Digital Eye Strain Detection By Applying Deep Learning Technique To Categories Of Images

Dr. Deepti Sharma[1]*, Dr. Archana B. Saxena[2], Dr. Deepshikha Aggarwal[3]

[1*, 2, 3]Department of Information Technology, Jagan Institute of Management Studies, Sector-5, Rohini, Delhi, India.

***Corresponding Author: -** Dr. Deepti Sharma
*Department of Information Technology, Jagan Institute of Management Studies, Sector-5, Rohini, Delhi, India.

Abstract
This work presents the development of an ADES (advanced digital eye strain) detector system. It focuses on digital eye strain detector whose objective is to alert the users of digital devices to alert them if they are using devices for a long period of time. If a user is working on digital devices for prolonged hours, it is necessary that fatigue detection is performed in a non-intrusive way, and that the user should be alert with alarms. Our approach to this open problem uses sequences of images that are 60 s long and are recorded in such a way that the user's face is visible. To detect whether the user shows symptoms of tiredness or not, a solution is developed focusing on the minimization of false positives. This solution uses a recurrent and convolutional neural network to extract numeric features from images. The accuracy obtained by the proposed system is similar: around 85% accuracy over training data, and 80% accuracy on test data.

Keywords: ADES; drowsiness; deep learning; convolutional neural networks; computer vision syndrome

1. Introduction

Eyestrain is a frequent condition that occurs when your eyes tyre after being used for an extended period of time, such as when driving a long distance or staring at screens of computers and other electronic devices [1] . In essence, eye strain is just tired eyes. Eye tiredness is another name for it. The causes of digital eye strain are varied [2]. According to studies, people blink less when using computers or other digital screens [3] [4]. Dry eyes result from this, which may increase eye fatigue. Due to their glare or when there is insufficient contrast between the type and the background, digital devices can also strain the eyes. Eye strain can also be caused by an improper viewing distance and bad illumination. Eye strain typically manifests as watery eyes, dry eyes, blurred vision, sensitivity to light, headache, neck and shoulder pain, difficulty concentrating, burning eyes, itchy eyes and hard time keeping your eyes open [5].

Digital eye strain, dry eyes, irritation, and discomfort can be brought on by squinting to read small print, poor lighting, holding devices at an awkward angle or too far away from our eyes, blue light emitted by digital devices, or wearing eyeglasses that are not designed for viewing computers and other electronics up close [1]. Limiting activities that strain the eyes as well as altering one's lifestyle might help control eye strain. Digital eye strain can be treated in a variety of ways. First and foremost, screen time needs to be moderated and frequently interrupted. There are additional strategies to mitigate the negative effects of screen time on your eyes if you are unable to cut back on your gadget use. Reading from a printed page puts less strain on the eyes than using a computer because people tend to blink less when using a computer, and blinking is essential for keeping the eyes moist. See digital screens from unfavourable angles or distances [6]. The majority of the time, symptoms of digital eye strain appear when a task requires more visual clarity than the person's eyes can easily accommodate. Those who use a computer or other digital screen device for two or more hours every day are most at risk for getting digital eye strain.

2. Related Work

[1]Authors have discussed what prime causes of DES in great depth. Every symptom and association with the DES explained very well using medical terminology. The paper explained, what all measures users should adopted to identify DES condition at early stage. The differences how digital study material and printed study material effect our eye sight explained very well.

[2] A study was done on 350 people of various ages to validate the theoretical ideas about how excessive usage of digital gadgets causes DES or CVS. The study's main focus was on the participants' eye health. A questionnaire is used to gather the information. A carefully crafted questionnaire was created to gather data regarding the respondents' interactions with their digital devices, as well as the devices' types, lifespans, and intended uses. Data obtained by various methods and evaluated by various Analytical techniques are used to draw conclusions.

[3] An extensive study of graduate and postgraduate students examined the impact that excessive usage of digital devices throughout the years 2018 and 2019 had on ocular and related health conditions.

[4] In order to determine whether a user is experiencing eye strain while using a digital gadget, authors have used machine learning techniques. Monitoring numerous traceable and non-traceable parameters allows for the establishment of the correlation. These elements will serve as the decision tree algorithm's input. Through the creation of a decision tree, several rules are established, and these rules will serve as indicators of whether a user has CVS or not. By an LED light installed on the user's machine, the same is transmitted to the user.

[5] It is based on research done during the COVID-19 era and the application of technology at that time. The effect of these numerous digital devices on users' vision and eye strain has been assessed by TAM (Technology Adoption Model) in several nations.

3. System Implementation

ADES (advanced digital eye strain) are a type of safety equipment that can stop eye disorders from happening to those who use digital devices for extended periods of time. The goal of this research is to create a detection system that can recognise when someone's eyes are closed for a brief period of time [7].

For gathering webcam photos and feeding them into a Deep Learning model that would categorise whether the user's eyes are "Open" or "Closed," the creators of this work employed the OpenCV python package [8]. The following strategy was employed by the writers for this study project:

Step 1: Take a picture from a camera using a webcam as input in step one.
Step 2: Create an Area of Interest in Step 2 by identifying the face in the image (ROI).
Step 3: Use the ROI to find the eyes and provide that information to the classifier.
Step 4: The classifier will classify whether the eyes are open or closed in step four.
Step 5: Count the instances that someone closes their eyes.
Step 6: Compute the score in Step 6 to see if the user is experiencing eye strain.
Step 7 – Alert the user with Alarm.

3.1 Advanced Digital Eye Strain Detector System Dataset

The dataset utilised for this model was created by the authors themselves. A script is built that records the eyes from a web camera and saves them to the local disc in order to construct the dataset. Pictures are divided into their appropriate labels and the labels "Open" and "Closed" are produced [9]. The undesired photos that were not required for creating the model were manually removed from the data. About 6000 photos of people's eyes in various lighting environments are included in the data. The final weights are documented and a model architecture file is prepared after the model has been trained on the dataset. The classification of whether a user's eye is open or closed can be done afterwards using this model. An alert will be sent if it is closed for the specified threshold [10].

3.2 The Model Architecture

Convolutional Neural Networks (CNN) are useful for image recognition, image classification and computer vision (CV) applications because they provide highly accurate results, especially when a lot of data is involved [11]. In this study, convolutional neural networks are used to build a model with Keras (CNN). Convolutional neural networks are a specific variety of deep neural networks that excel at classifying images [12]. As the object data passes through the CNN's numerous layers, it also picks up the features of the item over time. In essence, a CNN is made up of three layers: an input layer, an output layer, and a hidden layer with potential for more layers. These layers are put through a convolution operation with a filter that multiplies their 2D matrices together.

3.3 Research Model Prerequisites

The requirement for this research project is a webcam through which we will capture images, Python installed on the system. Following necessary packages must be installed on the machine.

1. **Open CV –** it will be used for face and eye detection.
2. **Tensor Flow –** keras uses TensorFlow as backend.
3. **Keras –** it will be used to build our classification model.
4. **Pygame –** to play alarm sound.

After installation, following steps will be used for performing digital eye strain detection.

Step 1 – Take Image as Input from a Camera

The first step is to capture the images using webcam. In this work, authors have created an infinite loop to capture each frame of pics inputted from the webcam. cv2.videocapture(0) is a method provided in OpenCV library which is used to access the camera and set the capture object. This method is used in this work to access webcam and then read function is used to read each frame and to store the image in a frame variable.

Step 2 – Detect Face in the Image and Create a Region of Interest (ROI)

After images are captured, the next step is to detect the face in the image and then to create a region of interest (ROI). For this, there is a need to first convert the image into grayscale. As the OpenCV algorithm for object detection takes gray images in the input, there is no need of color information to detect the objects [13]. Here, authors have used haar cascade classifier to detect faces. After that, faces are detected using face. detect Multi Scale(gray) method. This function returns an array of face detections with x,y coordinates, and height, the width of the boundary box of the object. Now iteration is possible over the faces and draw boundary boxes for each face.

Step 3 – Detect the eyes from ROI and feed it to the classifier

The next step is to detect eyes from the captured face. The method for detecting eyes is the same as that for detecting faces. In order to detect the eyes using leye, the cascade classifier is first established for eyes in leye and reye, respectively. Detect Multi Scale(gray). It is now necessary to merely extract the eyeballs' info from the complete image. This may be done by removing the eye's boundary box, after which we can use this code to extract the eye's picture from the frame. Only the eye's image data is contained in l_eye. This information will be used by our CNN classifier to determine whether the eyes are open or closed. The right eye will also be extracted and stored in r_eye.

Step 4 – Classifier will Categorize whether Eyes are Open or Closed

In this work, CNN classifier is used for predicting the eye status. This status will then be fed into the model. The model needs the correct dimensions to start with so there is a need to perform certain functions. Initially, we use cv2.cvtColor(r eye, cv2.COLOR BGR2GRAY) to convert the colour image to grayscale. The image is then scaled down to 24 * 24 pixels because our model was trained on 24 * 24 pixel images using the function cv2.resize(r eye, (24,24)). To improve convergence, we standardise our data. r_eye/255 (All values will range from 0 to 1). Boost the dimensions to add to our classifier's input. Now, using model.predict classes(l eye), we forecast each eye. When lpred[0] = 1, it indicates that the eyes are open, and when lpred[0] = 0, it indicates that the eyes are closed.

Step 5 – Calculate Score to Check whether Person is Drowsy

After eyes prediction is done and it is decided that eyes are open or closed, the next step is to calculate the score to find whether a person is feeling drowsy or not. The score is essentially a number that will be used to calculate how long the subject has kept his eyes closed. Hence, if both eyes are closed, we will continue to increase score, and if one eye is open, we will drop score. Using the cv2.putText( ) function, we are drawing the outcome on the screen to show the person's status in real time. A threshold is established, for instance, if the score exceeds 15, it indicates that the subject has been staring into space for a considerable amount of time. Low blink rates indicate eye strain in the person.
At this point, the alarm will sound. play ( )

4. Results and Analysis

To perform eye strain detection, a system is designed which captures images from web camera and detect face from those images. The eyes are detected using region of interest. After feeding it into classifier, the status will be declared as ‘open’ or ‘closed’. If eyes are closed till the score of 15, a sound will be alarmed which will alert user about the eye strain. Following figure shows the status of eyes (open or closed) and the value of score.

a) Fig1 shows the eyes are open and the value of score is 0.

Fig 1: Status: open and Score = 0

b) Fig2 shows the eyes are closed and the value of score is 1.

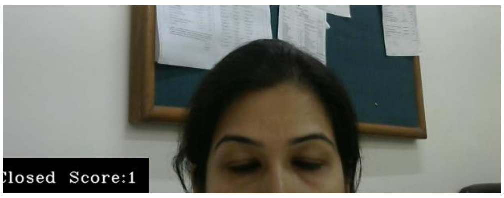

Fig 2: Status: closed and Score = 1

c) Fig 3 shows the eyes are closed and the value of score is 7.

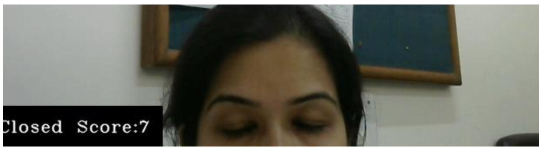

Fig 3: Status: closed and Score = 7

d) Fig 4 shows the eyes are closed and the value of score is 8.

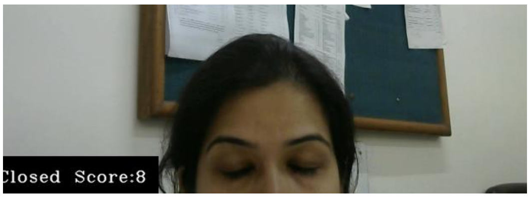

Fig 4: Status: closed and Score = 8

e) Fig 5 shows the eyes are closed and the value of score is 17 which is greater than 15. Red line at corner indicates the alarm glows to alert the user.

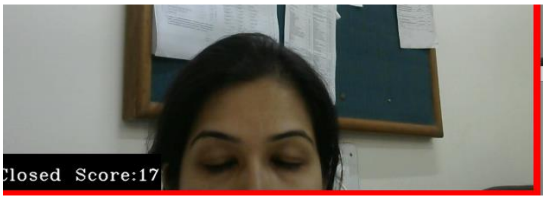

Fig 4: Status: closed and Score = 17

5. Summary

In this research work, we have built an ADES (advanced digital eye strain) detector system that can be implemented in numerous ways. Authors have used OpenCV to detect faces and eyes using a haar cascade classifier and then we used a CNN model to predict the status. When a person is feeling drowsy, he/she will be alerted and then he can do various things to reduce that strain. He can take break, follow 20-20-20 rule, blink his eyes more frequently, use suggested eye drops or can visit doctor in extreme case.

References:

1. D. S. a. A. B. S. D. Aggarwal, “Detection of eye strain due to usage of electronic devices,” International Journal of Health Sciences, vol. 6, no. Speial Issue, 2022.
2. D. A. a. A. B. S. D. Sharma, “A Contemporary Approach To Analyse The

Impact Of 8 Increased Screen Time On Individuals And Associated Issues," Indian Academic Research Association, vol. 18, no. 6, 2022.

3. D. S. a. A. B. S. D. Aggarwal, ""View of A Pragmatic Approach to the Usage of Digital 1 Devices in Education in Developing Countries," Turkish Journal of Computer and Mathematics Education, vol. 12, no. 13, 2021.
4. H. m. t. y. b. i. a. day?, "https://www.healthline.com," e. Healthline, 2022. [Online]. Available: https://www.healthline.com/health/how-many-times-do-you-blink-a-day.
5. D. S. a. D. A. A. B. Saxena, "CVS Identification through Live streaming using 9 Machine Learning: An Elaborative Framework," Turkish Online Journal of Qualitative Enquiry, vol. 12, 2021.
6. www.researchgate.net, "Average-time-per-day-using-digital-devices-when engaging-in-other-activities," researchgate.net, 2015.
7. W. r., " OpenCV, Face Detection using Haar Cascades," :https://docs.opencv.org/3.4/d2/d99/tutorial_js_face_detection.html, 2021.
8. P. R. A. Zoroofi, Open/Closed Eye Analysis For Drowsiness Detection.
9. S. Pouyanfar, S. Sadiq, Y. Yan, H. Tian, Y. Tao, M. Reyes, M. Shyu, S. Chen and S. Iyengar, "A Survey on Deep Learning: Algorithms, Techniques, and Applications," ACM Comput. Surv., vol. 51, 2018.
10. M. A. A. K. J. R. L. Hsu, Face Detection In Color Images.
11. S. R. A. P. T. B. Schlkopf-A.Blake, Computationally Efficient Face Detection.
12. R. D. A. P. E. Hart, Use Of The Hough Transformation To Detect Lines And Curves InPicture.
13. [Online]. Available: Http://Dlib.Net/Face_Landmark_Detection_Ex.Cpp.Htm.
14. . S. Watson, "Eye Health," 17th Aug 2019. [Online]. Available: https://www.webmd.com/eye-health/computer-vision-syndrome.
15. e. Healthline, "How many time you blink in a day?," [Online]. Available: https://www.healthline.com/health/how-many-times-do-you-blink-a-day. [Accessed 2nd August 2022].
16. "Image_devices used for different purposes," [Online]. Available: https://www.researchgate. net/figure/Average-time-per-day-using-digital-devices-when-engaging-in-other-activities-in-2015_fig2_321068001. [Accessed 2nd Sep 2022].