Blink Detection to Control Display Light

Dandala Bhagath Reddy
Department of Computer Science Engineering
Apex Institute of Technology
Chandigarh University
Punjab, India
iambhagath@gmail.com

Prince Jain
Department of Computer Science Engineering
Apex Institute of Technology
Chandigarh University
Punjab, India
princeece48@gmail.com

***Abstract*— A cross-platform application to control screen light to turn on/off based on whether his eyes are open or not. People who spend a lot of time in front of screens, there's a high chance they've computer vision syndrome. They suffer from various eye problems, the major one being dry eyes cause dropped blinking rate. This is a background running application useful to automatically turn off-screen while the user will close his eyes, immediately turn on-screen while he opens his eyes and keep the screen-on while no eyes are detected. It ignores the casual blinks without resulting in screen flickering.**

Keywords*— *Awake Detection, Blink Detection, Control Display Light On/Off, Computer Vision, Screen Brightness.

I. INTRODUCTION

People who spend a lot of time in front of screens, there's a high chance they've computer vision syndrome (computer eye strain) [1], suffer from various eye problems like eye fatigue, headache, blurry vision, tearing or red eyes major one being dry eyes cause dropped blinking rate [1, 2]. After watching desired content people unintentionally end up continuously looking at useless content [3, 4], as all know it is important to know what is important and also important to know what is not important. While observing too many sharp, tiny and high contrast details we tend to not close eyes, not to miss those details, when you blink less the liquids protecting eyes quickly get evaporated, resulting in dry eyes. It is important to Blink because it keeps your eyes moisturized [1, 2, 5]. The popular many believe that eye problem arises due to blue light coming out of the screen but the main reason behind this is you do not close your eyes while you are looking at the screen light [6]. According to the research of University of Lowa Hospitals, a person blinks up to 66 % less frequently while using a computer and 3 times less blinking rate while looking at the screen. According to the research of Optometry (Clinical and experimental) the average blink rate and a computer user's average blink rates are 17 blinks/min and 6.5 blinks/min respectively [7, 8], the accommodative method causes blur vision, double-vision, presbyopia, myopia and slow focal change. In a study, it was reported that transient-myopia, on an observation 20% of computer users at the end of their work shift [9].

The average blink takes 400ms [10], but to avoid screen flickering in between, so we put an experimented constraint above 1.8 sec this could be a minimal constraint for high-performing systems, it is recommended to keep the constraint higher than 2.5 in the range of 6.5 sec. You must modify the constraint like; lower the computer performance (GPU) higher the constraint it should be maintained to avoid human-computer interaction lag in between user and system considering the performance. This is made as background running application that works on any window that you will work on It can display a notification while the user has not blinked like in the last 13 seconds you will get a reminder message to blink and protect your eyes. This motivates people to get rest and relax in between long hours over screen watching time and gives users the freedom to take a few naps in front of the screen.

To an extent, this will mostly protect user's privacy by making the content is invisible to others and this is also an energy saving method to reduce the power consumption due to screen display light. Mostly after watching desired content people unintentionally end up continuously looks over useless content [11, 12]. Observing too many details we tend to not close eyes, for not missing those details, though the camera would not able to recognize you, not to worry because it keeps the screen on while it does not detect any face or eyes and It also does not interfere with the computer is when your camera is unable to detect you. Your device will work the same as normal before. It is most suitable as directly incorporating into existing operating systems. As operating system is the medium in between human and computer interaction. It is also a responsibility for operating systems to care for user's betterment. We hope that the operating systems will not look at this as a downfall in commercial use of computers.

II. PROPOSED METHOD

We propose a simple but efficient solution to solve the computer vision syndrome to develop a robust connection of human eyes with the screen display light. The ability of state-of-art human computer interaction integrating eye blink detection from the front camera or facing towards the user face for controlling screen display light by modifying screen brightness, also to control screen flickering due to casual blinks that are usually made time to time. It is useful to automatically turn off-screen while the user will close his eyes, which immediately turn on-screen while user opens his eyes and keep the screen-on while no eyes are detected. This novel approach gives users the freedom to take a few naps in front of the screen. To an extent, this will mostly protect

978-1-6654-6643-1/22/$31.00 ©2022 IEEE

user's privacy by making the content invisible to others It is also an energy saving method to reduce the amount of power consumed due to screen display light. This integrates eye blinks for display light on/off and also to calculates the amount of time taken since your last blink and provide you with a message, to remind you to blink. A novel real-time eye blink detection to control display light under different situations:

- Case 1: Automatically turns off display light, detecting user eyes while the user is recognized with closed eyes (After ignoring time for casual blinks).
- Case 2: Immediately turn on the display light while the user will open his/her eyes.
- Case 3: The screen stays on while no eyes are detected or unable to detect eyes (Due to environment lighting conditions or due to external obstacles between user eyes and camera).

Note: It is trained to ignore the casual blinks that are under 1.5 sec make display light stable to avoid screen flickering.

III. Results And Discussions

A cross-platform background running application which is built using electron to smartly to control screen light to turn on/off based on whether his eyes are open or not. It is useful to control the display light considering situations, it automatically turns off the display light by detecting the user eyes while the user will close his eyes, immediately turn on the screen while he opens his eyes and keep the screen-on while no eyes are detected. It ignores the casual blinks that are under 1.5 sec to avoid screen flickering. This counts the total amount of time from your last blink and set notifications to display an alert message, to remind you that you haven't closed your eyes for a long time, will give a pop-up message while the user has not blinked for a long period of time.

A. Detecting blinks using facial landmarks and OpenCV

Initialize face-detector DLIB and facial landmark detector, DLIB uses a pre-trained face detector [12, 13], as shown in Fig.1. it is formed by 68 different facial landmark points. By this we can determine the starting and ending array of index values (x, y), to use coordinates of left eye and right eye.

Figure 1. DLIB facial landmarks [11]

B. Face landmark detection to localize eyes

We calculate the eye aspect ratio for each eye, which gives the distances between vertical eye landmark points to the distances between horizontal land-marking points. We represent it as 6 coordinates (x, y) it starts from the left-most corner of the eye and moves clockwise around the remaining regions

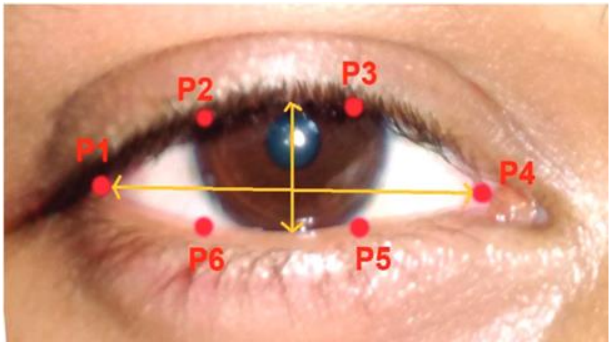


Figure 2. landmarks for an eye.

The relation between the coordinates with respect to width and height. Eye aspect ratio (EAR) formula as proposed from [11]. ||P2-P6||+||P3-P5|| → used to calculate distance between vertical eye landmarks. 2||P1-P4|| → used to calculate distance between horizontal eye landmarks.

$$\mathrm{EAR} = \frac{\|p_2 - p_6\| + \|p_3 - p_5\|}{2\|p_1 - p_4\|}$$

EAR is approximately constant while the eye is open, and goes down to zero when a blink takes place. When the distance between p2, p6 and p3, p5 close to zero as shown in Fig. 3., so the EAR value will drop to zero.

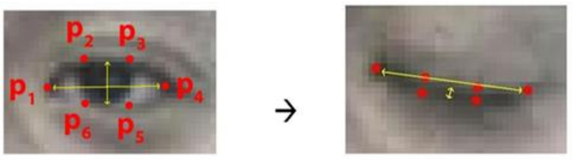


Figure 3. Landmarks of open eye (left) & closed eye (right).

The graph in Fig. 4. shows EAR is constant (if the person keeps his eyes open). Then drops near to zero and increases again, This indicates a blink has been completed.

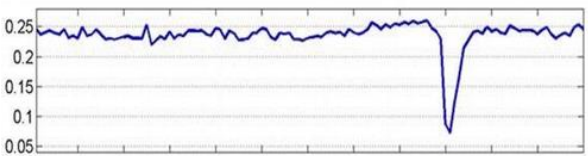
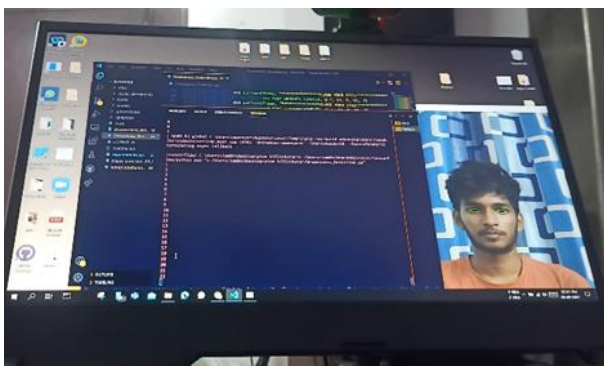
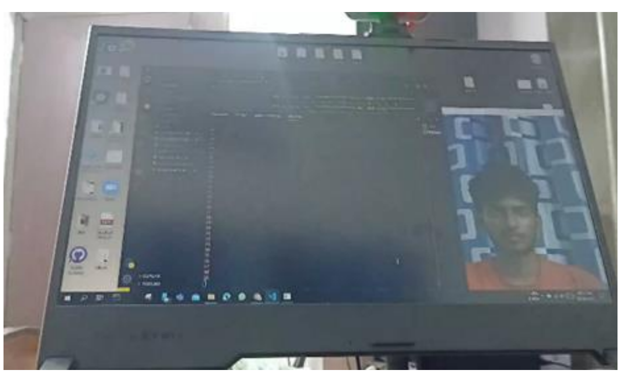


Figure 4. Eye aspect ratio graph.

A background application useful to control the display light considering conditions: It automatically turns off display light by detecting user eyes while the user will close his eyes, immediately turn on- screen while he opens his eyes keep the screen-on while no eyes are detected as shown in Fig. 5.



(a)



(b)

Figure 5. Display light is on while user eyes are open (a) and low display light or display light off while eyes are sclosed (b).

C. Removal of screen flickering

Flickering is experienced by the user while casual blinking to eliminate this we ignore the casual blinks which are less than 2 seconds at least. It is recommended to use in the range of 2 to 10 seconds for maximum efficient use of the application performance. It ignores the casual blinks by implementing sleep function (Time. Sleep (2)) this pause reading (flagging stops) the face for 2 seconds and will continue reading the blinks after 2 seconds, count restarts from the beginning. As shown in Fig. 6. It is Trained to ignore the casual blinks which are happened under 2 sec make display light stable and avoid screen flickering.

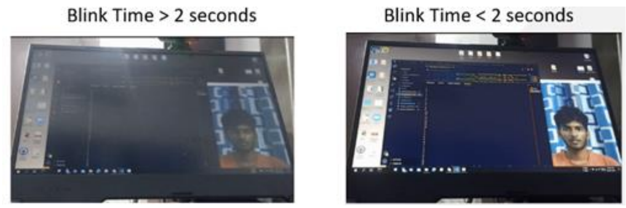


Figure 6. Turn-off screen only when blink greater than 2 seconds.

This process integrates eye blinks for display light on/off and calculate the amount of time taken from your last blink and provide you with popup message, to remind you to blink. As it runs in the background it gives a popup message which works on any window that you are working on. As shown in Fig.7. It can display a notification while the user has not blinked like in the last 13 seconds you will get a reminder message to blink and protect your eyes. Can set the flag for 140(takes around 13 seconds) Which give a notification "take rest, close your eyes" It appears for 5 sec and continue to keep tracking the eyes to count the blinks again. Count the total amount of time that you have not taken a blink till the last the blink. Set notifications to display an alert message to remind you to close your eyes. Will give a message while the user has not blinked in the last 13 seconds (as per the user preference settings).

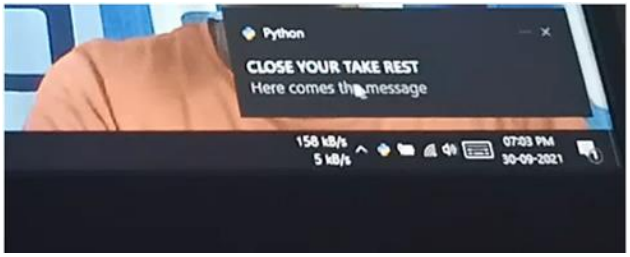


Figure 7. Popup notification as a reminder to blink.

For eye blink detection flowchart of the proposed system is given Fig. 8. From this flowchart some steps can be deduced. In the first step, it reads the frame from the webcam and searches for face to detect the eye. The EAR method from [11]. At first it finds the eye then it will calculate EAR, compares it to its previous value. Further the flag is generated and keeps incrementing till the blink happens, it helps to count and find blink rate with time. Application methodology for controlling display light of the device integrating eye blinks. As shown in Fig. 9. Following steps will be executed, control the display light considering situations. Automatically turn off display light detecting user eyes while the user will close his eyes, immediately turn on-screen while user opens his eyes. Keep the screen-on while in case no eyes are detected.

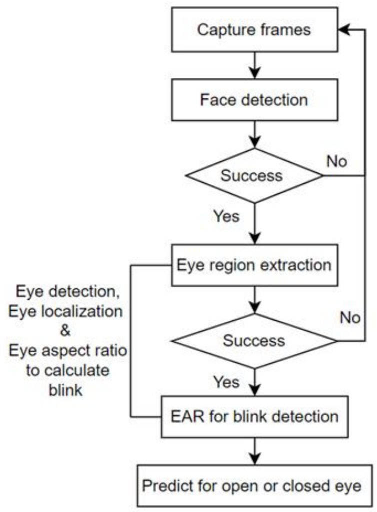


Figure 8. Flowchart for blink detection.

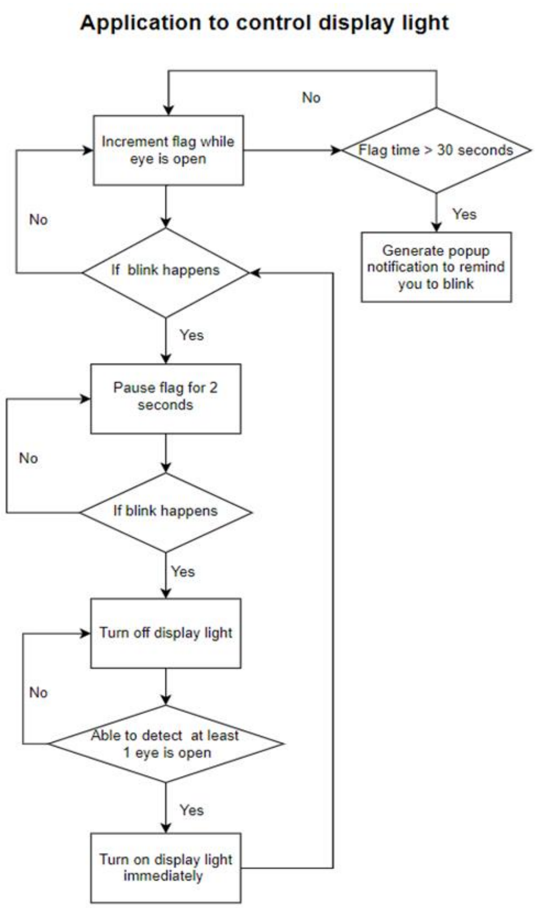


Figure 9. Flowchart of application to control display light.

It Ignores the casual blinks that are under 2 sec make display light stable to avoid screen flickering. Count the total amount of time that you haven't taken a blink till the last blink. Set notifications to display you an alert message to remind you to close your eyes. Will give you a pop-up message while the user has not blinked in the last 13 seconds (as per the user preference), is displayed over any application that you're currently using. Automatically turns off display light detecting user eyes while the user is recognized with closed eyes. As shown in Fig. 5. Displays light on while user eyes are open (a) and low display light or display light off while eyes are sclosed (b). Will keep the screen on while no user is detected. Ignores casual blinks that are under 2 seconds. Keeps the screen light on in case one eye is open as shown in Fig. 10. Also puts the screen on in case no eyes or no face is detected. This tells that the maximum capabilities are covered to be able to look at the screen directly with the naked eyes.

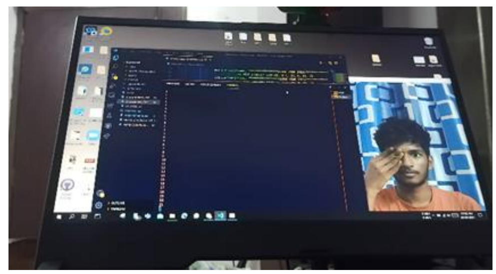


Figure 10. Keeps display on while one is closed and another is open.

IV. CONCLUSION

It increases the productivity of the user as people likely tend to watch useless content after completing the desired content. To a major extent, this will also protect user's privacy by making the content invisible to others but it is not totally safe when someone turns off the light or when you cannot be recognized due to physical or surrounding light changes. Efficiently save a lot of energy that will be consumed over screen display light, though it uses camera for eye detection in most of the cases the power consumed due to camera is much less than the display light. Of course, it will not work while your eyes are unable to detect or not visible for detecting your eyes. But it will not turn your screen off so you can keep using the device. For demonstrating purpose it was done as an application based GUI interface to execute. But this can be directly useful to add as an operating system feature. We believe that the operating systems will only look at this as up-gradation of the existing operating systems and hope it is not considered as a downfall in the commercial of computers.

REFERENCE

[1] A. Altalhi, W. Khayyat, O. Khojah, M. Alsalmi, and H. Almarzouki, "Computer vision syndrome among health sciences students in Saudi Arabia: prevalence and risk factors," Cureus, vol. 12, no. 2, 2020.

[2] I. Bacivarov, M. Ionita, and P. Corcoran, "Statistical models of appearance for eye tracking and eye-blink detection and measurement," *IEEE Trans. Consum. Electron.*, vol. 54, no. 3, pp. 1312-1320, 2008.

[3] A. Irimiás, T. Csordás, K. Kiss and G. Michalkó, "Aggregated roles of smartphones in young adults' leisure and well-being: a diary study," *Sustainability*, vol. 13, no. 8, p. 4133, 2021.

[4] Z. Li, G. Sun, F. Zhang, L. Jia, K. Zheng, and D. Zhao, “Smartphone-based fatigue detection system using progressive locating method,” *IET Intell. Transp. Syst.*, vol. 10, no. 3, pp.148-156, 2016.

[5] C. Blehm, S. Vishnu, A. Khattak, S. Mitra and R.W. Yee, “Computer vision syndrome: a review,” *Surv. Ophthalmol.*, vol. 50, no. 3, pp.253-262, 2005.

[6] P. Gringras, B. Middleton, D.J. Skene and V.L. Revell, “Bigger, brighter, bluer-better? Current light-emitting devices–Adverse sleep properties and preventative strategies,” *Public Health Front.*, vol. 3, p. 233, 2015.

[7] C.A. Chu, M. Rosenfield, and J.K. Portello, “Blink patterns: reading from a computer screen versus hard copy,” *Optom. Vis. Sci.*, vol. 91, no. 3, pp. 297-302, 2014.

[8] S. Jaiswal, L. Asper, J. Long, A. Lee, K. Harrison, and B. Golebiowski, “Ocular and visual discomfort associated with smartphones, tablets and computers: what we do and do not know,” *Clin. Exp. Optom.*, vol. 102, no. 5, pp.463-477, 2019.

[9] K.Y. Loh, and S.C. Redd, “Understanding and preventing computer vision syndrome,” *Malays. Fam. Physician*, vol. 3, no. 3, p.128, 2008.

[10] A. Królak, and P. Strumiłło, “Eye-blink detection system for human–computer interaction,” *Univers. Access Inf. Soc.*, vol. 11, no. 4, pp.409-419, 2012.

[11] S. Al-gawwam, and M. Benaissa, “Robust eye blink detection based on eye landmarks and Savitzky–Golay filtering,” *Information*, vol. 9, no. 4, p. 93, 2018.

[12] A. Islam, N. Rahaman and M.A.R. Ahad, A study on tiredness assessment by using eye blink detection,” *Jurnal Kejuruteraan*, vol. 31, no. 2, pp.209-214, 2019.

[13] A. Udayashankar, A.R. Kowshik, S. Chandramouli, and H.S. Prashanth, “Assistance for the paralyzed using eye blink detection,” In 2012 Fourth International Conference on Digital Home (pp. 104-108), IEEE.