Journal of Information Systems Engineering and Management
2025, 10(26s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Converting Eye Blink to Speech for People with Motor Neuron Disorders

M Dhanalakshmi1, Vyshali Rao K P2, R.Vijaya Arjunan*3, Ratikanta Majhi4, Lavanya Naik5

[1]Dept of CSE New Horizon College of Engineering Bangalore, India. Email:mdhanalak72@gmail.com

[2]Dept of ISE JSS Science and Technological University Mysore, India. Email:vyshali.rao@jssstuniv.in

[3]Department of Computer Science and Engineering, Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal, Karnataka 576104, India. Email: vijay.arjun@manipal.edu

[4]Dept of CSE, Bangalore Technological Institute, Bangalore, India

[5]Dept of CSE(Data Science), B.M.S. College of Engineering, Bangalore, India

Corresponding author: R.Vijaya Arjunan* (Email: vijay.arjun@manipal.edu)

ARTICLE INFO

Received: 22 Dec 2024

Revised: 14 Feb 2025

Accepted: 25 Feb 2025

ABSTRACT

A novel communication system is designed for individuals with Motor Neuron Disease (MND), a rare neuro-degenerative disorder that impairs voluntary muscle function. Our system leverages Human-Computer Interaction (HCI) principles, utilizing eye motion and blink detection to facilitate communication with the external environment. The proposed system is non-invasive, ensuring no harm to the patient or their surroundings. The core of our methodology involves the application of AdaBoost Training and Cascading Classifiers for effective face detection. This technique translates detected eye blinks into synthesized speech, enabling users to communicate despite severe physical limitations. The system's efficiency is enhanced by a robust eye-tracking mechanism that focuses on the pupil's movements, offering a reliable alternative communication interface for those with significant physical disabilities.

Keywords: Eye blink to voice, Motor, Neuron Disorders, Haar Cascade Classifier, Eye blink Detection, Speech Conversion.

INTRODUCTION

The continuous advancements in electronic devices, particularly in the realm of smartphones and tablets, have led to an unprecedented demand for more sophisticated and accessible technology. As these devices become increasingly integral to daily life, there is a growing emphasis on developing innovative methods for human-computer interaction (HCI) and human-mobile interaction (HMI) [1], particularly for individuals with severe physical impairments, such as those suffering from tetraplegia, a condition characterized by the inability to move parts of the body below the neck. [2] One of the most promising developments in this area is the use of eye blink detection as an alternative input modality [3]. Eye blinking, a rapid and involuntary action of closing and reopening the eyelids, can be harnessed to enable users to interact with electronic devices without the need for traditional physical input methods. This approach is particularly beneficial for individuals with disabilities, as it provides a means of communication and control that is both intuitive and accessible. To achieve accurate eye blink detection, advanced image processing techniques are employed. The Haar Cascade Classifier [4], a machine learning-based approach, is frequently used for detecting objects in images, such as faces and eyes. This method, combined with the Camshift algorithm [5] for tracking facial features, allows for the continuous monitoring of eye movements and the detection of blinks.

The positioning of the eyes within a frame is further refined using an adaptive Haar Cascade Classifier, which applies to a cascade of boosted classifiers based on Haar-like features. This technique utilizes the geometric relationship between the eyes and the facial axis to ensure precise eye localization, which is crucial for reliable blink detection. The algorithmic efficiency of these methods has been demonstrated in real-time applications, underscoring their potential for widespread use. One notable application of this technology is the EyePhone system [6], which leverages the front-facing camera of a smartphone to track the user's eye movements across the device's display. By interpreting these movements and associated actions, such as winking, EyePhone allows users to control mobile applications and functions in a completely hands-free manner. This not only enhances accessibility for individuals with physical disabilities but also opens new possibilities for interaction in a variety of contexts, including gaming, virtual reality, and even driving safety. The EyePhone system represents a significant leap forward in the field of HCI, offering a glimpse into the future of mobile interaction where traditional input devices are no longer necessary. As these technologies continue to evolve, they hold the potential to transform the way we interact with our devices, making them more inclusive and responsive to the needs of all users [7][8]. The ongoing research and development in this field are poised to deliver even more advanced and intuitive solutions, further bridging the gap between human capability and technological innovation.

2 RELATED WORKS

2.1 Communication Technologies Based on Voluntary Blinks: Assessment and Design

The detection of eye blinks within the context of image processing methods generally encompasses two primary stages:

• Eye Position Detection

• Classification of Eye State (Open or Closed)

For the initial step, one of the most prevalent algorithms utilized is the Viola-Jones object detector, which leverages Haar-like features in conjunction with an AdaBoost learning algorithm [9] [10]. The AdaBoost algorithm described in Algorithm 1 is employed to iteratively select the most informative features from the dataset and to train a robust classifier capable of accurately identifying eye positions. The Electro-Encephalogram, a test that measures electrical activity in the brain and the output is analyzed in Figure 1.

2.2 Converting the Intent of Physically Impaired Individuals into Text Using Eye Blink Detection

The identification of involuntary eye blinks as a means of converting user intent into textual output has been explored through various predetermined actions. Applications implementing this technique are typically developed in Python (using the PyCharm IDE) and make extensive use of the OpenCV library [11][12]. The system architecture is built using readily accessible hardware, such as a standard camera and a personal computer, which together provide the foundational infrastructure for operation. To optimize system performance, it is crucial that the distance between 66 the user and the webcam does not exceed 100 cm. The user, positioned approximately 50 cm from the screen, visualizes the interface, where a blinking cursor (blinker) navigates sequentially through sets of horizontal matrices [13][14]. Upon detecting a blink, the system shifts the cursor's motion vertically, with a brief time delay to confirm the user's selection. If the input is incorrect, [15] the entire process resets, and the sequence restarts from the beginning. Once the correct input is confirmed, the system immediately displays the selection above the matrix.

2.3 Eye Blink-Controlled Virtual Interface Using OpenCV and Dlib.

• Face Detection: Face detection is a critical technology for recognizing human faces in images or video streams. OpenCV and dlib libraries are commonly employed to accomplish this task using a variety of methods [16]. In this context, the detector is constructed using the Histogram of Oriented Gradients (HOG) feature combined with a linear classifier. The dlib library also implements a facial landmarks detector, which is instrumental in identifying key facial features such as the eyes, ears, and nose.

• Eye Detection: Following face detection, the eye region is isolated by leveraging facial landmark features. The face landmarks dataset includes 68 specific points, each of which is assigned a unique index. These indices allow for precise detection of desired facial regions. For eye detection, the following indices are used: [17]

Left eye: (37, 38, 39, 40, 41, 42)

Right eye: (43, 44, 45, 46, 47, 48)

Once the eye region is extracted, it undergoes further processing to detect eye blinks. This eye region detection occurs at the initial stage of the system, forming the basis for subsequent blink analysis.

2.4 Face Detection Using Haar Cascades to Filter Selfie Face Images on Instagram

The Haar cascade method [18], developed by Paul Viola and Michael Jones, is a foundational technique for human face detection [19]. In this method, Haar features play a crucial role as they form the backbone of the Haar cascade classifier. These features are employed to determine the presence of specific elements within a given image. Each Haar feature computes a single value by subtracting the sum of pixel intensities under a white rectangle from the sum under a black rectangle. These Haar features act as rectangular filters that rapidly detect human faces in images. The scanning process using Haar-like features begins at the upper left corner of the image and continues until the lower right corner reaches [20]. Multiple scans are performed to ensure the accurate detection of faces within the image, making this method particularly effective for applications such as filtering selfie images on social media platforms like Instagram.

3 EQUATIONS AND MATHEMATICAL EXPRESSIONS

3.1 Camera and Frame capturing

The first phase of the proposed system involves initialization. Initially, a brief video of the participant's face is captured using the front-facing camera, typically on a laptop. The captured video is then processed into individual frames using a method known as Process Frame. These frames, originally in color, are converted to grayscale by isolating only the luminance component. The conversion employs the luminosity method, a refined approach that considers human visual sensitivity by assigning greater weight to green. The formula used in the luminosity method is 0.21R + 0.72G + 0.07B, making it highly effective for this purpose. To expedite the calculation of rectangular features, the concept of integral images is utilized. This method requires four corner values to determine the pixel count within a specified rectangle. In an integral image, the pixel value at coordinates (x, y) represents the sum of all pixels above and to the left of (x, y).

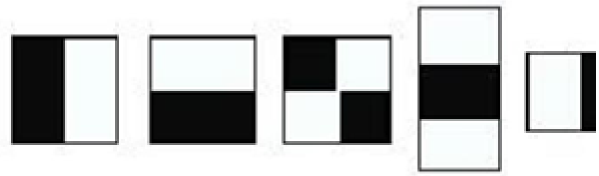

Figure 1: Electro-Encephalogram

The Viola-Jones algorithm begins feature evaluation using a 24x24 window as the base size. Considering all possible Haar feature parameters, including type, position, and scale, results in approximately 160,000 features within this window. However, evaluating all these features is computationally infeasible. To address this, the AdaBoost algorithm is employed. AdaBoost is a machine learning technique that identifies the most significant features among the 160,000 available. These selected features serve as weak classifiers, and AdaBoost constructs a robust classifier by combining these weak classifiers in a linear fashion.

3.2 Face Detection

The face detection algorithm in 2 works in the following order

Initialization: Weights for each training example are initialized equally.

Weak Classifier Training: For each iteration, a weak classifier is trained to minimize classification error. This classifier is typically a simple decision rule based on features extracted from the image patches.

Weight Calculation: The error rate of the weak classifier is used to calculate its weight. A classifier with lower error contributes more to the final model.

Weight Update: The weights of training examples are updated to focus more on misclassified examples. This allows the algorithm to improve in areas where previous classifiers performed poorly.

Strong Classifier: After T iterations, the final classifier H(x) is a weighted combination of all the weak classifiers. This strong classifier is then used for face detection.

3.3 Eye Detection

To effectively detect the eye, it is crucial to first train the Haar Cascade Classifier. This process necessitates the implementation of both the AdaBoost algorithm and Haar feature extraction methods. Training the classifier requires two distinct sets of images: one set containing scenes with the object of interest (in this case, the eye) and another set containing scenes without the object. The Enhanced Boosted Cascade Model (EBCM) leverages the Haar Cascade Classifier to identify regions of interest, with detected areas highlighted using bounding boxes. The AdaBoost algorithm plays a pivotal role in refining the classifier by training node classifiers on a set of Haar-like features, which enhances the generalization capacity of the classifier nodes. The application of AdaBoost in this context aims to optimize the performance of face detection systems. Experimental evaluations have demonstrated that this approach effectively reduces the number of weak classifiers required, accelerates the detection process, and provides a modest improvement in detection accuracy. Consequently, the proposed methodology not only streamlines the classification process but also enhances the overall efficiency and effectiveness of face detection.

Algorithm 1 AdaBoost Algorithm for Face Detection

Input: Training data $D = \{(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)\}$, where x_i is the image patch and y_i is the label ($y_i \in \{-1, +1\}$)

Number of iterations T

Output: Final strong classifier $H(x)$

Input :Initial weights: $w_i = \frac{1}{N}$ for $i = 1, 2, \ldots, N$

for $t = 1$ ***to*** T **do**

Train a weak classifier $h_t(x)$ using the weighted training data Compute the weighted error of $h_t(x)$:

$$\epsilon_t = \frac{\sum_{i=1}^{N} w_i \cdot [h_t(x_i) \neq y_i]}{\sum_{i=1}^{N} w_i}$$

Compute the weight α_t for the weak classifier $h_t(x)$:

$$\alpha_t = 0.5 \cdot \ln\left(\frac{1 - \epsilon_t}{\epsilon_t}\right)$$

Update the weights for all training examples:

$$w_i \leftarrow w_i \cdot \exp(-\alpha_t \cdot y_i \cdot h_t(x_i))$$

Normalize the weights so that they sum to 1:

$$w_i \leftarrow \frac{w_i}{\sum_{j=1}^{N} w_j}$$

Output: Final strong classifier:

$$H(x) = \text{sign}\left(\sum_{t=1}^{T} \alpha_t \cdot h_t(x)\right)$$

3.4 Eye tracking

In the context of eye movement tracking using the Enhanced Boosted Cascade Model (EBCM), two critical features are extracted from the eye: the corneal reflection and the pupil center. These features are essential for accurate tracking of eye movement. By identifying and analyzing the position of the pupil's center and the corneal reflection, we can measure the vector between these two points. Further trigonometric calculations allow for the determination of the point-of-regard, which is crucial for understanding where the individual is looking. The EBCM method demonstrates significant efficacy in ensuring that the movement of the eye's pupil and the face occur in synchrony and in the same direction. To illustrate, consider X as the detected human face, and P1 and P2 as two points associated with the left eye. The movement of P1 and P2 is directly correlated with the movement of X, maintaining synchronized motion. This synchronization indicates that the EBCM method effectively tracks both the face and the eye, ensuring cohesive and accurate movement detection.

3.5 Eye Aspect Ratio

Facial landmark detection is a powerful technique used to localize key facial regions, including the eyes, eyebrows, nose, ears, and mouth. This approach enables the extraction of detailed facial structures by referencing the indices of specific facial components. For blink detection purposes, the focus is primarily on the eyes, which are characterized by a set of six (x, y) coordinates. These coordinates begin at the left corner of the eye and proceed in a clockwise manner around the eye's perimeter. To quantify eye blink dynamics, we can derive a crucial metric from these coordinates. Specifically, the relationship between the width and height of the eye can be described using the following formula:

$$EAR = 2 \cdot \|p1 - p4\| / \|p2 - p6\| + \|p3 - p5\|$$

where p1, p2, p3, p4, p5, and p6 represent the 2D coordinates of the facial landmarks around the eye. This metric, known as the Eye Aspect Ratio (EAR), provides a quantitative measure of the eye's openness, which is instrumental in detecting blinks and monitoring eye movement.

3.6 Speech Conversion

The conversion to speech is carried out by using the GTTS tool. This tool works by taking eye movements as an input and producing speech output.

Algorithm 2 Eye Blink to Speech using DTMW developed algorithm

Input: Morse coded input signal
Output: Decoded text output
3 **while** *(Text not fully decoded)* **do**

1. Read Morse coded input signal
2. Determine space moments and the number of letters in the signal
3. Determine the position of negative peaks
4. Determine the number of eye blinks
5. Determine eye blink moments
6. Apply rectangular windowing
7. Determine whether eye blink is single or double using DTMW (Dynamic Time and Memory Wrapping)
8. Determine the letter

if *(Last letter is incorrect)* **then**
Goto step 3
Determine the hold Output text to speech

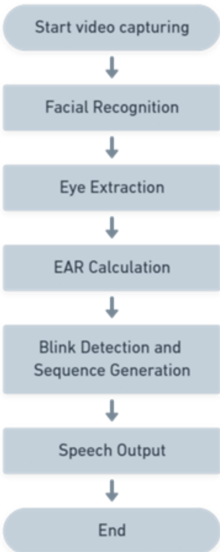

Figure 2: Data Flow Diagram of the Proposed process

4 RESULTS

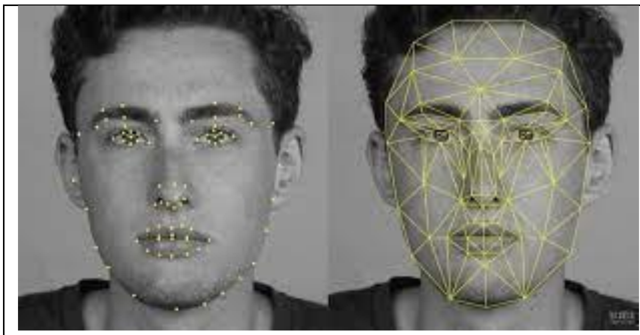

Figure 3: Detection of facial features

(a) Detection of Human Face

(b) Detection of Human Eye;

Table 3. T1, T2, T3, T4, | Xpos |, | Xneg | and DNG values for 10 subjects

Subject	T1 (Sample)	T2 (Sample)	T3 (Sample)	T4 (Sample)	|Xpos| (µV)	|Xneg| (µV)	DNG (Sample)
1	490	993	1010	2212	1315	2050	313
2	485	975	995	2256	1398	2014	395
3	493	983	995	2213	1301	2095	325
4	475	976	992	2202	1489	2095	345
5	488	989	1005	2401	1445	2084	412
6	491	988	1002	2207	1400	2052	486
7	465	952	975	2387	1398	2041	366
8	481	961	1012	2276	1395	2020	399
9	480	990	995	2209	1495	2101	452
10	491	985	1002	2235	1325	2074	305

To enhance the accuracy of pupil detection, the following approach is employed:

4.1 Grayscale Conversion and Head Detection

The initial step involves converting the input image to grayscale. This preprocessing simplifies 179 the image and improves the efficiency of subsequent operations. Following this, the Haar Cascade Classifier is utilized to detect the head within the image. Sample data taken for consideration are given in table 1.

4.2 Eye Region Cropping

Given the limitations of direct eye detection with Haar cascades—where it often erroneously identifies eyebrows as eyes—the cropping approach is employed to approximate the eye region 185 more reliably. The cropping process is defined by the following parameters:

• Origin of Cropped Image: The origin of the cropped region is calculated as the origin of the detected head plus an offset of (0, Heighthead/5.5).

• Width of Cropped Image: The width of the cropped region is set equal to the width of the detected head.

• Height of Cropped Image: The height of the cropped region is determined as (Heighthead/3.0).

This methodology is preferred over direct eye detection due to its improved accuracy. Direct eye detection often suffers from inaccuracies, such as detecting eyebrows as eyes, which can lead to suboptimal results. By approximating the eye region based on the head's dimensions, the cropping method provides a more reliable basis for further analysis. The dataflow in the proposed process in described in Figure 2.

4.3 Use of Iris as Pupil Approximation

In subsequent processing stages, the iris is used as an approximation for the pupil. This approach is chosen to mitigate the challenges associated with direct pupil detection and to leverage the more consistent detection of the iris. This approach ensures a more robust extraction of the glint and accurate approximation of the pupil, thereby improving the overall reliability of the eye tracking system. Figure 3 shows features in the human face and eye to be detected for processing.

4.4 Analysis of Morse Code Signal Characteristics

Two critical parameters are essential for Morse code input analysis: T4 moments and the Xneg 204 value. The parameters α and β, defined in Equations (1), vary across subjects during data entry. 205 Analysis of data from 10 subjects, as summarized in Table 1, reveals that these parameters converge to approximately $\alpha = 2$ and $\beta = 2.2$

$$T1 < T2 \text{ and } T3 < T4 \; T2 = \alpha \cdot T1 \text{ and } T4 = \beta \cdot T3 \tag{1}$$

In terms of the temporal characteristics of the Morse code signals, the average values observed are as follows:

- T1 (maximum): 500 samples
- T2 (maximum): 1000 samples
- T3 (average): 1000 samples
- T4 (minimum): 2200 samples

The amplitude values of the positive and negative peaks also play a crucial role. According to Equation (2), the positive peak amplitude |Xpos| ranges between 1300 and 1500 μV, while the negative peak amplitude |Xneg| ranges from 2000 to 2100 μV. Notably, the amplitude of the negative peak is consistently higher than that of the positive peak, which enhances the accuracy of eye-blink detection by focusing on negative peaks. When calculated using Equation (3), it has been observed that positive peaks typically occur 300 to 400 samples earlier than negative peaks. To account for this temporal discrepancy, an initial shift of DNG = 500 samples is applied, ensuring that both negative and positive peaks are captured within the window period. This alignment improves the accuracy of template matching in the Dynamic Time Warping (DTW) algorithm.

$$Xpos(k) < Xneg(k) \tag{2}$$

$$X(n) > X(n+1) \; X(n) > X(n-1) \; X(n-1) = X(n+1) \; X(n) > E \tag{3}$$

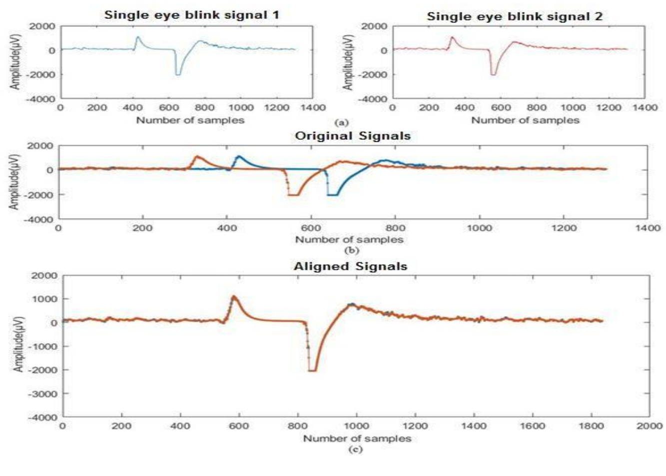

Figure 6. (a) Two signals with a single eye-blink (b) Indication of the signals in (a) on the same axis set (c) Aligned signals obtained from the original signals in (b)

Figure 4 shows eye-blink with single and double patterns. In this alignment, one signal represents the single blink pattern used by the DTW algorithm, while the other represents the signal from a subject at a given moment. The DTW algorithm corrects for time shifts between the signals, revealing a Euclidean distance of 20.137 between the aligned single blink signals. This distance metric quantifies the similarity between the observed and reference signals, demonstrating the effectiveness of the DTW alignment in handling temporal variations in eye-blink detection. The proposed system for real-time eye blink detection offers several advantages, making it an effective solution for enhancing human-mobile interaction, particularly for individuals with disabilities such as tetraplegia. With high detection accuracy of 98% for eye detection and 100% for blink recognition, the system ensures reliable performance, even under artificial lighting and at 35 cm. Its real-time efficiency, with an average frame processing time of 71 milliseconds, enables seamless interaction without noticeable lag. By distinguishing between intentional long blinks and natural blinks, it enhances precision and reduces false triggers, making it ideal for controlling mobile devices. The lightweight design, using the Haar cascade algorithm with AdaBoost classifiers, ensures compatibility with mobile platforms while supporting a range of applications, including healthcare monitoring, mobile interaction, and driving safety. Furthermore, the system generates voice and text outputs, facilitating effective communication for users with speech or motor impairments, and demonstrates adaptability across variable lighting conditions, broadening its usability in diverse environments. The proposed system has several limitations that need to be addressed.

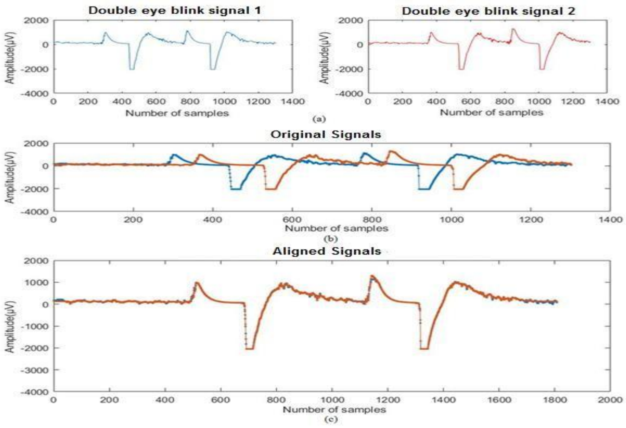

Figure 7. (a) Two signals with a double eye-blink (b) Indication of the signals in (a) on the same axis set (c) Aligned signals obtained from the original signals in (b)

It is sensitive to lighting conditions, with reduced accuracy in low-light or dynamic environments. The system's high accuracy is maintained only at a fixed distance of 35 cm, limiting its flexibility. The Haar cascade algorithm struggles with occlusions, such as glasses, and rapid head movements. It does not account for more nuanced eye gestures beyond normal and intentional long blinks. Variability in user characteristics, such as eye shape and blinking behavior, may result in inconsistent performance. Differences in camera quality and processing power across mobile devices can also impact system reliability. Future improvements will focus on developing algorithms robust to dynamic lighting and varying distances. Advanced models like Convolutional Neural Networks (CNNs) will improve detection under challenging conditions. Expanding gesture recognition capabilities to include winks and double blinks will enhance usability. Efforts will also include personalized calibration, cross-device optimization, and better energy efficiency for prolonged use. Real-world testing in diverse environments will ensure scalability and practical usability.

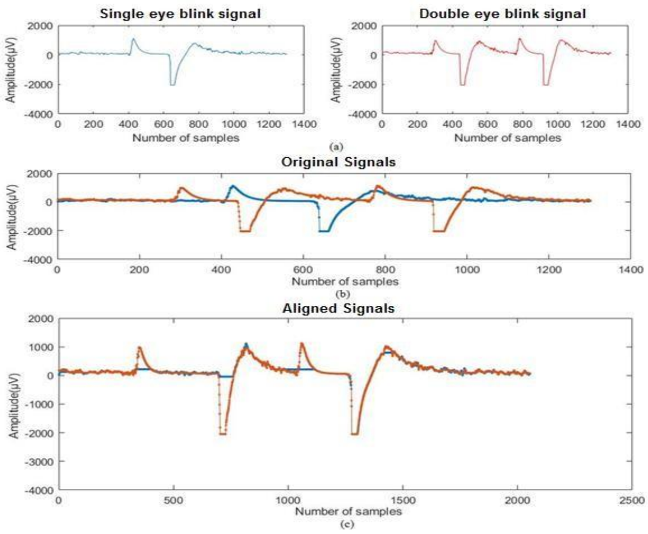

Figure 8. (a) Signals with a single eye-blink and a double eye-blink (b) Indication of the signals in (a) on the same axis set (c) Aligned signals obtained from the original signals in (b)

5 CONCLUSION

Detecting eye blinks in real-time for mobile phone control is challenging due to variability in eye movement, lighting conditions, and camera distance. The proposed methodology improves accuracy for eye detection and blink recognition by 8%, achieving 98% and 100% detection accuracy, respectively, under artificial lighting at 35 cm. With an average frame processing time of 71 milliseconds, it ensures real-time efficiency. Tetraplegia, which limits movement below the neck, underscores the importance of human-computer interaction (HCI) technologies for individuals with disabilities. Eye blinks serve as a critical input modality, enabling interaction with mobile devices and computing systems. The system distinguishes between normal and intentional long blinks to enhance user interaction. Video frames from the system's camera undergo face detection to identify a bounding box around the face. Using the Haar cascade algorithm with an AdaBoost classifier, the eye region is pinpointed within the detected face. Eye tracking and blinking data are then processed to generate corresponding voice and text outputs. This enables effective communication and control across applications in mobile interaction, healthcare, and driving safety.

AUTHORS' CONTRIBUTION:

M Dhanalakshmi[1] : Conceptualization building and Data Collection.

Vyshali Rao K P[2] : Methodology and Data analysis

R.Vijaya Arjunan*[3] : Writing - Review and Editing

Ratikanta Majhi[4]: Writing - Review and Editing and overall supervision

Lavanya Naik[5]: Review and Editing

All authors have read and approved the final version of the manuscript

Funding Information: This research received no external funding.

Data Availability Statement :

The data that supports the findings of this study are available from the corresponding author(s), [R.Vijaya Arjunan and D.Cenitta], upon reasonable request.

Conflict of Interest: The authors declare that they have no conflict of interest.

REFERENCES

[1] Cui L, Liu J. Virtual human: A comprehensive survey on academic and applications. IEEE Access. 2023.

[2] Sumit Kushwaha MD. Efficient Liver Disease Diagnosis Using Infrared Image Processing For Enhannced Detection And Monitoring. In: Journal of Environmental Protection and Ecology 25, No 4, 1266–1278 (2024) Public health; 2024.

[3] Devisivasankari P, Vijayakumar R; IEEE. Conversational AI Annotation System. 2023 Second International Conference on Augmented Intelligence and Sustainable Systems (ICAISS). 2023.

[4] Kareem OS, et al. Face mask detection using haar cascades classifier to reduce the risk of Coved-19. International Journal of Mathematics, Statistics, and Computer Science. 2024;2:19-27.

[5] Gurin D, Yevsieiev V, Maksymova S, Alkhalaileh A. CAMShift Algorithm for Human Tracking in the Collaborative Robot Working Area. 2024.

[6] Bastani PB, Rieiro H, Badihian S, Otero-Millan J, Farrell N, Parker M, et al. Quantifying induced nystagmus using a smartphone eye tracking application (EyePhone). Journal of the American Heart Association. 2024;13(2):e030927.

[7] P Hosanna Princye MD. expression analysis of death receptor ligands using biomedical images. In: 294 Journal of Environmental Protection and Ecology 25, No 4, 1279–1290 (2024) Public health; 2024.

[8] M Sarat Chandra Prasad BSRVSB M Dhanalakshmi. Machine Learning-Integrated Sustainable Engineering and Energy Systems Innovations at the Nexus. In: in IGI Global Publishing, May 2024; 2024.

[9] Hornyák O, Iantovics LB. AdaBoost algorithm could lead to weak results for data with certain characteristics. Mathematics. 2023;11(8):1801.

[10] Hao L, Huang G. An improved AdaBoost algorithm for identification of lung cancer based on electronic nose. Heliyon. 2023;9(3). 302

[11] Subha Devi DMSASMLLN M Thanga. Anomaly Detection in Video Surveillance. In: 2024 IEEE 9th International Conference for Convergence in Technology, I2CT 2024; 2024.

[12] Bhavana V AS Dhanalakshmi M. A Brief survey on facial expression detection. In: 10th IEEE Uthar Pradesh Section International Conference on Electrical, Electronics and Computer Engineering, UPCON 2023, pp. 110–115.; 2023.

[13] Zeng W, Xiao Y, Wei S, Gan J, Zhang X, Cao Z, et al. Real-time multi-person eyeblink detection in the wild for untrimmed video. In: Proceedings of the ieee/cvf conference on computer vision and pattern recognition; 2023. p. 13854-63.

[14] Gaddam S, Sudhakar K, Britto CF, Roopa H, Dani R, Kumar K. Recognition of Brain Tumors using Convolutional Neural Networks. In: 2023 Second International Conference on Augmented Intelligence 312 and Sustainable Systems (ICAISS). IEEE; 2023. p. 258-64.

[15] M Dhanalakshmi BGR Vyshali Rao. Air Quality Predictor to Reduce Health Risks and Global Warming. 314 In: 2nd International Conference on Automation, Computing and Renewable Systems (ICACRS),IEEE Conference Dec 11 - 13 2023; 2023.

[16] Agarwal A, Das A. Facial Gestures-Based Recommender System for Evaluating Online Classes. In: Recommender Systems. CRC Press; 2023. p. 173-89.

[17] Sreerenganathan A, KP VR, Dhanalakshmi M. Assistive Application for the Visually Impaired using Machine Learning and Image Processing. In: 2023 14th International Conference on Computing 320 Communication and Networking Technologies (ICCCNT). IEEE; 2023. p. 1-6. 321

[18] Kaur N, Rani S, Kaur S. Real-time video surveillance based human fall detection system using hybrid haar cascade classifier. Multimedia Tools and Applications. 2024:1-19. 323

[19] Aras RA, Endang H. Implementation of Haar Cascade and Adaboost Algorithms in Photo Classification on Social Networks. Inspiration: Jurnal Teknologi Informasi dan Komunikasi. 2023;13(1):59-68. 325

[20] Bahit, M., Utami, N. P., Candra, H. K., & Supit, Y. (2023). Validation of the Haar Cascade Classification Method in Face Detection. *Journal Of Informatics And Telecommunication Engineering*, *7*(1), 233-243.