Drowsiness Eye Detection using Convolutional Neural Network

Heru Arwoko[1], Susana Limanto[1*], Endah Asmawati[1]

[1] Universitas Surabaya, Jln. Raya Kalirungkut, Surabaya, Indonesia
heru_a@staff.ubaya.ac.id, susana@staff.ubaya.ac.id*,
endah@staff.ubaya.ac.id

Abstract. Eye fatigue while driving can cause drivers to be drowsy and less alert, which can potentially increase the risk of an accident. Existing data shows that the number of accidents in the world is increasing from year to year. One of the most common causes of accidents is fatigue and the leading cause of death is car accidents. Therefore, efforts are needed to reduce accidents due to fatigue. To overcome this, in this study, a system was developed to detect driver eye fatigue using the Convolutional Neural Network method with varying image sizes as input. The dataset consists of 1289 facial images that contain the eyes and is divided into 614 drowsiness eyes and 675 non-drowsiness eyes. In dealing with variations in image size, scaling was carried out using five interpolation methods, namely nearest-neighbor, bilinear, bicubic, inter-area, and lanczos4. The performance of the sleepy eye detection model will be evaluated based on accuracy and processing time. The results show that the image size of 64×64 with bilinear interpolation and 96×96 with inter-area interpolation gives the highest accuracy of 99%. Based on processing time, resizing the image to 8×8 size by using bilinear, bicubic, inter-area, and lanczos4 interpolation, results in the fastest processing time and high accuracy of 94% - 95%. The difference in accuracy with other image sizes is only 5%, with processing time for other size images up to 200 times longer than processing time for 8×8 image sizes.

Keywords: Drowsiness, Eye Fatigue Detection, Convolutional Neural Network, Interpolation.

1. Introduction

The eye is one of the five senses that helps humans in carrying out daily activities. Without the help of the eyes, some activities cannot be carried out, such as driving a vehicle. However, as one of the organs of the body, the eyes can experience fatigue, so they need rest. Eyes that are tired if forced to move will cause decreased concentration and can even interfere with eye health.

Someone who drives a vehicle needs the help of the eyes to know the situation of the road being traversed so that they can move properly. However, driving for too long, hot weather, or the situation on the road being traversed can cause the eyes to

M. Hartono et al. (eds.), *Proceedings of the 4th International Conference on Informatics, Technology and Engineering 2023 (InCITE 2023)*, Atlantis Highlights in Engineering 21,
https://doi.org/10.2991/978-94-6463-288-0_54

become tired so that the driver becomes drowsy. This condition causes the driver to be less concentrated and alert so prone to accidents [1], [2]. Accidents can result in motorists or other people experiencing injuries, material losses, being unable to return to normal activities, and can even cause death [3]. Existing data shows that the number of accidents in the world is increasing from year to year [4], [5], Indonesia is no exception [6]. In 2020, the number of traffic accidents in Indonesia reached 100,028 and in 2021 it increased to 103,645 [7]. One of the most common causes of accidents is fatigue [4] and the leading cause of death is car accidents [5]. Therefore, efforts are needed to reduce accidents due to fatigue. According to data statistics, nearly 1.3 million people have road traffic accidents every year, one of which is due to drowsiness [3], and Traffic Safety Foundation of the American Automobile Association report that 16-21% of traffic accidents are caused by driver fatigue [8].

Various studies have been developed to detect driver sleepiness based on changes in driver behavior, such as eye movements, mouth, head, and facial expressions with the help of cameras and image or video processing techniques [6]. Sleep detection based on whether the eyes are open or closed was developed by [5]. The determination of open or closed eyes is done using the Eye Aspect Ratio (EAR), which calculates the ratio of the distance between horizontal and vertical eye landmarks. The detection results will be used to sound an alarm. If the driver is still detected with eyes closed after 50 alarms, the system will send an SMS and email to the driver's family members containing a photo, the driver's location, and a message that the driver is sleepy. The same research was conducted by [3] by using the Convolutional Neural Network (CNN) and [9] by using Deep CNN. Reference [10] also doing research to detect open or closed eyes but considering different resolution variations and lighting conditions. The research was conducted using images that have been scaled to a size of 224×224 pixels using a deep residual convolutional neural network.

Drowsiness, defined as a state when a person needs rest to sleep, can cause major accident hazards while on task, delayed response time, lack of awareness, or microsleeps [2]. A drowsiness detection system based on the geometric features of the eyes and mouth was developed by [4]. Eye-based drowsiness detection is done using EAR. While yawning detection is done by calculating the distance between the lower lip and upper lip and then compared with a predetermined threshold value. The level of drowsiness obtained is used to sound the alarm. Research yields close to 100% accuracy when the image is in the right position and not wearing anything.

Research conducted by [2] detects driver fatigue from a series of images with a duration of 60 seconds. Before use, the resulting image is scaled so that it has a size of 64×64 pixels. After being scaled, the images are processed using two methods, namely: CNN and a combination of AI and CNN. The second method [2] uses a combination of linear SVM and histogram of oriented gradients (HOG) to detect the location of the driver's face. The accuracy of the research results from the two methods did not differ, namely: around 65% for training data and more than 60% for test data. However, the second method has advantages, namely it can work continuously without disturbing the driver when he is not sleepy because the system only sounds an alarm incorrectly once out of 60 videos.

In this study, the detection of drowsiness eyes will be carried out using the CNN method. This method was chosen because the classification ability of this method in image recognition is very good [10]. The model will be developed using various image sizes. Image size scaling will be performed using five different interpolation methods to reduce the possibility of loss of information. The use of interpolation for image scaling is an advantage of this study. The performance of the sleepy eye detection model will be evaluated based on accuracy and processing time. This research is part of a larger study that focuses on developing methods to detect generally sleepy eyes based on real-time data captured by the camera. What is meant by "in general" is not limited to vehicle drivers, but can also be used for other activities, such as detecting drowsiness in people at a meeting.

2. Materials and Methodology

The research methodology consists of five stages, namely: dataset collection, preprocessing, classification model, training and testing, and performance evaluation. Fig. 1 shows the flow of the research methodology used in this study.

2.1 Workflow

Eye fatigue detection is important for rider safety. For this reason, it is necessary to develop an intelligent system that can detect drowsiness accurately and quickly. In this research, eye fatigue was detected using the Convolutional Neural Network (CNN). Accuracy and convergence of CNN process results are determined by the setting hyperparameters, including the number of epochs, momentum, and learning rate are the most common hyperparameters [11]. In this study, the CNN accuracy performance in detecting drowsiness was increased from different points of view. The CNN process requires a uniform input image [12]. The existing input images are not always uniform and vary widely. So, it is necessary to transform the image into a fixed size for the CNN process. To resize an image from a large size to a smaller size raises the significant loss of image pixel information. The image needed for the smoothing process through the interpolation method [13], as shown on the process of flow diagram in Fig. 1.

In this research, the optimization of the CNN process was used by changing the size of the image. In the first step, dataset collection is carried out. This dataset has image sizes that vary from 206×162 to 255×200. In this study, the size was made uniform to 8×8, 16×16, 32×32, 64×64, 96×96, 128×128, 160×160, and 192×192. Meanwhile, the interpolation process uses five methods, including nearest-neighbor, bilinear, bicubic, inter-area, and lanczos4 interpolation. Then measure the image quality using the Peak Signal-to-Noise Ratio (PSNR), which is an expression for the ratio between the maximum intensity value of the image and the noise strength that affects its representation.

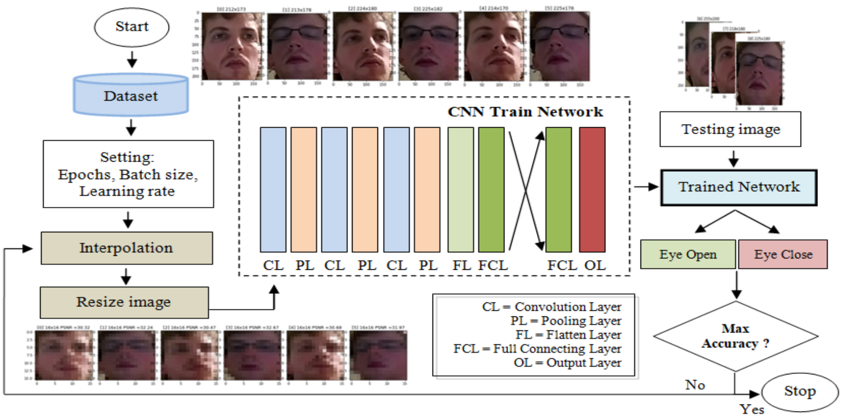


Fig. 1. Flow Diagram of Drowsiness Detection.

2.2 Dataset

Sleepy and not sleepy face images are collected from the public dataset available repositories from http://vlm1.uta.edu/ and were used in this work. The dataset consists of a total of 1289 facial images, which were divided into 614 sleepy faces and 675 not-sleepy faces. Image sizes vary from 206-255 pixels in height, 162-200 pixels in width. The average shape image is 231 × 181 × 3. Next, the image must be set to the same resolution for CNN processing.

2.3 Image Interpolation

Interpolation is a method of generating new data points within a range of discrete sets. The most widely used interpolation techniques for vision computers are nearest-neighbor, bilinear, bicubic, inter-area, and lanczos4. In general, to enlarge an image, it is better to use bilinear or bicubic interpolation, otherwise for shrinking the image, it is better to use inter-area interpolation.

Nearest-neighbor interpolation is a simple method because it replaces the new pixel value with the nearest neighbor. In addition, the method nearest-neighbor does not have calculations. This pixel replication will replace the existing pixel value which repeats the predetermined pixel value according to the desired magnification size.

Bilinear interpolation, new pixel values based on the weighted average of the 4 pixels nearest neighbors $2x2$ pixels in the image original [13]. Bilinear interpolation determines a new pixel value based on the average weight of 4 pixels at the level of 2 × 2 neighbors in the original image.

Bicubic interpolation is an interpolation method that uses 16 pixels in the nearest neighboring 4 × 4 pixels in the original image. Using this bicubic interpolation method can make the edges of the resulting image smoother [14].

Inter-area interpolation is like the Nearest neighbor method when downsampling the image. Inter-area does better at thinning the image and avoids spurious inference patterns in the image.

Lanczos4 interpolation over an 8×8 pixel neighborhood. It performs the same task as the bicubic interpolation method, which is slow and resource intensive.

2.4 Peak Signal-to-Noise Ratio

The quality of the resized image compared to the original image using Peak Signal to Noise Ratio (*PSNR*). The calculation of *PNSR* requires Mean Square Error (*MSE*). Let image A(x, y) is the intensity function of the original image and image B(x, y) is the intensity of the transformed image, both the same size as $M \times N$. Where x and y are spatial domains of the image. Then, *MSE* is calculated using (1). *MSE* is calculated calculated pixel by pixel by adding up the squared differences of all pixels and is divided by the total number of pixels $M \times N$.

$$MSE = \frac{1}{MN} \sum \sum [A(x, y) - B(x, y)]^2 \tag{1}$$

If H is the range of the pixel values, then *PSNR* is calculated using (1). The higher of the *PSNR* value, the higher the image quality.

$$PSNR = 10\, Log\, 10\, [H^2/MSE] \tag{2}$$

2.5 The Architecture of Proposed CNN

The proposed CNN model contains three convolution blocks followed by a fully connected layer and an output layer (see Fig. 2). The CNN architecture consists of an input layer, convolution layer: conv-1, conv-2, conv-3, and each at the end with a max pooling layer. At the end of the network section is a full-connecting layer and output layers. The convolution layer of CNN is responsible for extracting features from input images using some convolutional filters. Convolution layer contains the weights that must be optimized using gradient descent training. The features extracted from convolution layer are mapped into feature space using the nonlinear Rectified Linear Units (ReLU).

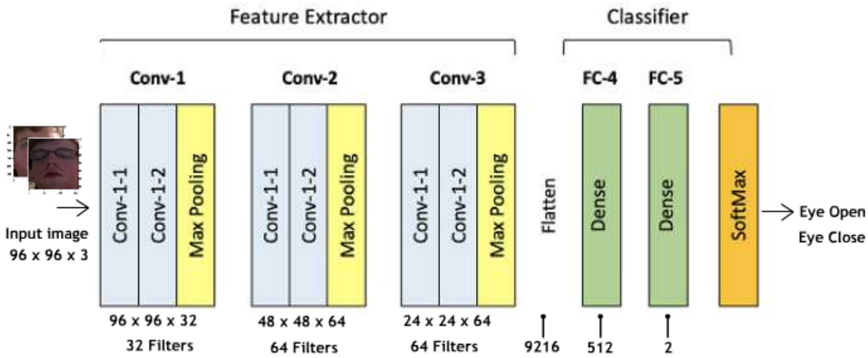


Fig. 2. Architecture of Proposed CNN.

This input image is convoluted with 32 filters of 3×3 size, so the shape of the image becomes $96 \times 96 \times 32$. Then the end of the first convolution is given max pooling with a size of 2×2 so every image size is reduced to 48×48. In the second convolution the shape of the image becomes $48 \times 48 \times 64$ and followed by a 2×2 max pooling so that the image size becomes $24 \times 24 \times 64$ in the third stage of convolution. In the last convolution process after max pooling, the image shape is $12 \times 12 \times 64$, so during the flattening process, $12 \times 12 \times 64 = 9216$ extraction features are obtained. The next step is the multi-layer perceptron (MLP) process by providing 512 hidden neurons in layers FC-4 and 2 neurons FC-5 as the final network target. One characteristic of a drowsy driver is the driver's eyes close for a few moments. So, in this research, the output to be used are eyes open and eyes closed.

3. Result and Discussion

The main objective of this study is designed to get high accuracy in the detection of drowsiness and the required process quickly. So, it is necessary to do experiment to get the optimal input image size for the optimal CNN model. The CNN model in Fig. 1 is implemented through a series of experiments using the hardware specification, Intel(R) Core(TM) i7-7700HQ CPU model, 2.80 GHz main frequency, 8G memory, and GTX1050 GPU.

In our experiment, the TensorFlow Keras platform was used to build a convolutional neural network. The initial values for the CNN training parameters were 20 epochs, 30 batch sizes, 1 stride, same padding, 0.3 validation split, and 0.001 for learning rate. There were 1289 images in the dataset divided into 90% for training and 10% for testing. So for the training data with shape (1160, height, width, 3) and the remaining for the testing data with shape (129, height, width, 3). The height and width of image size vary from 8×8, 16×16, 32×32, 64×64, 96×96, 128×128, 160×160, 192×192. Model performance was measured based on accuracy and time required for the training and testing process.

Table 1 shows the accuracy of the test results with the CNN method for 8 image sizes and five interpolations. The test results for image sizes of 16×16 and above with various interpolation methods yield an accuracy of between 95% - 99%. The best test results were obtained for images measuring 96×96 with an average accuracy of 98.2%. While the best interpolation method is obtained when bicubic interpolation is used with an average accuracy of 96.8%. The highest accuracy, 99%, is obtained when using 64×64 images with bilinear interpolation and 96x96 images with interpolation between areas.

Table 2 shows the time required to obtain classification results using the CNN method with a variety of image sizes and interpolation methods. Based on the average use of processing time, it appears that the larger the image size, the more processing time is required. The average accuracy of all image sizes is above 88%. Although each 8×8 image interpolation method produces the smallest accuracy compared to the others, the processing time is much faster. For 96×96 image size which is the best size, processing time is 49 times longer than the processing time of 8×8 image size.

Based on the accuracy results in Table 1 and the processing time in Table 2, for each interpolation method, resizing the image to an 8×8 size using an interpolation other than the nearest produces the fastest processing time with a high level of accuracy. The difference in accuracy with other image sizes is only 5%, whereas the processing time for other sizes can be up to 200 times the processing time of 8×8 images. Table 3 shows the resized image in several sizes. Even though the results of resizing to size 8×8 are not very clear, the accuracy results are good.

Table 1. Accuracy model for image size - interpolation methods.

Image size	Accuracy					
	Nearest-neighbor	Bilinear	Bicubic	Inter-area	Lanczos4	Average
8×8	0.88	0.94	0.93	0.94	0.93	0.924
16×16	0.95	0.96	0.96	0.95	0.96	0.956
32×32	**0.98**	0.96	**0.98**	0.98	**0.98**	0.976
64×64	0.97	**0.99**	**0.98**	0.98	**0.98**	0.98
96×96	**0.98**	0.98	**0.98**	**0.99**	**0.98**	**0.982**
128×128	**0.98**	0.96	0.96	0.96	0.95	0.962
160×160	0.95	0.97	**0.98**	0.97	0.97	0.968
192×192	0.96	0.97	0.97	0.95	0.97	0.964
Average	0.956	0.966	0.968	0.965	0.965	

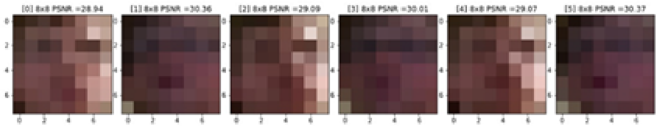
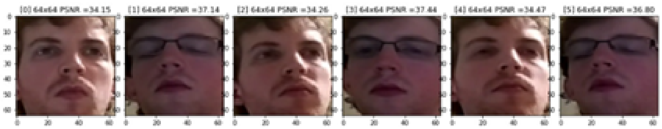
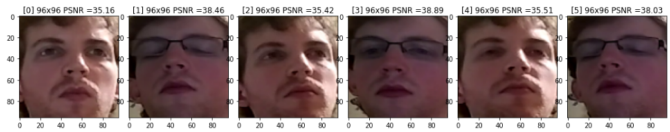
Table 2. Time processing (s) for image size - interpolation methods.

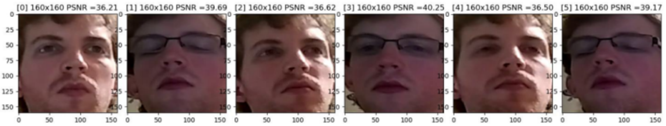
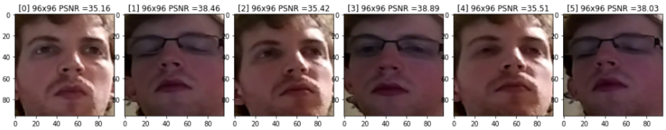
Image size	Time process (s)					
	Nearest-neighbor	Bilinear	Bicubic	Inter-area	Lanczos4	Average
8×8	8	9	10	9	9	9
16×16	18	20	19	19	19	19
32×32	56	58	56	57	58	57
64×64	191	183	189	190	194	189.4
96×96	437	446	453	443	439	443.6
128×128	749	751	761	761	759	756.2
160×160	1152	1156	1164	1181	1178	1166.2
192×192	1599	1607	1615	1654	1648	1624.6

For image sizes that are too small, the result is lower accuracy. It can be understood that for a small image size the noise value is high, and a lot of lost important information. Mean Squared Error (*MSE*) is a risk function that determines the average squared difference between the pixel value of the original image with the resized image. This can be seen in Fig. 3, that for a smaller image size, the *PSNR* value will decrease, and the *MSE* value will increase. Mean Squared Error is used to compare the pixel value of the original image to the degraded image. This means that the smaller the image size causes a decrease in image quality.

For image sizes that are too large, the accuracy results are stable. It is understood that for large image sizes, the important information from pixels can be retained. However, it can be seen in Fig. 4 that for large image sizes the processing time is much longer, this is not efficient for the safety purpose of detecting drowsiness in drivers in real-time.

Table 3. Image resizing results in various sizes.

Image size	Image
Original image	
8×8	
16×16	
32×32	
64×64	
96×96	
128×128	

160×160	
192×192	

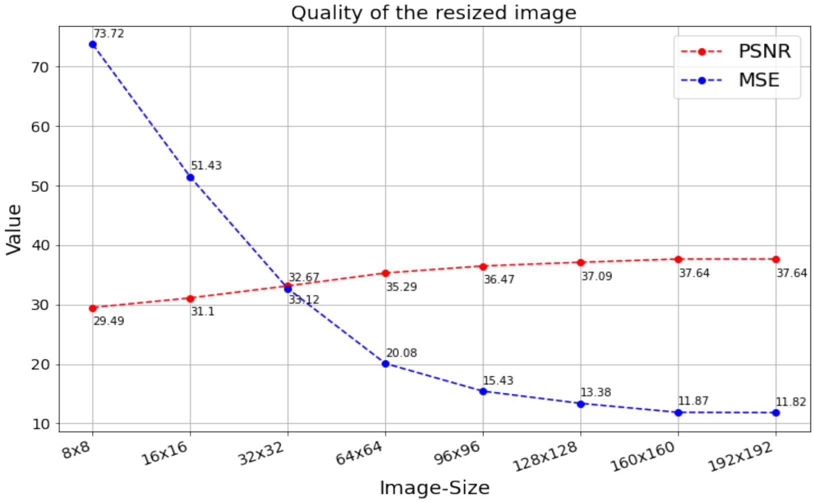


Fig. 3. PSNR and MSE for various image sizes.

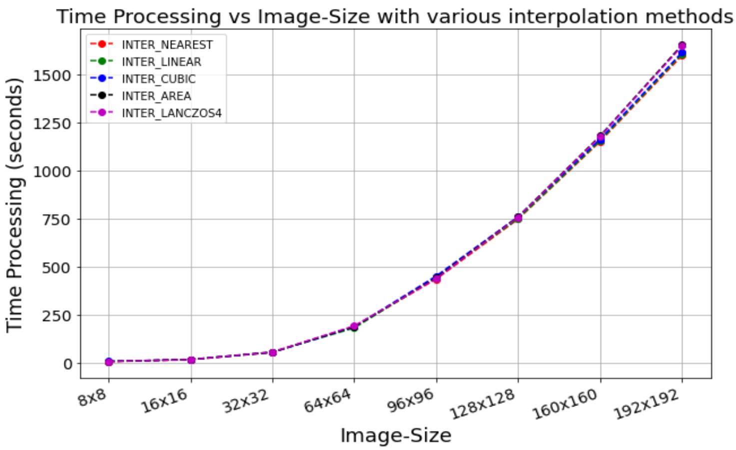


Fig. 4. Time processing for various image sizes.

4. Conclusion

In this study, a performance comparison of various image sizes and interpolation methods was carried out in detecting drowsiness eyes using the CNN method. Based on the results of a comparison of the four interpolation methods at 8 different image sizes, it was found that the optimal image size (highest accuracy) for sleep detection using CNN occurs at an image size of 64×64 for bilinear interpolation and an image size of 96×96 for inter-area interpolation. The interpolation that produces stable accuracy for all image sizes is the bicubic interpolation method. The larger the size of the image, the more time is used to process it, and increase significantly. For detecting drowsiness in drivers, it is necessary to pay attention to work points that require a relatively fast processing time. Resizing to an 8×8 image size produces the fastest processing time with an accuracy rate of 93% - 94% using the bilinear, bicubic, inter-area, and lanczos4 interpolation methods.

In future studies, the best model for detecting drowsiness eyes will be carried out using real-time data taken from the camera. The data used is not only for vehicle driving but also for other conditions. In future research, the performance of the CNN model will also be compared with the performance of other methods such as transfer learning (VGG16, Mobilnet, Resnet50, and others).

References

[1] Y. Albadawi, M. Takruri, and M. Awad.: A Review of Recent Developments in Driver Drowsiness Detection Systems. Sensors 22 (5), 1–41 (2022).

[2] E. Magán, M. P. Sesmero, J. M. Alonso-Weber, and A. Sanchis.: Driver Drowsiness Detection by Applying Deep Learning Techniques to Sequences of Images. Appl. Sci. 12 (3), 2022.

[3] S. Sheela, D. E. Jothika, N. Swarnapriya, and B. R. Vishali.: Drowsiness detection using CNN. in Advances in Parallel Computing, vol. 38, pp. 575–579, IOS Press Ebooks, 2021.

[4] M. Jain, B. Bhagerathi, and S. C N.; Real-Time Driver Drowsiness Detection using Computer Vision. Int. J. Eng. Adv. Technol. 11 (1), 109–113 (2021).

[5] S. Titare, S. Chinchghare, and K. N. Hande.: Driver Drowsiness Detection and Alert System. Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol. 7 (3), 583–588 (2021).

[6] V. Triyanti and H. Iridiastadi.: Challenges in detecting drowsiness based on driver's behavior. in IOP Conference Series: Materials Science and Engineering, 2017, vol. 277, no. 1.

[7] Statistics Indonesia Homepage, Traffic Accident, Killed Person, Seriously Injured, Slight Injured and Expected of Material Losses Value 2019-2021. https://www.bps.go.id/indicator/17/513/1/jumlah-kecelakaan-korban-mati-luka-berat-luka-ringan-dan-kerugian-materi.htmlhttps://www.bps.go.id/indicator/17/513/1/jumlah-kecelakaan-korban-mati-luka-berat-luka-ringan-dan-kerugian-materi.html, last accessed 2023/06/17.

[8] Z. Zhao, N. Zhou, L. Zhang, H. Yan, Y. Xu, and Z. Zhang.: Driver Fatigue Detection

Based on Convolutional Neural Networks Using EM-CNN. Comput. Intell. Neurosci. 2020 (3), 2020.

[9] V. R. Reddy Chirra, S. R. Uyyala, and V. K. Kishore Kolli.: Deep CNN: A Machine Learning Approach for Driver Drowsiness Detection Based on Eye State. Rev. d'Intelligence Artif. 33 (6), 461–466 (2019).

[10] K. W. Kim, H. G. Hong, G. P. Nam, and K. R. Park.: A study of deep CNN-based classification of open and closed eyes using a visible light camera sensor. Sensors (Switzerland) 17 (7), 2017.

[11] T. Goel, R. Murugan, S. Mirjalili, and D. K. Chakrabartty.: OptCoNet: an optimized convolutional neural network for an automatic diagnosis of COVID-19. Appl. Intell 51 (3), 1351–1366 (2021).

[12] H. He *et al.*: A Real-time Driver Fatigue Detection Method Based on Two-Stage Convolutional Neural Network. IFAC-PapersOnLine 53 (2), 15374–15379 (2020).

[13] B. K. Triwijoyo and A. Adil.: Analysis of Medical Image Resizing Using Bicubic Interpolation Algorithm. J. Ilmu Komput. 14 (1), 20 (2021).

[14] S. Fadnavis.: Image Interpolation Techniques in Digital Image Processing: An Overview. J. Eng. Res. Appl. *www.ijera.com* 4 (10), 70–73 (2014), [Online]. Available: www.ijera.com.