Detecting Driver Drowsiness in Real Time Through Deep Learning Based Object Detection

Muhammad Faique Shakeel, Nabit A. Bajwa, Ahmad Muhammad Anwaar, Anabia Sohail, Asifullah Khan, and Haroon-ur-Rashid(✉)

Pakistan Institute of Engineering and Applied Sciences (PIEAS), Nilore, Islamabad 45650, Pakistan
haroon@pieas.edu.pk

Abstract. Vehicle accidents due to drowsiness in drivers take thousands of lives each year worldwide. This fact clearly exhibits a need for a drowsiness detection application that can help prevent such accidents and ultimately save lives. In this work, we propose a novel deep learning methodology based on Convolutional Neural Networks (CNN) to tackle this problem. The proposed methodology treats drowsiness detection as an object detection task, and from an incoming video stream of a driver, detects and localizes open and closed eyes. MobileNet CNN architecture with Single Shot Multibox Detector (SSD) is used for this task of object detection. A separate algorithm is then used to detect driver drowsiness based on the output from the MobileNet-SSD architecture. In order to train the MobileNet-SSD Network a custom dataset of about 6000 images was compiled and labeled with the objects face, eye open and eye closed. Out of these, 350 images were randomly separated and used to test the trained model. The trained model was evaluated on the test dataset using the PASCAL VOC metric and achieved a Mean Average Precision (mAP) of 0.84 on these categories. The proposed methodology, while maintaining reasonable accuracy, is also computationally efficient and cost effective, as it can process an incoming video stream in real time on a standalone mobile device without the need of expensive hardware support. It can easily be deployed on cheap embedded devices in vehicles, such as the Raspberry Pi 3 or a mobile smartphone.

Keywords: Drowsiness detection · Deep learning · Object detection · MobileNets · Single Shot Multibox Detector (SSD) · Android

1 Introduction

Vehicle crashes and accidents due to drowsy driving are prevalent all over the world. Thousands of people die every year resulting from vehicle accidents due to drowsy driving [1, 2]. Finland, Australia, England and other European countries have consistent crash reporting procedures and in data analyzed from these countries drowsy driving represents 10 to 30% of all crashes [3]. In order to reduce such accidents and

© Springer Nature Switzerland AG 2019
I. Rojas et al. (Eds.): IWANN 2019, LNCS 11506, pp. 283–296, 2019.
https://doi.org/10.1007/978-3-030-20521-8_24

enhance the safety of the driver and the passengers, driver drowsiness detection systems have been worked on and developed by various researchers all across the world.

These drowsiness detection systems can be broadly categorized to depend on the following methods [4, 5]: Vehicle Based, Behavioral Based, and Physiological Based. Vehicle based drowsiness detection systems work by monitoring the vehicle's lane changes, steering wheel rotation, speed, pressure on accelerator pedal etc. Behavior based drowsiness detection systems on the other hand depend on the behavior of the driver. To be more specific eye closure, yawn, and head posture are monitored through a camera to detect drowsiness in such systems. Lastly, physiological based drowsiness detection systems rely on the correlation between physiological signals ECG (Electrocardiogram) and EOG (Electrooculogram) to detect driver drowsiness.

All these categories of drowsiness detection system have their respective advantages and limitations. Physiological based drowsiness detection systems such as [6] have the limitation that the driver has to wear electrodes on his body that could prove to be a hindrance and an annoyance to the driver. Vehicle based drowsiness detection systems such as [7] are not robust because they are subjected to constraints related to the kind of driver and vehicle, road conditions etc. Hence, it is most practical to develop drowsiness detection systems based on the visual assessment of the drivers face as these systems do not require the driver to wear anything, and they can be implemented in any type of vehicle without modifications. Moreover, the current computer vision techniques based on convolutional neural networks enable one to develop highly robust systems.

Broadly, there are two main categories of computer vision techniques [8], traditional vision and deep learning. The traditional vision approaches extract human engineered features like edges, colors, corners, texture and hence depend on traditional image processing techniques. Among the popular traditional vision approaches are The Viola-Jones detector, The SIFT (Scale-Invariant Feature Transform) [9] algorithm, Spatial Pyramid Matching [10] and Histogram of oriented gradients (HOG) [11]. Contrary to this, deep learning based methods can automatically learn excellent abstract features by exploiting the underlying relationships in image data on their own. Hence, they provide a better representation of raw data that can be used for prediction purposes. Tiresome efforts on hand crafting features and designing the right filters are not needed in the case of deep learning methods. Deep learning methods are also very good at generalizing and hence do not suffer from the limitations of traditional computer vision techniques. For example, the Viola Jones detector requires upright face images to detect faces otherwise it won't give the desired performance, whereas deep learning models overcome this limitation. Moreover, almost all the traditional approaches listed above are largely ineffective against illumination changes, occlusion, deformation, background clutter etc. It is for this reason that deep learning based methods are now being employed to solve numerous computer vision tasks. In this work, we also employ a CNN architecture to develop the drowsiness detector application.

The main contribution of this work can be categorized in the following two aspects: (1) A new, more resource efficient, and accurate drowsiness detection methodology based on object detection using Convolutional Neural Networks (2) A new, annotated, Drowsy dataset to support the presented drowsiness detection methodology.

2 Literature Review

In this section, we attempt to briefly review the approaches used previously by researchers for vision-based drowsiness detection, along with their limitations. Eyelid closure is considered to be the most reliable indicator of drowsiness [12], and hence a lot of the systems developed seem to depend on eyelid closure for driver drowsiness detection. However, the following visual characteristics are also an indicator of driver drowsiness: a longer blink duration, yawning, slow eyelid movement, frequent nodding, fixed gaze, sluggish facial expression, and drooping posture [13].

Different algorithms have been presented in the past to detect drowsiness. Some use standard cameras [14], and some IR and stereo cameras [13, 15]. Horng et al. [16] localize the eyes using edge information, and track them using dynamical template matching to detect driver fatigue. In [13], 6 parameters are measured: Blink frequency, nodding frequency, eye closure duration, percent eye closure (PERCLOS), fixed gaze, and face position. These measured parameters are then combined using a fuzzy classifier to detect driver's drowsiness. In [17], yawning detection is done to determine if the driver is drowsy or not. Yawn and mouth regions are detected using a modified version of the Viola Jones algorithm, and the face is tracked using Kalman filter motion tracking. Danisman et al. [18] measure the distance between eyelids. This distance is then used to measure the level of drowsiness. The level of drowsiness is distinguished by the blink frequency, where the frequency increases as the person becomes sleepier and vice versa. In [19] a drowsiness detector is presented based on PERCLOS measurement, which is more robust against strong illumination variations. To reiterate, many of these methods depend on eyelid closure to detect drowsiness of the driver.

All of the above mentioned algorithms to detect drowsiness suffer from typical limitations of traditional image processing techniques, and hence may fail in varying illumination conditions, varying user appearance and fast head movements. Recently, Convolutional Neural Networks have truly revolutionized the field of computer vision, outperforming every other algorithm/technique in many applications such as image classification, object detection, emotion recognition, scene segmentation [20–23] etc.

Hence, convolutional neural networks have been employed to develop drowsiness detection systems in latest research. Dwivedi et al. [24] is one of the first attempts to have used convolutional neural networks to address this problem. As stated earlier, convolutional neural networks are able to learn an automated and efficient set of features that provide a better representation of the raw data, and hence help us classify the driver as drowsy or non-drowsy very accurately. However, [24] focused only on increasing the accuracy of drowsiness detection, and in real applications speed of the system is also a major concern.

Reddy et al. [25] presented an accurate but fast real time driver drowsiness system for embedded systems. This feat was achieved by compressing the convolutional network model. The drowsiness detection system based on two successive compressed deep neural networks in [25] was deployed to a Jetson TK1 GPU kit that is reasonably expensive (It costs $199.99 [26]). Even though the neural network was compressed in [25], the algorithm was still not efficient enough to be deployed on a Raspberry Pi 3 or mobile platform. Jabbar et al. [27] presented a compressed light weight deep neural network based method for real-time drowsiness detection on an Android device. This method is efficient enough to be deployed to an Android device, however, it achieves a reported accuracy of slightly more than 80%, which can be improved. In [28], Lyu et al. proposed a Long-term Multi-granularity Deep framework to detect driver drowsiness. They were able to achieve an accuracy of 90.05%, but the framework was not capable of being deployed to mobile devices due to its complexity.

3 Proposed Methodology

This paper presents a method to design and develop an accurate, cost-effective, and robust drowsiness detection system for real driving conditions. The research objectives were to develop a robust algorithm based on an accurate and resource efficient deep learning architecture, which could be deployed to a development board like that of a Raspberry Pi 3 or to a mobile platform such as Android/iOS. In short, our main goal was to develop a system that benefits from the unmatchable accuracies of convolutional neural networks in computer vision tasks while at the same be computationally resource efficient so that it could be deployed to cheaper embedded devices.

In order to achieve the goal of developing an accurate, resource efficient and cost-effective drowsiness detection system, we treated drowsiness detection as an object detection task. In our work, we employ the lightweight but accurate convolutional neural network architecture MobileNets [29], along with the Single Shot Multibox Detector (SSD) [30] framework on top of the MobileNet architecture. MobileNets are specifically designed for mobile vision applications, and hence were particularly well-suited for our task. MobileNet-SSD [31] is an object detection framework capable of detecting and localizing multiple objects in an image, in a single forward pass of the network. For our task of drowsiness detection, we trained it to detect human face, eye open and eye closed. Eye open and eye closed were treated as two separate objects. The incoming video stream was taken from a standard camera, as we work on the assumption that this task will be completed in daylight conditions only, and the training dataset was compiled as such on this assumption, as detailed further along in this paper. Based on these detections in a given time duration, we use a separate algorithm to determine whether the driver is drowsy or not drowsy.

The proposed methodology is accurately summarized in the flow chart below.

The threshold in the flowchart in Fig. 1 is a function of the image frames processed per second (FPS), and the longest duration of a blink was found to be equal to 400 ms [32].

$$\text{Threshold} = 0.4 \times \text{FPS} \tag{1}$$

Hence, if the processing is being done at 24 FPS, the Threshold value could come out to be 9.6–10. This means that if both of the eyes of the driver are detected to be closed for 10 successive frames in the incoming video stream (or in other words if the eyes are found to be closed for a slightly longer time than the longest average blink duration) the proposed system would declare the driver to be drowsy, and an alarm would be generated to waken the driver. This is done to prevent accidents at high speeds because in such situations a few seconds of carelessness can result in a fatal accident.

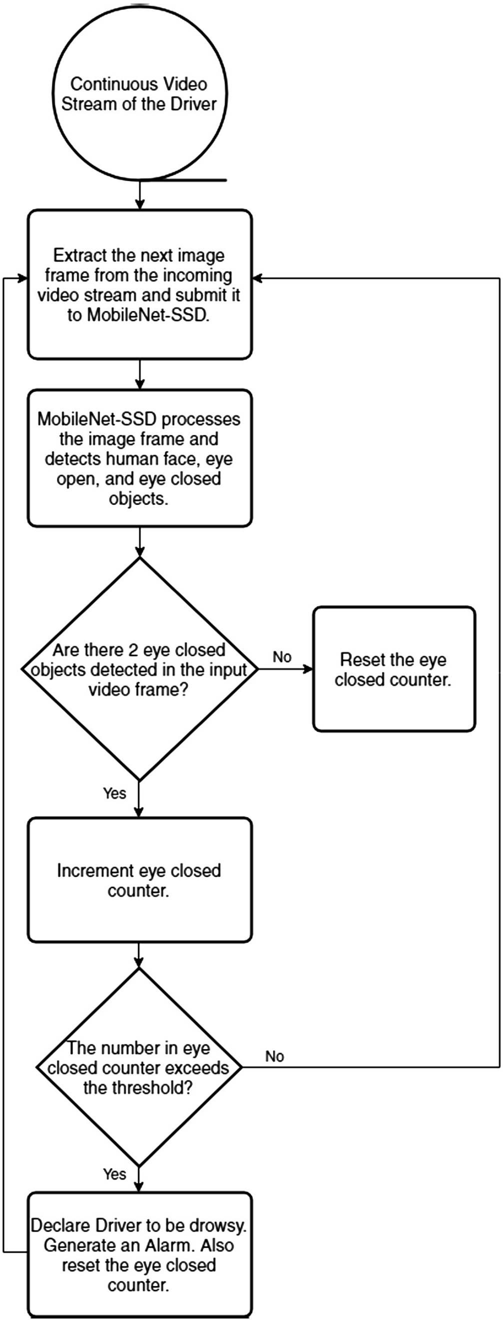

Fig. 1. Drowsiness detection system methodology

The above-mentioned method uses only a single convolutional neural network and hence the overall complexity of the system is very less. Moreover, the architecture being used, MobileNet-SSD, as mentioned previously, is developed specifically for mobile vision applications. What now remains to be seen is how good this proposed methodology works in real world driving conditions.

3.1 Drowsy Dataset

As our proposed methodology is very different from the approaches used previously to tackle the problem of drowsiness detection, no previous dataset was available which was tailored to our task. Hence, we compiled and annotated a custom dataset which was used to train the MobileNet-SSD framework.

For our task, we required annotated images that contained human faces, eye open, and eye closed. Our dataset is composed of images from a few freely available datasets online published by reputed institutions. Along with that, we also compiled data from online stock image websites. Details of the datasets and images used to create our custom dataset are given below.

FDDB

The dataset “FDDB” (Face Detection Data Set and Benchmark) [33] is provided by the University of Massachusetts, Amherst. The dataset consists of 2845 images that contain 5171 faces. The images in the FDDB dataset provide a wide range of difficulties including obstructions, problematic postures, and low resolution and out-of-focus faces. Both colored and grayscale images are included in the dataset.

The annotations for the FDDB dataset are available so it did not require us to manually annotate all the images. However, we did separate and manually annotate those images which had closed eyes in them as the FDDB dataset does not differentiate between closed eyes and open eyes. For the rest of the images, we simply re-labeled the given annotation for eyes as “Eye Open”.

YawDD

The dataset “YawDD” (Yawning while Driving) [34] is provided by the University of Ottawa, Canada. The dataset consists of two sets of videos that are recorded using two different locations of camera inside a car. For one set, the camera is placed on the dashboard of the car while for the other set, the camera is placed just underneath the front mirror of the car. Further, there are multiple drivers (male and female) in the videos. All the possible conditions that a driver may be subjected to are present in the videos. This includes driver with glasses, driver without glasses, smiling driver, yawning driver, drowsy driver and the driver while looking around.

The videos in the given dataset are recorded in 640×480 24-bit true color (RGB) 30 frames per second (fps) AVI format without audio. The total data size of the data is about 5 GB. In addition, no annotations were available for this dataset.

For this dataset, we extracted around 600 frames from all of the videos. Each individual frame extracted was then labeled by hand as annotations were not provided.

Closed Eyes in the Wild (CEW)

The dataset closed eyes in the wild [35] was compiled by Department of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, China. The dataset Closed Eyes in the Wild contains all the images of humans having closed eyes.

The images are available in JPEG format. A total of 1192 images are available which is about 21 MBs. Out of the available images we used 137 images in our dataset. In addition to this, no annotations were available for this dataset. Since annotations for Closed Eyes in the Wild (CEW) dataset were not available so we hand-labeled each individual image.

Custom Images Used

In addition to the datasets listed above, our training dataset contained images acquired by us through web-search of different image databases. The images used were open-source and licensed for re-use. The purpose of using such images was to enhance our training dataset. We selected images such that our dataset had variety of illumination conditions, a variety of poses, and so that it was more diverse.

The images are available in JPEG format. A total of 2691 images are available which is about 235 MBs in size. As these were images downloaded through various web-sources, no annotations were available for them. For this dataset, we hand-labeled features in the images for training purposes.

It is to be noted that while the compiled custom dataset is labelled with the category "yawn", it is currently not being used in the proposed methodology to detect drowsiness.

The following Fig. 2 shows some images from the Drowsy Dataset.

It can be seen from the sample images above that the Drowsy dataset has been compiled with efforts to incorporate images from a wide variety of poses, angles and illumination conditions, so as to make a diverse dataset. This is done in order to achieve high accuracy and generalization ability of the object detection framework. This dataset, however, does not contain low light images as for now the task was limited to detecting drowsiness in daylight conditions.

3.2 Training Methodology

As the compiled dataset does not contain enough training images to train the object detection framework from scratch, the concept of transfer learning [36, 37] was utilized. A MobileNet-SSD model pre-trained on the MS COCO dataset [38] was taken from the TensorFlow Object Detection model zoo [31] and was first fine-tuned on the FDDB dataset to make it detect Face and Eyes in the image. After that, we further trained this model on our custom Drowsy dataset to develop the drowsiness detection system.

Fig. 2. Drowsy dataset images

3.3 Hardware and Software Environments

The model was trained on an NVIDIA GTX 1070 GPU. TensorFlow runtime version 1.6 was used for training and evaluation. A batch size of 7 was used during the training process.

4 Experimental Results

The trained MobileNet-SSD model was evaluated on the test dataset using the PASCAL VOC evaluation metric [39]. Average Precision (AP) was calculated at 0.5 Intersection over Union (IoU) ratio for each of the individual categories, and the AP are averaged to yield the Mean Average Precision (mAP). The results are summarized in Table 1 below.

Table 1. Results summary

	AP @ 0.5 IoU eyes closed	AP @ 0.5 IoU eyes open	AP @ 0.5 IoU face	**mAP @ 0.5 IoU**
Drowsy trained MobileNet-v1-SSD	0.776	0.763	0.971	**0.837**

Figure 3 below shows the trained model in action, with bounding boxes showing the confidence scores of the detections made by the model for a particular category at a particular instant in time.

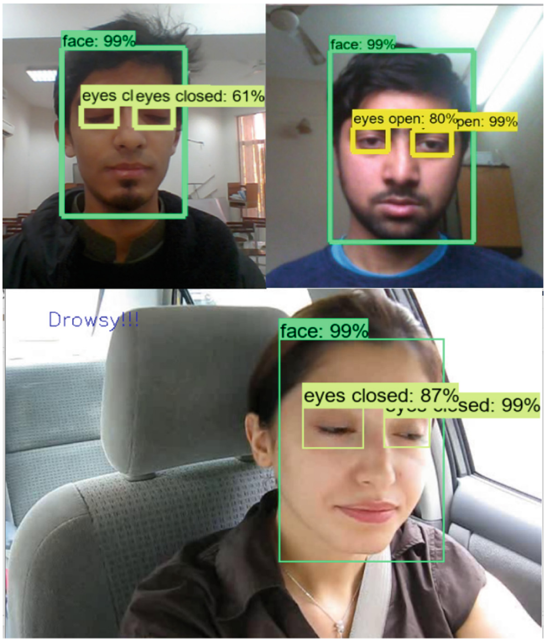

Fig. 3. Trained model in action

The trained model was also converted to a quantized TFLite model [40] for deployment to an Android mobile device. The deployment hardware used was a Sony Xperia Z smartphone [41], a somewhat outdated smartphone released in 2013. It is pertinent to mention that the smartphone used for testing was running Android OS 5.0, which does not support hardware acceleration using Android Neural Networks API. On the mentioned hardware, the trained and quantized TFLite model was able to process an incoming video frame in around 200 ms, which is deemed acceptable. On a high end phone like the Google Pixel 2, it is reported that 12–16 fps for inference can be achieved using TFLite quantized models for object detection and classification [42], but could not be independently tested on our given task.

Figures 4 and 5 illustrate the performance of the quantized TFLite model on the above mentioned Android smartphone.

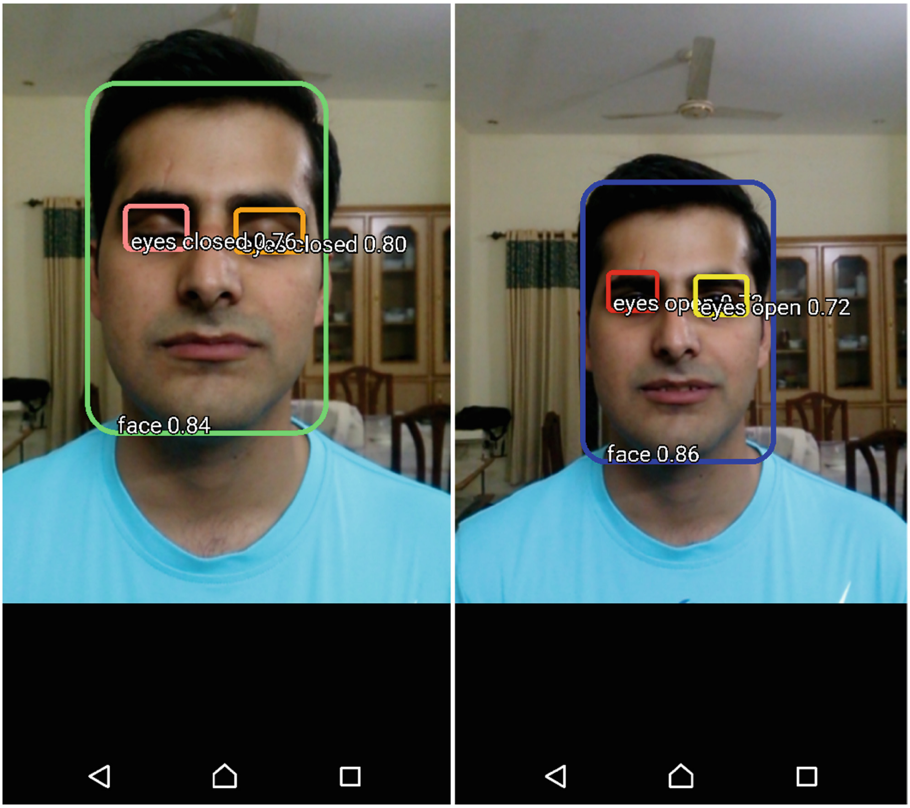

Fig. 4. Quantized model in action on an Android smartphone (Xperia Z)

5 Discussion

The custom Drowsy dataset trained object detection model was thoroughly tested in a wide variety of conditions, with varying illumination conditions, poses and occlusions, and it's performance in real world driving conditions was deemed outstanding. However, we found certain scenarios existed where its performance was not up to par. Situations where bright light was directed towards the camera lens in the background, or very low light conditions are examples of when the trained model did not perform well. However, this is deemed acceptable because the dataset compiled does not contain any images in low light conditions and hence, we do not expect it to work well in these conditions either.

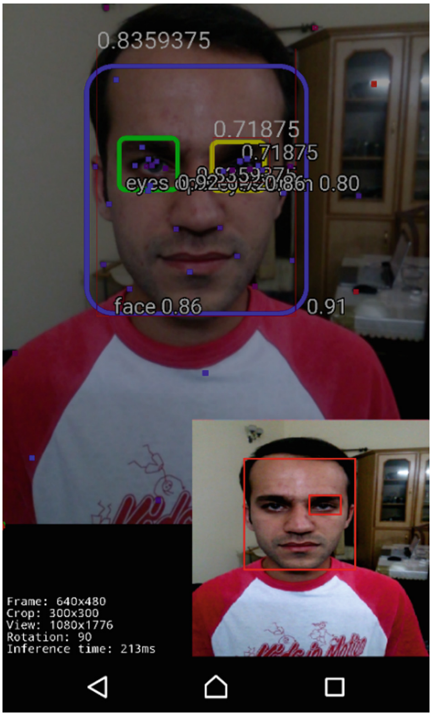

Fig. 5. Inference time for detections in the quantized TFlite model on the Sony Xperia Z

6 Conclusion and Future Works

Based on the results on the test dataset and real-world testing in a variety of illumination and occlusion conditions, we conclude that the proposed methodology of treating the task of drowsiness detection as an object detection task is practical and reliable. Moreover, this technique works in real time on an Android device, which makes it very accessible.

One major improvement that could be done in the future is that of enhancing the Drowsy dataset and adding low light images (in near infra-red light) to enable the model to detect drowsiness in low light conditions. This would be beneficial as drowsiness related accidents have a high chance of occurrence during night-time driving. Moreover, work needs to be done in incorporating yawning information (which is labeled in the dataset but not used in this methodology) in our algorithm to improve the accuracy and reliance of the drowsiness detection system.

Another area for improvement could be to modify the MobileNet-SSD architecture to better suit the drowsiness detection application.

Lastly, we would like to acknowledge the efforts of the FDDB, YawDD and CEW dataset creators for providing us with the data without which this work would not have been possible.

References

1. Mohn, T.: Around 5,000 people were killed last year due to drowsy driving. https://www.forbes.com/sites/tanyamohn/2016/08/08/nearly-83-6-million-american-drivers-are-sleep-deprived-new-report-highlights-dangers-high-cost/#3bbc82664007
2. Rapaport, L.: Drowsy drivers often behind fatal crashes. https://www.reuters.com/article/us-health-driving-sleep/drowsy-drivers-often-behind-fatal-crashes-idUSKBN15P2PM
3. Facts and Stats: Drowsy driving – stay alert, arrive alive. http://drowsydriving.org/about/facts-and-stats/
4. Saini, V., Saini, R.: Driver drowsiness detection system and techniques: a review. Int. J. Comput. Sci. Inf. Technol. **5**, 4245–4249 (2014)
5. Bhatt, P.P., Trivedi, J.A.: Various methods for driver drowsiness detection: an overview. Int. J. Comput. Sci. Eng. **9**, 70–74 (2017)
6. Chieh, T.C., Mustafa, M.M., Hussain, A., Hendi, S.F., Majlis, B.Y.: Development of vehicle driver drowsiness detection system using Electrooculogram (EOG). In: 1st International Conference on Computers, Communications, and Signal Processing With Special Track on Biomedical Engineering (CCSP), Kuala Lumpur, Malaysia, pp. 165–168 (2005)
7. Takei, Y., Furukawa, Y.: Estimate of driver's fatigue through steering motion. In: 2005 IEEE International Conference on Systems, Man and Cybernetics, vol. 2, pp. 1–6 (2005)
8. Lee, A.: Comparing deep neural networks and traditional vision algorithms in mobile robotics. Swart. Coll. (2015)
9. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vis. **60**(2), 91–110 (2004)
10. Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: spatial pyramid matching for recognizing natural scene categories, pp. 2169–2178. IEEE Computer Society (2006)
11. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: Conference on Computer Vision and Pattern Recognition (2005)
12. Bittner, R., Hána, K., Poušek, L., Smrka, P., Schreib, P., Vysoký, P.: Detecting of fatigue states of a car driver. In: Brause, R.W., Hanisch, E. (eds.) ISMDA 2000. LNCS, vol. 1933, pp. 260–273. Springer, Heidelberg (2000). https://doi.org/10.1007/3-540-39949-6_32
13. Bergasa, L.M., Nuevo, J.: Real-time system for monitoring driver vigilance. In: Proceedings of the IEEE International Symposium on Industrial Electronics, vol. III, pp. 1303–1308 (2005)
14. Suzuki, M., Yamamoto, N., Yamamoto, O., Nakano, T., Yamamoto, S.: Measurement of driver's consciousness by image processing - a method for presuming driver's drowsiness by eye-blinks coping with individual differences. In: Conference Proceedings - International Conference on Systems, Man and Cybernetics, vol. 4, pp. 2891–2896 (2007)
15. Technology | Smart Eye. http://smarteye.se/technology/
16. Fan, C.: Driver fatigue detection based. In: Proceedings of the 2004 IEEE International Conference on Networking, Sensing and Control, pp. 7–12 (2004)
17. Abtahi, S., Hariri, B., Shirmohammadi, S.: Driver drowsiness monitoring based on yawning detection. In: IEEE International Instrumentation and Measurement Technology Conference, Binjiang, Hangzhou, China (2011)
18. Danisman, T., Bilasco, I.M., Djeraba, C., Ihaddadene, N.: Drowsy driver detection system using eye blink patterns. In: Proceedings of the 2010 International Conference on Machine and Web Intelligence, ICMWI 2010, pp. 230–233 (2010)
19. Bronte, S., Bergasa, L.M., Almaz, J., Yebes, J.: Vision-based drowsiness detector for real driving conditions (2012)

20. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: Neural Information Processing Systems (2012)
21. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Conference on Computer Vision and Pattern Recognition (2016)
22. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. In: Neural Information Processing Systems, pp. 1–14 (2015)
23. Shelhamer, E., Long, J., Darrell, T.: Fully convolutional networks for semantic segmentation. IEEE Trans. Pattern Anal. Mach. Intell. **39**, 640–651 (2017)
24. Dwivedi, K., Biswaranjan, K., Sethi, A.: Drowsy driver detection using representation learning. In: Souvenir 2014 IEEE International Advanced Computing Conference, IACC 2014, pp. 995–999 (2014)
25. Reddy, B., Kim, Y.-H., Yun, S., Seo, C., Jang, J.: Real-time driver drowsiness detection for embedded system using model compression of deep neural networks. In: Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 438–445 (2017)
26. Amazon.com: NVIDIA Jetson TK1 Development Kit: Computers & Accessories. https://www.amazon.com/NVIDIA-Jetson-TK1-Development-Kit/dp/B00L7AWOEC
27. Jabbar, R., Al-Khalifa, K., Kharbeche, M., Alhajyaseen, W., Jafari, M., Jiang, S.: Real-time driver drowsiness detection for android application using deep neural networks techniques. Procedia Comput. Sci. **130**, 400–407 (2018)
28. Lyu, J., Yuan, Z., Chen, D.: Long-term multi-granularity deep framework for driver drowsiness detection (2018)
29. Howard, A.G., et al.: MobileNets: efficient convolutional neural networks for mobile vision applications (2017)
30. Liu, W., et al.: SSD: single shot MultiBox detector. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9905, pp. 21–37. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46448-0_2
31. Huang, J., et al.: Speed/accuracy trade-offs for modern convolutional object detectors. In: Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, pp. 3296–3305, January 2017
32. Schiffman, H.R.: Sensation and Perception: An Integrated Approach. Wiley, New York (2000)
33. Jain, V., Learned-Miller, E.: FDDB: a benchmark for face detection in unconstrained settings. UM-CS-2010-009 (2010)
34. Abtahi, S., Omidyeganeh, M., Shirmohammadi, S., Hariri, B.: YawDD. In: Proceedings of the 5th ACM Multimedia Systems Conference on - MMSys 2014, pp. 24–28. ACM Press, New York (2014)
35. Song, F., Tan, X., Liu, S.: Eyes closeness detection from still images with multi-scale histograms of principal oriented gradients (2014). http://parnec.nuaa.edu.cn/xtan/data/ClosedEyeDatabases.html
36. Pan, S.J., Fellow, Q.Y.: A survey on transfer learning. IEEE Trans. Knowl. Data Eng. **22**, 1345–1359 (2010)
37. Oquab, M., Bottou, L., Laptev, I., Sivic, J.: Learning and transferring mid-level image representations using convolutional neural networks. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1717–1724 (2014)
38. Lin, T.-Y., et al.: Microsoft COCO: common objects in context. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8693, pp. 740–755. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10602-1_48

39. Everingham, M., et al.: The PASCAL Visual Object Classes (VOC) challenge. Int. J. Comput. Vis. **88**, 303–338 (2010)
40. TensorFlow Lite | TensorFlow. https://www.tensorflow.org/lite/
41. Xperia TM Z (2013)
42. Krishnamoorthi, R.: Quantizing deep convolutional networks for efficient inference: a whitepaper (2018)