

Driver Drowsiness Detection and Alert System Development Using Object Detection

Jian-Da Wu*, Chia-Hsin Chang

Graduate Institute of Vehicle Engineering, National Changhua University of Education, Taiwan

Corresponding Author Email: jdwu@cc.ncue.edu.tw



<https://doi.org/10.18280/ts.390211>

ABSTRACT

Received: 8 January 2022

Accepted: 13 March 2022

Keywords:

fatigue driving, driver drowsiness detection, object detection, advanced driver assistance system

Fatigue driving is an invisible killer in car accidents and one of the main causes of traffic accidents. In order to reduce traffic accidents caused by driving fatigue, this research has developed a safety assist device to prevent such traffic accidents. In this research, a non-contact driver drowsiness detection and alert system is established in the vehicle cabin. The real-time facial image of the driver is obtained through the camera installed in front of the driver, and then the image is input to the NVIDIA Jetson TX2 embedded module. YOLO (You Only Look Once) object detection algorithm is used to detect the opening and closing of the driver's eyes, and by processing the eye area, to determine whether the driver is currently awake or fatigued while driving. The driver drowsiness detection and alert system established in this research can be applied to the vehicle interior environment to monitor the driving status. When the driver is fatigued, the system will simultaneously emit sound and light signals to promptly warn such dangerous driving behaviors. It can prevent the driver from continuing to drive when fatigued, and ensure driving safety.

1. INTRODUCTION

Major automakers have actively developed smart vehicle technologies, including advanced driver assistance systems (ADAS). According to the "2018 Global Road Safety Report" released by the World Health Organization in December 2018 [1], approximately 1.35 million deaths occur on roads worldwide every year. More than 3,000 people die in traffic accidents every day. The economic losses caused by these traffic accidents worldwide are up to 518 billion U.S. dollars [2]. Road traffic accidents cause injuries to the drivers and also bring huge economic impacts and physical damage to the victims and their families. Drowsy driving is one of the main causes of traffic accidents [3]. According to the NHTSA (National Highway Traffic Safety Administration) statistical report, the assistance of related intelligent systems (e.g. driver monitoring and warning systems), fatal car accidents in the United States were reduced from 37,526 cases in 2000 to 30,057 cases in 2013. The accident rate can be effectively reduced by nearly 20%. The development of driver drowsiness detection and alert systems is very important to prevent traffic accidents related to drowsy driving.

In the past, to avoid drowsy-driving crashes, many methods were proposed to monitor the driver's alertness. Timely alarms were triggered when abnormal driving behaviors occurred during driving. The driver's physiological signals can be obtained using two methods, contact and non-contact detection. The first method uses contact detection to analyze the driver's physiological signals. This method includes electrocardiogram (ECG) recording of the heart electrophysiological activity [4, 5] and electroencephalogram (EEG) of the brain electrophysiological activity [6-8]. Contact detection requires the subject to wear additional sensors. This type of sensor is usually installed onto the human body or scalp. The electrodes must be in contact with the human body and do

not operate continuously. The contact detection method can still be improved for the purpose of collecting continuous signals. If the detection period is long term, it will cause the driver some bodily discomfort. The second detection method analyzes the driver's physiological state. This is called non-contact detection. Non-contact detection is a detection method that does not need the driver to wear any sensors that touch the body. Eye blink frequency and duration detection is currently one of the most convenient and accurate non-contact driver drowsiness detection methods. The advantage of this method is that it is easy to use and does not require any additional operations by the driver. A non-contact driver drowsiness detection and alert system was developed in this study. The proposed method can be helpful to detect driver drowsiness and prevent road traffic accidents.

As early as 2008, Wu and Chen [9] developed a vehicle driver drowsiness warning system using fuzzy logic inference image processing technology. The system proposed in their research used the driver's facial image analysis technology and fuzzy logic to determine the driver's fatigue. That system provided a warning signal when the driver dozed off or lacked concentration. Therefore, road traffic accidents could be prevented. In 2016, Khunpisuth et al. [10] used the head tilt and of eye blink frequencies to calculate the driver's drowsiness. Embedded devices (Raspberry Pi 3 and Raspberry Pi Camera) were used to achieve closed eye detection with a driver fatigue warning system. In 2017, Reddy et al. [11] proposed a new method for real-time drowsiness detection based on deep learning. This system can be implemented on low-cost embedded devices with relatively high accuracy. This research compressed the benchmark model into a lightweight model that can be used in embedded devices (Jetson TK1). This lightweight model can be deployed on embedded devices while still maintaining reasonable accuracy. In 2020, Arefnezhad et al. [12] proposed a deep neural

network (DNN) based on a combination of convolutional neural network (CNN) and recurrent neural network (RNN) to implement a drowsiness detection system. The CNNs are used to classify different levels of drowsiness to improve driver drowsiness detection accuracy. In 2020, Maior et al. [13] used the Eye Aspect Ratio (EAR) as an indicator of eye blink detection. This study evaluated open or closed eyes by dividing each eye by 6 coordinate representations (x, y) . The aspect ratio of each eye was then calculated. The eye aspect ratio is roughly constant when the eye is open, but when the eye blinks, the eye aspect ratio will quickly drop to zero. This study used the EAR as an indicator of eye blink detection. In 2020, Ryan et al. [14] used an event camera to perform real-time face and eye tracking and eye blink detection. At that time a method was proposed to detect and analyze the driver's eye blinking by exploiting the high temporal resolution and high dynamic range of the event camera. Face detection and eye tracking can be performed at the same time to monitor the driver condition. A unique, fully convolutional RNN architecture was presented.

This research established a non-contact driver drowsiness detection and alert system in the vehicle cabin using the NVIDIA Jetson TX2 embedded device, based on the YOLO object detection algorithm. The proposed system has the advantages of high accuracy and real-time computing. The YOLO object detection algorithm is used to detect the opening or closing of the driver's eyes. By processing the eye area of the input image, the system can determine whether the driver is awake or fatigued. When the system detects that the driver is in a state of fatigue, the system will simultaneously emit sound and light signals to warn the driver that he or she should stop driving immediately and take proper rest. This non-contact driver drowsiness detection and alert system can reduce the occurrence of road traffic accidents due to fatigue driving, thereby improving driving safety.

2. PRINCIPLES OF NEURAL NETWORK AND OBJECT DETECTION

In the application of image recognition and natural language processing (NLP), CNN is often used. The network architecture is shown in Figure 1. When using a CNN for image recognition, a convolution operation will be performed on the input image, as shown in Figure 2. The convolution equation can be expressed as follows:

$$g(x, y) = \omega * f(x, y) = \sum_{dx=-a}^a \sum_{dy=-b}^b \omega(dx, dy) f(x+dx, y+dy) \quad (1)$$

where, $g(x,y)$ is the filtered image, $f(x,y)$ is the original image, ω is the filter kernel. Every element of the filter kernel is considered by $-a \leq dx \leq a$ and $-b \leq dy \leq b$. The convolution operation principle uses the convolution kernel to slide on the image, multiply the pixel gray value on the image with the value on the corresponding convolution kernel. All of the multiplied values are added to obtain the output value. In this way, the entire image is traversed by sliding the convolution kernel. Through this process, the input image features are extracted and the most suitable features are used for effective object classification.

Deep learning is a subfield of machine learning, a field that

continuously learns and improves by studying its own algorithms. As early as the 1960s and 1970s, scientists were inspired by the biological nervous system and proposed a multi-level neural network, which is the origin of the DNN. By simulating the biological nervous system, DNNs enable computers to possess high intelligence like humans. However, at that time, limited by the hardware computing power, and the acquisition of large amounts of digital data was difficult, the neural network did not obtain amazing execution results. In recent years, with the advancement of computer hardware, huge amounts of digital data can be quickly and effectively input into computers for neural network model training. When the computing power and the acquisition of a large amount of data are no longer difficult, deep learning has become a mainstream, popular technology today. The DNN has greatly promoted the AI and machine learning fields.

The DNN is roughly divided into three parts: input layer, hidden layer, and output layer, as shown in Figure 3. The DNN input and output layers correspond to the digital data input and output through neurons. The so-called deep learning means that there are many layers in the hidden layer, and each layer contains tens to hundreds of neurons. The neurons in the upper layer are passed to the neurons in the lower layer after weight calculation. As the number of iterations during model training increases, the amount of accumulated weight calculations increases, allowing the DNN to learn from past experience like the human brain, and then make reasonable decisions and respond quickly.

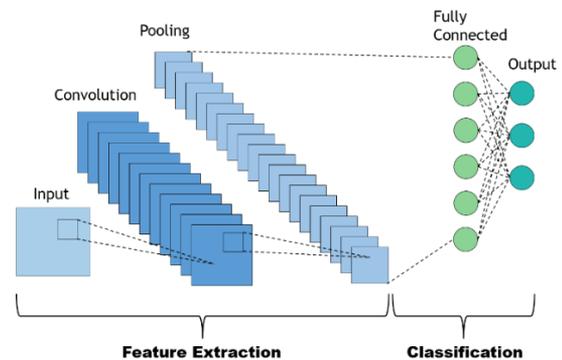


Figure 1. Convolutional neural networks

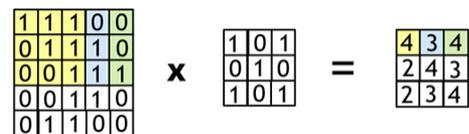


Figure 2. Convolutional layer

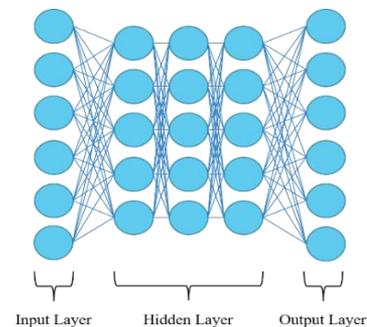


Figure 3. Deep neural networks

In 2001, the first instance of real-time face recognition appeared. In 2010, the ImageNet dataset was open sourced. It contained more than 14 million images and covered more than 20,000 categories. More than one million of them have clear category labels and the location of specific objects in the images, laying the foundation for CNN and the deep learning models used today. In 2012, a team from the University of Toronto proposed a large-scale deep convolutional neural network called AlexNet [15]. This deep convolutional neural network can significantly reduce the image recognition error rate. Computer vision is a field of AI that enables computers and systems to obtain meaningful information from digitized images, films, and other visual data, and then determine appropriate responses or suggestions based on this information. If AI enables computers to think, then computer vision allows computers to see, observe, and understand like humans. Image classification is a method using multiple sets of images labeled into different categories to train the neural network, then predict the category for new test images, and analyze the accuracy of the prediction results. At present, the most popular image classification architecture is CNN. Semantic segmentation refers to pixel blocks being distinguished according to the object category. All pixels in the image are then classified. Classification plus localization labels the location and size of a single object. Instance segmentation is a combination of object detection and semantic segmentation. The target object is detected in the image (object detection), and each pixel is marked (semantic segmentation). In this study, with the advantages of high accuracy and real-time computing, the YOLO object detection algorithm is used to detect the opening and closing of the driver's eyes.

YOLO is currently an important algorithm for real-time object detection. It was first proposed by Redmon et al. in 2015 [16]. The YOLO algorithm treats object detection as a regression problem, analyzes the target object location in the picture, and predicts which category it belongs to. The YOLO object detection algorithm prediction result from the input image to the output is realized by only a CNN. CNN can predict multiple bounding boxes at the same time and calculate the probability of different kinds of objects for each bounding box. The entire image is directly put into the neural network for training, so that the end-to-end algorithm can avoid the disadvantage of separate training for traditional object detection methods, thus greatly speeding up the hardware computing speed. The YOLO detection model evaluation mainly adopts the common object detection indicators: Intersection over Union (IoU) and Mean Average Precision (mAP). IoU was first proposed by Paul Jaccard in the early 20th century and is also called the Jaccard index. IoU is a number from 0 to 1 that calculates the overlap rate between the predicted and ground truth bounding box, that is, the ratio of their intersection to the union, as shown in Figure 4, and the calculation as expressed by Eq. (2). The ideal situation is complete overlap, when the ratio is 1. Otherwise, an IoU of 0 means that there is no overlap between the bounding boxes. IoU is generally used to evaluate whether the target object detection result is correct. The most commonly used threshold is 0.5. If $IoU > 0.5$, it means that this is a successful prediction, otherwise it will be considered a wrong prediction.

$$IoU = \frac{\text{Area of Overlap}}{\text{Area of Union}} \quad (2)$$

The mAP is the average value of the average accuracy of all

categories in the object detection data set, and its value represents the sum of the average accuracy values of all categories divided by the number of all categories, as expressed by Eq. (3).

$$mAP = \frac{\sum_{i=1}^c AP_i}{C} \quad (3)$$

where, C is the total number of categories. AP is average prediction results accuracy.

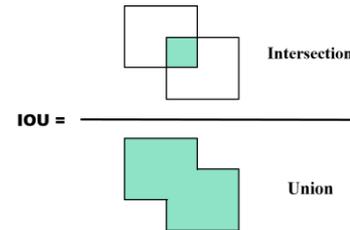


Figure 4. IoU accuracy calculation

3. IMPLEMENTATION AND EXPERIMENTAL WORK

A non-contact driver drowsiness detection and alert system is established in the vehicle cabin in this study. The equipment system structure is shown in Figure 5. The system obtains the real-time facial image of the driver through the Jetson TX2 board camera installed in front of the driver. The image is input into the NVIDIA Jetson TX2 embedded module. The program is executed on the Linux Ubuntu 18.04 LTS operating system (OS). The hardware specifications and operating system are shown in Table 1. The input image is calculated using a well-trained YOLO model to detect the opening and closing of the driver's eyes. The characteristic eye areas are processed to determine whether the driver is currently awake or fatigued while driving. If the driver is awake, no warning will be issued and the driver will continue to be monitored by the system. If the driver is fatigued the Jetson TX2 embedded module immediately outputs current to light up the LED module, and at the same time triggers a buzzer to emit a warning sound for dangerous driving behaviors. This driver drowsiness detection and alert system is implemented using the Python programming language.

There are several common driver drowsiness detection methods, including driver's operating behavior, vehicle status, and driving conditions detection, which will vary greatly due to different driving environments and driving purposes. Among them, the more reliable method is driver physiological state detection [17]. The methods for detecting the driver's physiological signals can be divided into contact and non-contact types. Contact detection, such as brain wave detection, headband camera detection, often requires the driver to wear additional sensors. If the detection time is long term, the driver will experience some bodily discomfort. Therefore, non-contact detection is used to detect the driver's fatigue state in this study. Driver's facial features recognition is an important key to determine whether the driver is awake or fatigued [18]. When people are tired, their faces often show obvious drowsiness. Schleicher [19] believes that the driver's blinking frequency is related to the degree of fatigue. Detecting the

driver's blinking eyes is one of the most convenient and accurate methods for driver drowsiness detection. According to the YOLO object detection algorithm high accuracy and real-time calculation, this study uses the YOLO algorithm to detect the driver's eye opening and closing. By processing the characteristic eye area, whether the driver is currently awake or fatigued while driving is determined.

Before using the YOLO model to detect the target object in the input image, it is necessary to perform neural network model training to obtain an optimal weight file that can be used in the real-time detection system. To improve YOLO object detection model accuracy, this study obtained facial images for the training model from six different drivers. The training images include different people (considering gender, wearing glasses or not), and the frontal rotation angle of the face, face tilt angle, image shooting distance, sharpness, brightness, to increase the quality, quantity and diversity of training data set. A total of 4,800 images were collected in this study to create a training data set. This data set was imported into four different YOLO object detection models (YOLOv3, YOLOv3-tiny, YOLOv4, YOLOv4-tiny) for training. The types and numbers of different training samples are shown in Table 2.

This experiment first classified and labeled the facial image used for training according to different categories. The driver's eye status was divided into two categories: open eyes and closed eyes for eye detection model training. The image annotation tool Labeling was used to mark the eye position on the driver's face in the training set. The eye category was labeled open or closed, as shown in Figure 6. After all training images were annotated a desktop computer was used to train four different YOLO eye detection models (YOLOv3, YOLOv3-tiny, YOLOv4 and YOLOv4-tiny). In the training process the YOLO object detection model the generated loss function curves are shown in Figure 7 and Figure 8. It can be seen that in the initial neural network model training stage, the loss function curve of the YOLOv4-tiny lightweight model had faster convergence speed than the YOLOv4 model. The learning efficiency was also higher than that of the YOLOv4 model. When the YOLO object detection model was trained, as the number of repeated operations increased, the training curve slope gradually decreased. Finally, the model learning efficiency gradually reached saturation, and the loss value was about 0.02.

The Jetson TX2 board camera is installed in front of the driver to obtain a real-time driver facial image. The input image is calculated using the well-trained YOLO model to detect if the driver's eyes are open or closed. The criterion for determining whether the driver is fatigued or not in this experiment is based on a study published by Wang et al. in the Journal of Investigative Ophthalmology and Visual Science [20]. In that study, the blink frequency and characteristics of subjects undergoing visual field tests were recorded and analyzed to establish whether the blink parameters (frequency and duration of blinking) are related to threshold variability. The results showed that the normal blink frequency is on the order of 9 to 13 per minute in the daytime, increasing to 20 to 30 per minute in sleep-deprived subjects or patients with abnormal sleep patterns. In addition, eye blinks were defined as eyelid closures with duration of 50 to 500 milliseconds. Eye closures in excess of 500 milliseconds were defined as micro sleep episodes. Accordingly, this research adopted algorithm as the judgment rule for driver fatigue. Driver drowsiness detection algorithm: IF the blinking rate is > 13 times per minute OR eyelid closure duration is > 0.5 s THEN danger.

When the driver's eye closure lasts longer than 0.5 seconds in each blink, or the number of blinks is more than 3 times per 15 seconds (the eye blink frequency is too high), it means that the driver is in a state of fatigue. The system immediately triggers the LED warning light module through the current NVIDIA Jetson TX2 output, and at the same time emits a warning sound from the buzzer to warn the driver not to continue driving. This driver drowsiness detection and alert system architecture is shown in Figure 9.

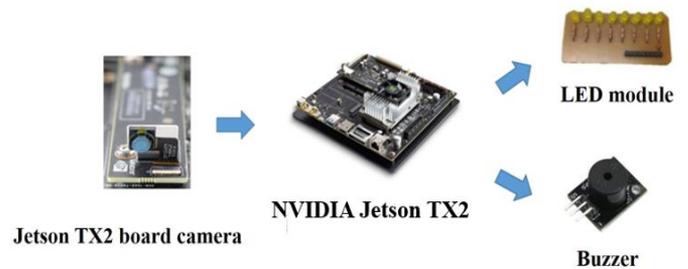


Figure 5. Structure of embedded computer and input/output equipment

Table 1. Equipment and operating environment used to operate the driver drowsiness detection and alert system

OS	Ubuntu 18.04 LTS
CPU	Dual-Core NVIDIA Denver 2 64-Bit CPU Quad-Core ARM® Cortex®-A57 MPCore
GPU	256-core NVIDIA Pascal™ GPU architecture
RAM	8GB

Table 2. Training samples of eye detection

Classification	Open	Open	Close	Close
	(Wearing glasses)	(Naked eye)	(Wearing glasses)	(Naked eye)
Samples				
A	200	200	200	200
B	200	200	200	200
C	200	200	200	200
D	200	200	200	200
E	200	200	200	200
F	0	400	0	400

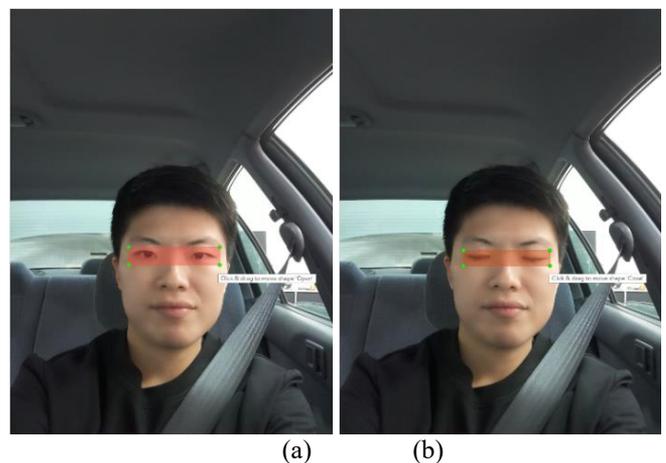


Figure 6. Label object bounding boxes in images. (a) Open eyes (b) Close eyes

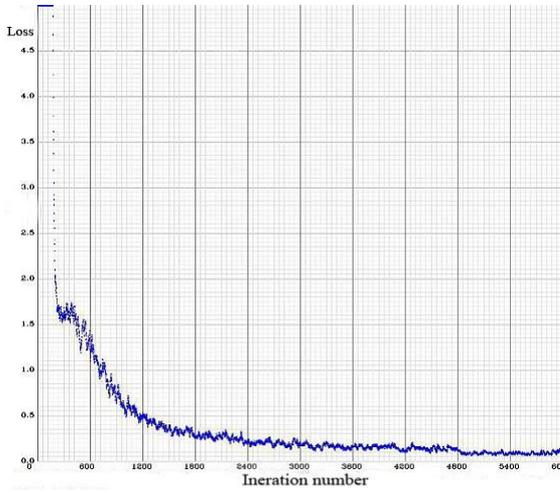


Figure 7. Loss function curve of YOLOv4 model

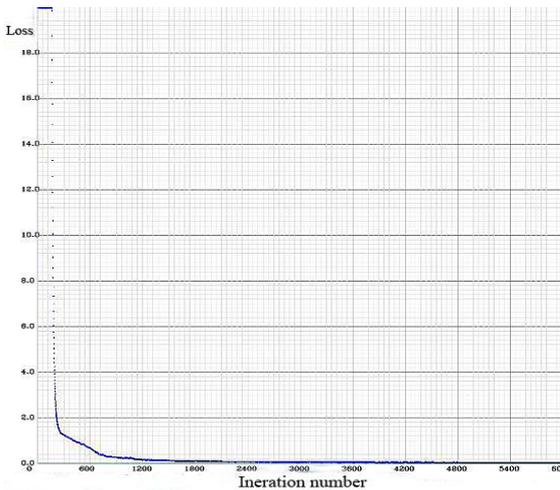


Figure 8. Loss function curve of YOLOv4-tiny model

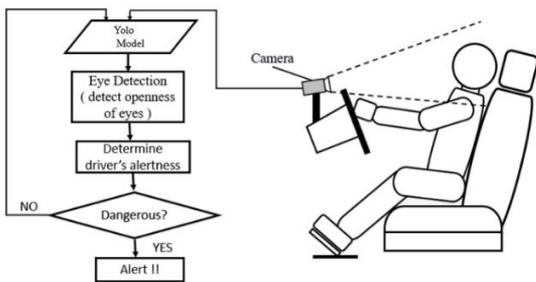


Figure 9. Flow chart of driver drowsiness detection and alert system

4. EXPERIMENTAL RESULTS AND DISCUSSION

This research trained and tested four different YOLO object detection models, including YOLOv3, YOLOv3-tiny, YOLOv4 and YOLOv4-tiny. In terms of system hardware, NVIDIA Jetson TX2 is one of the fastest and most power-efficient embedded AI computing devices. The proposed system therefore uses NVIDIA Jetson TX2 embedded modules as hardware to execute the program. After actual testing, the results show that using the tiny lightweight version of the YOLO model can achieve real-time object detection. In the case of fixing the same training sample, the accuracy rates

of four different versions of YOLO object detection models are compared, and the common evaluation indicators of object detection models: IoU and mAP are mainly used. In this study, the YOLO model performance was actually evaluated, including the internal test results, training time and detection time of the four different YOLO object detection models, as shown in Table 3. The results show that the four different versions of the YOLO object detection model all have high accuracy rates. The mAP of the YOLOv3, YOLOv3-tiny, and YOLOv4 model reaches 99.98% when the IOU threshold is 0.5 (the result is considered a successful prediction if $\text{IoU} \geq 0.5$), especially the mAP of YOLOv4-tiny model achieves 100% best performance. The results also show that all four YOLO object detection models have good performance in fast detection, especially the two lightweight versions of the YOLO model (YOLOv3-tiny and YOLOv4-tiny), which are the fastest to detect objects, and only take about 0.004 seconds to correctly detect the target object in the input image.

Table 3. Comparison of accuracy, training time and detection time with different yolo model

Model name	Average IoU	mAP@0.50	Training time (hr)	Detection time (s)
yolov3	89.11%	99.98%	8.74	0.015
yolov3-tiny	84.24%	99.98%	2.52	0.004
yolov4	90.30%	99.98%	4.31	0.019
yolov4-tiny	84.43%	100.00%	5.27	0.005

The initial YOLO neural network model training learning rate is set to 0.001, and the maximum number of iterations is 6,000. Table 4 shows the accuracy of the four YOLO object detection models in this experiment at different iterations. It can be found that as the number of iterations increases, the four YOLO models all slowly converged, and the accuracy rates were correspondingly higher and higher. In general, in the machine learning and statistical classification model fields, the confusion matrix is an analysis method that summarizes the actual and predicted classification model results, as shown in Table 5. The model accuracy is usually used to measure the model effectiveness. The accuracy rate is calculated as

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + TN + FN} \quad (4)$$

where, TP is the subjects correctly identified as positive samples while subjects were actually positive samples; TN presents the subjects correctly identified as negative samples while subjects were actually negative samples; FP is the number of subjects incorrectly identified as positive sample while subjects were actually negative samples; FN is the number of subjects incorrectly identified as negative samples while subjects were actually positive samples. Among them, the positive sample is successfully classified as the correct object category, which is represented as TP . In this study, the training samples are divided into two categories: open eyes and closed eyes for YOLO models training. For the open eyes category, TP means the eyes correctly identified as open eyes while the eyes were actually open; for the close eyes category, TP means the eyes correctly identified as close eyes while eyes were actually close. Otherwise, the negative sample is incorrectly predicted as the object category of the positive sample, which is called FP (False Positive). Table 6 presents

the number of positive and false positives for the YOLOv3 model in internal tests at different iterations.

The driver drowsiness detection and alert system hardware devices were installed in the vehicle cabin, and a well-trained YOLO object detection model is used to detect driver eye opening and closing. In this experiment, a system program was designed to record the driver's eyelid closing duration every time the driver blinked. The number of eye blinks per 15 seconds was calculated (i.e., the eye blink frequency). The driver's current eye status information, including the closed eyes duration and the number of eye blinks per 15 seconds, are displayed on the upper left of the system screen. The driver's

status is continuously monitored. When the driver is in a normal awake state, the system does not display a warning prompt. When the driver's eye closure lasts longer than 0.5 seconds in each eye blink, or the number of eye blinks is more than 3 times per 15 seconds (i.e., the eye blink frequency is too high) the vehicle driver is in a state of fatigue. The system immediately illuminates the LED warning light module through the current output from the NVIDIA Jetson TX2, and at the same time triggers a buzzer to sound an alarm. The warning prompt is displayed on the screen. The system screen is shown in Figure 10.

Table 4. Comparison of accuracy in every thousand iterations with different yolo model

Model Iterations	Average IoU (%)				mAP@0.50 (%)			
	v3	v3-tiny	v4	v4-tiny	v3	v3-tiny	v4	v4-tiny
1000	73.36	62.97	64.35	48.41	97.62	94.16	98.48	70.24
2000	84.06	79.36	83.55	77.82	99.98	99.61	99.98	99.74
3000	86.72	80.98	85.38	81.59	99.98	99.91	99.98	100.00
4000	86.76	82.64	85.92	82.46	99.98	99.96	99.98	100.00
5000	88.74	84.24	89.02	84.43	99.98	99.98	99.98	100.00
6000	89.11	83.97	90.30	84.30	99.98	99.96	99.98	100.00

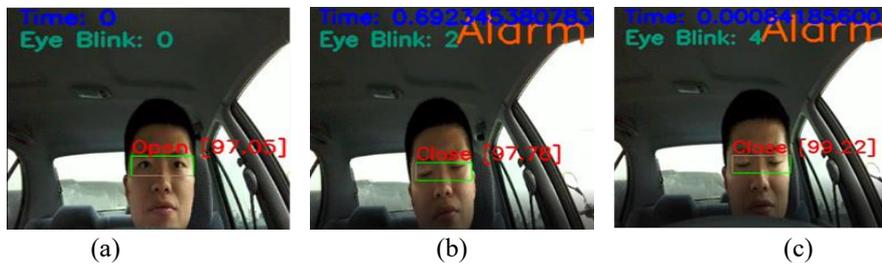


Figure 10. The system demonstration. (a) The driver is awake and the screen does not display a warning prompt; (b) Eye closure time is greater than 0.5 seconds, the screen displays a warning prompt; (c) Blink more than three times every 15 seconds, the screen displays a warning prompt

Table 5. Confusion matrix

Actual Class \ Predicted Class	True	False
	True	True Positive (TP)
False	False Negative (FN)	True Negative (TN)

Table 6. Comparison of true positive and false positive detections number in 3,000 and 6,000 iterations by yolov3 model

Iterations	Class name	TP	FP
3,000	Open	2506	0
	Close	2535	0
6,000	Open	2506	0
	Close	2535	0

5. CONCLUSION

A non-contact driver drowsiness detection and alert system was established in the vehicle cabin in this study. Computer vision object selection is used to detect whether the driver's eyes are open or closed. By processing the eye area of the input images, it is determined whether the driver is currently awake or fatigued while driving. When the system detects that the driver is in a state of fatigue, the system will simultaneously emit sound and light signals, to warn the driver that he must stop driving and take proper rest. The eye area of the input

images was labeled with these images used to train different YOLO object detection models, including YOLOv3, YOLOv3-tiny, YOLOv4 and YOLOv4-tiny. The research results show that YOLO object detection models have high accuracy and can realize real-time detection. The driver drowsiness detection and alert system established in this research can effectively prevent a vehicle driver from continuing to drive when fatigued. The risk of car accidents caused by fatigue driving is thereby reduced.

ACKNOWLEDGMENT

The study was supported by the Ministry of Science and Technology of Taiwan, Republic of China, under project number MOST 109-2221-E-018-013.

REFERENCES

- [1] World Health Organization. (2018). Global status report on road safety 2018: summary (No. WHO/NMH/NVI/18.20). World Health Organization. <http://doi.org/10.1136/ip.2009.023697>
- [2] Saradadevi, M., Bajaj, P. (2008). Driver fatigue detection using mouth and yawning analysis. International Journal of Computer Science and Network Security, 8(6): 183-188.

- [3] Meng, F., Li, S., Cao, L., Peng, Q., Li, M., Wang, C., Zhang, W. (2016). Designing fatigue warning systems: the perspective of professional drivers. *Applied Ergonomics*, 53: 122-130. <https://doi.org/10.1016/j.apergo.2015.08.003>
- [4] Vicente, J., Laguna, P., Bartra, A., Bailón, R. (2011). Detection of driver's drowsiness by means of HRV analysis. In *2011 Computing in Cardiology*, pp. 89-92. <https://ieeexplore.ieee.org/abstract/document/6164509>
- [5] Abbas, Q. (2020). HybridFatigue: A real-time driver drowsiness detection using hybrid features and transfer learning. *International Journal of Advanced Computer Science and Applications*, 11(1): 9. <https://doi.org/10.14569/IJACSA.2020.0110173>
- [6] Lal, S.K., Craig, A., Boord, P., Kirkup, L., Nguyen, H. (2003). Development of an algorithm for an EEG-based driver fatigue countermeasure. *Journal of Safety Research*, 34(3): 321-328. [https://doi.org/10.1016/S0022-4375\(03\)00027-6](https://doi.org/10.1016/S0022-4375(03)00027-6)
- [7] Rimini-Doering, M., Manstetten, D., Altmueller, T., Ladstaetter, U., Mahler, M. (2001). Monitoring driver drowsiness and stress in a driving simulator. In *First International Driving Symposium on Human Factors in Driver Assessment, Training and Vehicle Design*, pp. 58-63. <https://doi.org/10.17077/drivingassessment.1009>
- [8] Jap, B.T., Lal, S., Fischer, P., Bekiaris, E. (2009). Using EEG spectral components to assess algorithms for detecting fatigue. *Expert Systems with Applications*, 36(2): 2352-2359. <https://doi.org/10.1016/j.eswa.2007.12.043>
- [9] Wu, J.D., Chen, T.R. (2008). Development of a drowsiness warning system based on the fuzzy logic images analysis. *Expert Systems with Applications*, 34(2): 1556-1561. <https://doi.org/10.1016/j.eswa.2007.01.019>
- [10] Khunpisuth, O., Chotchinasri, T., Koschakosai, V., Hnoohom, N. (2016). Driver drowsiness detection using eye-closeness detection. In *2016 12th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS)*, pp. 661-668. <https://doi.org/10.1109/SITIS.2016.110>
- [11] Reddy, B., Kim, Y.H., Yun, S., Seo, C., Jang, J. (2017). Real-time driver drowsiness detection for embedded system using model compression of deep neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 121-128. <https://doi.org/10.1109/CVPRW.2017.59>
- [12] Arefnezhad, S., Samicee, S., Eichberger, A., Frühwirth, M., Kaufmann, C., Klotz, E. (2020). Applying deep neural networks for multi-level classification of driver drowsiness using Vehicle-based measures. *Expert Systems with Applications*, 162: 113778. <https://doi.org/10.1016/j.eswa.2020.113778>
- [13] Maior, C.B.S., das Chagas Moura, M.J., Santana, J.M.M., Lins, I.D. (2020). Real-time classification for autonomous drowsiness detection using eye aspect ratio. *Expert Systems with Applications*, 158: 113505. <https://doi.org/10.1016/j.eswa.2020.113505>
- [14] Ryan, C., O'Sullivan, B., Elrasad, A., Cahill, A., Lemley, J., Kielty, P., Posch, C., Perot, E. (2021). Real-time face & eye tracking and blink detection using event cameras. *Neural Networks*, 141: 87-97. <https://doi.org/10.1016/j.neunet.2021.03.019>
- [15] Krizhevsky, A., Sutskever, I., Hinton, G.E. (2012). ImageNet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems*, 60(6): 84-90. <https://doi.org/10.1145/3065386>
- [16] Redmon, J., Divvala, S., Girshick, R., Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 779-788. <https://doi.org/10.1109/CVPR.2016.91>
- [17] Ahmed, J., Li, J.P., Khan, S.A., Shaikh, R.A. (2015). Eye behaviour based drowsiness detection system. In *2015 12th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP)*, pp. 268-272. <https://doi.org/10.1109/ICCWAMTIP.2015.7493990>
- [18] Deng, W., Wu, R. (2019). Real-time driver-drowsiness detection system using facial features. *IEEE Access*, 7: 118727-118738. <https://doi.org/10.1109/ACCESS.2019.2936663>
- [19] Schleicher, R., Galley, N., Briest, S., Galley, L. (2008). Blinks and saccades as indicators of fatigue in sleepiness warnings: looking tired? *Ergonomics*, 51(7): 982-1010. <https://doi.org/10.1080/00140130701817062>
- [20] Wang, Y., Toor, S.S., Gautam, R., Henson, D.B. (2011). Blink frequency and duration during perimetry and their relationship to test-retest threshold variability. *Investigative Ophthalmology & Visual Science*, 52(7): 4546-4550. <https://doi.org/10.1167/iovs.10-6553>