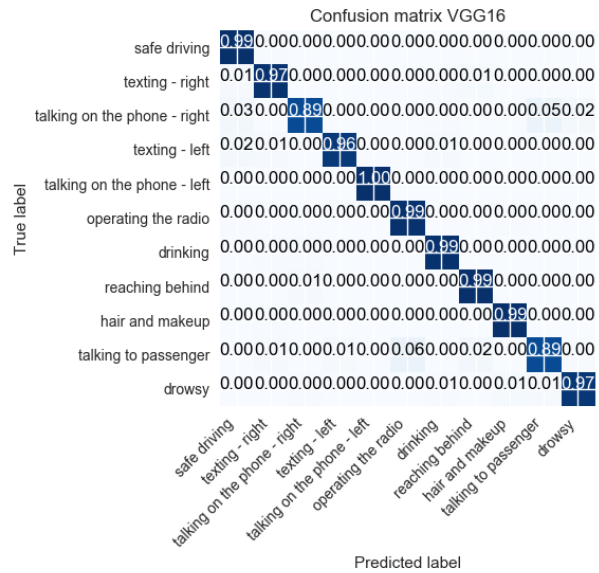


Fig. 7: Confusion matrix

acceptable. Accordingly, it is worthwhile to implement VGG-16 with the dual-camera approach to enhance the detection accuracy.

The proposed model's performance was also evaluated as shown in the confusion matrix in Figure 7. The activities of "talking on the phone with right hand" and "talking to passenger" were not classified as high as other activities. This is because drivers leaned to the right-hand side too much so that the front camera could not capture their face. This issue can be resolved by placing a front camera in a certain position so that the driver's face is always in its field of view.

IV. CONCLUSIONS & FUTURE WORKS

This paper proposed a distraction detection system which utilizes two CNN models to improve the detection performance. An assisted-driving testbed was developed to collect our own dataset, conduct experiments, and evaluate the proposed system. It can be shown that VGG-16 is suitable for our testbed. In addition, using two sources of data from the face and body cameras achieves a better performance than only using a single camera with a trade-off in reducing computation speed.

Currently, there are some mis-classifications when the driver moves out of the camera's field of view. To improve the proposed model's accuracy, we can place the front camera in a certain place so that it can always capture the driver's face. Another approach is to use sound and voice capture by in-vehicle microphones. This information is valuable for

distraction detection. In addition, the last Fully-connected layers can be replaced by the traditional classifiers such as SVM or HMM to classify the driving activities [33]. Furthermore, we can also accelerate the CNN computation by compressing the model as discussed in [34], [35].

V. ACKNOWLEDGMENT

This material is based upon work supported by the National Science Foundation under Grant No. CISE/IIS 1910993, EHR/DUE 1928711. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation. This project is also partially supported by the Natural Science Foundation of China-Shenzhen Basic Research Center Project No. U1713216 and the Open Research Project of the State Key Laboratory of Industrial Control Technology, Zhejiang University, China (No. ICT20026).

REFERENCES

[1] US Department of Transportation - National Highway Traffic Safety Administratio. Traffic safety facts. [Online]. Available: https://crashstats.nhtsa.dot.gov/Api/Public/ViewPublication/812318?-ga=1.78055380.1104132544.1489526594

[2] ——. Distracted driving. [Online]. Available: https://www.nhtsa.gov/risky-driving/distracted-driving

[3] TeenSafe. 100 distracted driving facts & statistics for 2018. [Online]. Available: https://www.teensafe.com/distracted-driving/100-distracted-driving-facts-and-statistics-2018/

[4] P. R. Tabrizi and R. A. Zoroofi, "Drowsiness detection based on brightness and numeral features of eye image," in *Fifth International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, Sept 2009, pp. 1310–1313.

[5] C. Craye and F. Karray, "Driver distraction detection and recognition using RGB-D sensor," *CoRR*, vol. abs/1502.00250, 2015. [Online]. Available: http://arxiv.org/abs/1502.00250

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

[7] Y. Liang, M. Reyes, and J. Lee, "Real-time detection of driver cognitive distraction using support vector machines," *IEEE Transactions on Intelligent Transportation Systems*, vol. 8, no. 2, pp. 340–350, June 2007.

[8] M. Kutila, M. Jokela, G. Markkula, and M. R. Rue, "Driver distraction detection with a camera vision system," in *IEEE International Conference on Image Processing*, vol. 6, no. 5, Sept 2007, pp. VI – 201–VI – 204.

[9] H. Gu and Q. Ji, "Facial event classification with task oriented dynamic bayesian network," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR*, vol. 2, June 2004, pp. II–870–II–875.

[10] Q. Ji, Z. Zhu, and P. Lan, "Real-time nonintrusive monitoring and prediction of driver fatigue," *IEEE Transactions on Vehicular Technology*, vol. 53, no. 4, pp. 1052–1068, July 2004.

[11] A. Eskandarian and R. Sayed, "Driving simulator experiment: Detecting driver fatigue by monitoring eye and steering activity," in *Proceeding of Annual Intelligent Vehicles Systems Symposium*, 2003.

[12] ——, "Analysis of driver impairment, fatigue, and drowsiness and an unobtrusive vehicle-based detection scheme," in *Proceedings of the 1st International Conference on Traffic Accidents*, 2005, pp. 35–49.

[13] State Farm Corporate. State farm distracted driver detection. [Online]. Available: https://www.kaggle.com/c/ state-farm-distracted-driver-detection

[14] S. Colbran, K. Cen, and D. Luo, "Classification of driver distraction," Stanford University, Stanford, CA, 2016. [Online]. Available: http://cs229.stanford.edu/proj2016/report/SamCenLuo-ClassificationOfDriverDistraction-report.pdf

[15] O. D. Okon and L. Meng, "Detecting distracted driving with deep learning," in *Interactive Collaborative Robotics*, 2017, pp. 170–179.

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

[17] A. Lőrincz, M. Csákvári, Á. Fóthi, Z. Á. Milacski, A. Sárkány, and Z. Tősér, "Cognitive deep machine can train itself," *arXiv preprint arXiv:1612.00745*, 2016.

[18] I.-H. Choi, S. K. Hong, and Y.-G. Kim, "Real-time categorization of driver's gaze zone using the deep learning techniques," in *International Conference on Big Data and Smart Computing (BigComp)*, Jan 2016, pp. 143–148.

[19] M. Venturelli, G. Borghi, R. Vezzani, and R. Cucchiara, "Deep head pose estimation from depth data for in-car automotive applications," *arXiv preprint arXiv:1703.01883*, 2017.

[20] T.-Y. Lin, A. RoyChowdhury, and S. Maji, "Bilinear cnn models for fine-grained visual recognition," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1449–1457.

[21] Z. Wu, X. Wang, Y.-G. Jiang, H. Ye, and X. Xue, "Modeling spatial-temporal clues in a hybrid deep learning framework for video classification," in *Proceedings of the 23rd ACM international conference on Multimedia*. ACM, 2015, pp. 461–470.

[22] Y.-G. Jiang, Z. Wu, J. Wang, X. Xue, and S.-F. Chang, "Exploiting feature and class relationships in video categorization with regularized deep neural networks," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 2, pp. 352–364, 2018.

[23] H. Su, S. Maji, E. Kalogerakis, and E. Learned-Miller, "Multi-view convolutional neural networks for 3d shape recognition," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 945–953.

[24] C. R. Qi, H. Su, M. Nießner, A. Dai, M. Yan, and L. J. Guibas, "Volumetric and multi-view cnns for object classification on 3d data," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 5648–5656.

[25] M. Savva, F. Yu, H. Su, M. Aono, B. Chen, D. Cohen-Or, W. Deng, H. Su, S. Bai, X. Bai *et al.*, "Shrec?16 track large-scale 3d shape retrieval from shapenet core55," in *Proceedings of the eurographics workshop on 3D object retrieval*, 2016.

[26] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[27] D. Tran, H. M. Do, W. Sheng, H. Bai, and G. Chowdhary, "Real-time detection of distracted driving based on deep learning," *IET Intelligent Transport Systems*, vol. 12, no. 10, pp. 1210–1219, 2018.

[28] Carnetsoft Inc., "Research driving simulator," online : http://www.carnetsoft.com/research-simulator.html.

[29] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4510–4520.

[30] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016, pp. 770–778.

[31] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting." *Journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[32] L. Bottou, "Large-scale machine learning with stochastic gradient descent," in *Proceedings of COMPSTAT*, 2010, pp. 177–186.

[33] M. D. Hssayeni, S. Saxena, R. Ptucha, and A. Savakis, "Distracted driver detection: Deep learning vs handcrafted features," *Electronic Imaging*, vol. 7, no. 10, pp. 20–26, 2017.

[34] R. Pilipovi, P. Buli, and V. Risojevi, "Compression of convolutional neural networks: A short survey," in *2018 17th International Symposium INFOTEH-JAHORINA (INFOTEH)*, March 2018, pp. 1–6.

[35] Y. Cheng, D. Wang, P. Zhou, and T. Zhang, "A survey of model compression and acceleration for deep neural networks," *arXiv preprint arXiv:1710.09282*, 2017.