



Article

Application Specific Reconfigurable Processor for Eyeblink Detection from Dual-Channel EOG Signal

Diba Das ¹, Mehdi Hasan Chowdhury ^{1,*}, Aditta Chowdhury ¹, Kamrul Hasan ¹, Quazi Delwar Hossain ¹
and Ray C. C. Cheung ²

¹ Department of Electrical and Electronic Engineering, Chittagong University of Engineering and Technology, Chattogram 4349, Bangladesh; diba.eee@cuet.ac.bd (D.D.); aditta.eee@cuet.ac.bd (A.C.); kamrulhira@cuet.ac.bd (K.H.); quazi@cuet.ac.bd (Q.D.H.)

² Department of Electrical Engineering, City University of Hong Kong, Kowloon, Hong Kong; r.cheung@cityu.edu.hk

* Correspondence: mhchowdhury@cuet.ac.bd

Abstract: The electrooculogram (EOG) is one of the most significant signals carrying eye movement information, such as blinks and saccades. There are many human–computer interface (HCI) applications based on eye blinks. For example, the detection of eye blinks can be useful for paralyzed people in controlling wheelchairs. Eye blink features from EOG signals can be useful in drowsiness detection. In some applications of electroencephalograms (EEGs), eye blinks are considered noise. The accurate detection of eye blinks can help achieve denoised EEG signals. In this paper, we aimed to design an application-specific reconfigurable binary EOG signal processor to classify blinks and saccades. This work used dual-channel EOG signals containing horizontal and vertical EOG signals. At first, the EOG signals were preprocessed, and then, by extracting only two features, the root mean square (RMS) and standard deviation (STD), blink and saccades were classified. In the classification stage, 97.5% accuracy was obtained using a support vector machine (SVM) at the simulation level. Further, we implemented the system on Xilinx Zynq-7000 FPGAs by hardware/software co-design. The processing was entirely carried out using a hybrid serial–parallel technique for low-power hardware optimization. The overall hardware accuracy for detecting blinks was 95%. The on-chip power consumption for this design was 0.8 watts, whereas the dynamic power was 0.684 watts (86%), and the static power was 0.116 watts (14%).

Keywords: blink; classification; electrooculogram (EOG); field-programmable gate array (FPGA); machine learning; saccade; support vector machine (SVM)



Citation: Das, D.; Chowdhury, M.H.; Chowdhury, A.; Hasan, K.; Hossain, Q.D.; Cheung, R.C.C. Application Specific Reconfigurable Processor for Eyeblink Detection from Dual-Channel EOG Signal. *J. Low Power Electron. Appl.* **2023**, *13*, 61. <https://doi.org/10.3390/jlpea13040061>

Academic Editor: Stefania Perri

Received: 12 September 2023

Revised: 18 October 2023

Accepted: 4 November 2023

Published: 23 November 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Various medical conditions, such as strokes, spinal cord injuries, amyotrophic lateral sclerosis (ALS), and locked-in syndrome (LIS) render patients paralyzed [1,2]. Paralyzed people are unable to move the affected parts of the body either partly or entirely. They face difficulties in performing routine activities and communicating with the external world, which impacts their quality of life [3]. Therefore, scientific research is being conducted to ease the lives of these partially or fully disabled people by creating an alternative form of communication.

Human–machine interfaces (HMIs), also known as human–computer interfaces (HCIs), can be helpful in this regard by connecting the human body with the external environment [2]. Generally, HMIs utilize electro-physiological signals such as electroencephalogram (EEG), electromyogram (EMG), and electrooculogram (EOG) signals [4,5]. Low-power HMIs integrated into portable applications can benefit paralyzed people [6]. However, designing HMIs involves various algorithms in different signal processing stages, resulting in complexity for hardware realization. Thus, there is scope for research on efficient hardware implementations of HMI.

Recently, EOG-based HMI has emerged as a promising technology [7,8]. EOGs contain information related to the eyes, representing various eye movements. Eye movements can be considered an essential element used to express the desires, emotional states, and needs of people [9]. Therefore, translating EOGs can greatly help people with major disabilities. The major advantages of EOGs include their non-invasive nature, consistent signal pattern, and low cost [10,11]. EOGs contain information regarding eye movements: blinks, saccades, and fixation. Blinking refers to the act of shutting and opening the eyes very quickly. Saccades are rapid movements of the eye that change the point of fixation. Fixation refers to the interval between two saccades. During fixation, the eyeball is fixed on the focus.

The blinking information from EOGs is utilized in many healthcare and technological applications. These include communication technologies [1], brain-computer interfaces (BCIs) [12], fatigue detection [13], drowsiness detection [14], and computer vision syndrome detection [15]. Eye blink-related features can also be combined with EEG signals for drowsiness detection in drivers [16]. On the other hand, in many applications of EEG signals, eye blinks can contaminate the original information [17,18]. As blinks are an uncontrollable and involuntary activity, they can create EOG artifacts. The accurate detection of blinks can be beneficial in denoising EEG signals in such cases. Therefore, designing an efficient blink detector is necessary.

Blinking and saccade information can be derived from EOGs using various algorithms, applying either machine learning or traditional approaches. Banerjee et al. [19] detected blinks utilizing the RBF kernel space vector machine (SVM) algorithm and achieved 95.33% accuracy. Ryu et al. [20] adopted a differential EOG signal based on a fixation curve (DOSbFC) to remove baseline drift and noise. They achieved 94.3% accuracy in detecting blinks, horizontal saccades, vertical saccades, and fixation. Molina-Cantero et al. [1] utilized adaptive K-means for classifying single blinks, double blinks, and long blinks with 89.9% accuracy. Gundugonti and Narayanam [21] detected blinks using the Haar discrete wavelet transform architecture with a Radix-2r multiplier and 4:2 compressor. Gundugonti and Narayanam [22] used thresholding for eye movement detection.

In recent years, point-of-care (POC) systems have become popular for providing early detection and quick monitoring [23]. POC refers to systems where testing is performed at the patient's location. These systems require digital designs. Digital design can be defined as the process of designing electronic circuits, systems, or devices that have specific purposes to address needs in the field of concern. These designs provide cost-effectiveness, quick monitoring, remote healthcare facilities, etc. [24]. POC systems based on EOG can be utilized in detecting eye diseases and various eye conditions. A digital blink detector can be considered an EOG-based POC system, as it can be utilized in the diagnosis of diseases by monitoring the blink rate. Digital blink detectors can detect blinks in real time. The detection of blinks can also be utilized in various applications other than disease detection. These applications may include drowsiness monitoring in drivers, which can save them from major road accidents, and fatigue detection in smart office workstations.

A popular approach to digital design involves application-specific integrated circuits (ASICs). These are also known as application-specific processors or application-specific integrated processors. An ASIC refers to an integrated chip designed for a particular application [25]. These designs are seen in many advanced technologies and electronic devices. Application-specific processors can be designed using field-programmable gate arrays (FPGAs). FPGAs have advantages such as a reconfigurable design capability, low latency, and low power [26,27]. FPGAs allow a custom-made circuit design scheme. It is possible to speed up data processing using a parallel architecture and minimize resource utilization via a serial architecture. ASIC design in an FPGA fixes the functionality and reduces the number of components used. Therefore, these compact designs provide the benefits of low cost, high performance, and power efficiency.

ZedBoard is a low-cost development board manufactured by Digilent. This board employs Xilinx Artix-7 FPGAs coupled with a dual-core ARM Cortex-A9 processor [27]. The ZedBoard has a broad range of applications, such as digital signal processing (DSP),

image processing, and industrial automation. This FPGA system-on-a-chip (SoC)-based board enables designers to accelerate the custom DSP algorithm, as it is possible to program the Zedboard whenever necessary [28]. The proposed work used a ZedBoard as the processing unit for hardware implementation. The ZedBoard was programmed using System Generator, a design tool for the implementation of DSP algorithms in Xilinx devices.

In this work, we classified blinks and saccades by adopting a machine learning approach. The software design with linear SVM offers 97.5% accuracy. The feasibility of our design is shown by implementing the design in ZedBoard Xilinx Zynq-7000 FPGA by hardware/software co-design. This work uses dual-channel EOG, i.e., horizontal and vertical EOG. The entire process takes advantage of hybrid serial-parallel techniques of FPGA. The hardware design offers 95% accuracy. The on-chip power consumption for this design is 0.8 watts, where dynamic power is 0.684 watts (86%) and static power is 0.116 watts (14%). The results show that our design can achieve fast, low-power, and real-time EOG processing. The paper makes the following key contributions:

- A dual-channel EOG signal processor for blink detection with better accuracy at the software level than state-of-the-art works.
- A first-of-its-kind hardware-implemented EOG signal processor for blink detection.
- Better accuracy in the hardware-implemented model than the state-of-the-art works.

The rest of this paper is arranged as follows. Section 2 discusses the background of EOG and various eye movements. Section 3 contains a brief description of the methodology of this work. The performance analysis of the proposed work and comparison with state-of-the-art works are depicted in Section 4. Finally, Section 5 concludes this paper.

2. Background

This section contains a brief discussion of the electrooculogram (EOG) signal and various eye movements.

2.1. Electrooculogram

The eye can be considered a dipole, having its positive and negative poles at the cornea and retina, respectively [8]. The eye has a steady corneo-retinal potential generated within the eyeball by the metabolically active retinal epithelium. The potential can be measured by placing electrodes on the skin surface around the eyes, which is referred to as the electrooculography technique. This technique provides electrooculograms (EOGs), which represent the recording of eye movements. The amplitude of EOG varies in the range of 50–3500 microvolts (μV). The frequency range of EOG is 0–30 Hz [10]. EOG is non-invasive, consistent, and inexpensive [10,11]. EOG can contain either horizontal data or vertical data, and in some cases, both depending on the electrode configuration of data acquisition. Nowadays, modified electrode configuration [29], eyeglasses [20], EOG sensors [30], optical sensors [31], and contact lenses [32] are also available for EOG data acquisition. The EOG data suffers from some noises that introduce inaccuracies in diagnosis and other applications. These noises include power line interference, baseline wander, etc. For any biomedical application of EOG, the efficient removal of the noise from the EOG signal is necessary. After the removal of the noises, these denoised signals can be sent for further processing. EOG has a wide range of applications in various fields, such as medical diagnosis, ophthalmic Research, and human-machine interface (HMIs), etc. EOG has a great impact on the development of HMI, as EOG is the primary tool for eye movement analysis. EOG helps to detect various eye movements such as blinks, saccades, and fixation.

2.2. Blinks

Blinking can be defined as an automatic process of eyes getting open and closed. This is a rapid and repetitive movement of the eyelids. Eye blinks are an integral part of the normal function of the eyes. This movement helps to spread a thin layer of tear film across the surface of the eye and ensures moisture in the eyes. Blinks also protect eyes from damage caused by excessive brightness by reducing the amount of light entering the eye.

Generally, blinking occurs throughout the whole day. The average blink duration ranges from 100 to 400 ms [8]. The average blink rate can be 12 to 19 blinks per minute at rest. This rate can be influenced by environmental and physical factors. Environmental factors include temperature, relative humidity, and brightness. The physical factors include the health of the eyes, activity level, level of cognitive workload, or fatigue.

2.3. Saccades

Saccades can be represented as the movement of the eyes while viewing a visual scene. During this activity, simultaneous movement of both eyes is seen. Saccade duration ranges from 300 to 400 ms [33]. During this period, the eyes abruptly change the point of fixation. Saccades play an important role in reading. They allow our eyes to move from one word and one line to the next. Saccades can be detected using eye-tracking technologies [20].

2.4. Fixations

Fixation is the stationary state of the eyes. Fixations represent the interval between two saccades. Fixation time is the time to focus after stopping the eyeball, which ranges from 100 to 200 ms [8]. This movement helps to construct a continuous and coherent visual perception. For example, during reading, fixations occur at each word in a text and process the visual information within each.

3. Methodology

A simplified block diagram of the complete system is illustrated in Figure 1. In this work, two-channel EOG is taken as input. The raw EOG data is then processed to detect blinks. As EOG contains a noise of baseline drift and measurement circuit, the raw data needs to be preprocessed to achieve usable signals. Thus, at the first stage of processing, filters are used to eliminate these noises. The high pass filter is dedicated to eliminating the baseline drift. The low pass filter allows the desired frequency components to be passed. The next stage is feature extraction. Statistical features are measured in this stage. These features are then used for classification. The classification is performed utilizing a linear support vector machine (SVM) algorithm. In this work, the EOG processor is primitively simulated in software. Then, the design is realized into hardware with Xilinx FPGA. During hardware implementation, both complexity reduction and speeding up are taken into account and traded off.

3.1. Dataset

In our design, the test EOG signal is obtained from the eye movement EOG dataset [34]. It contains blink and saccade information on a total of six subjects. The EOG signals are recorded with a sampling frequency of 256 Hz. The EOG data consists of horizontal and vertical EOG signals. The data was collected from a standard setup of electrodes as shown in Figure 1. The electrode pair (red) attached close to the lateral corners near the eyes provides the horizontal EOG, EOG_h . The electrode pair (green) placed above and under the right eye provides vertical EOG, EOG_v . There are ground electrode, G on the forehead, and a reference electrode, R behind the left ear.

The horizontal EOG, EOG_h and vertical EOG, EOG_v can be represented as given in Equations (1) and (2).

$$EOG_h(t) = V_3(t) - V_4(t) \quad (1)$$

$$EOG_v(t) = V_1(t) - V_2(t) \quad (2)$$

Here, $V_x(t)$ denotes the EOG potential recorded using electrode x where, $x = 1, 2, 3, 4$. The duration of one trial of EOG signal is 4 s, containing blink for 2 s and saccade for 2 s. For each subject, a total of 300 such trials were recorded in three separate sessions, specifically with 100 trials being recorded in each session. The raw signals of this dataset

represent voltage signals having an amplitude in the range of -6000 to $6000 \mu\text{V}$ range. The signals are normalized before use.

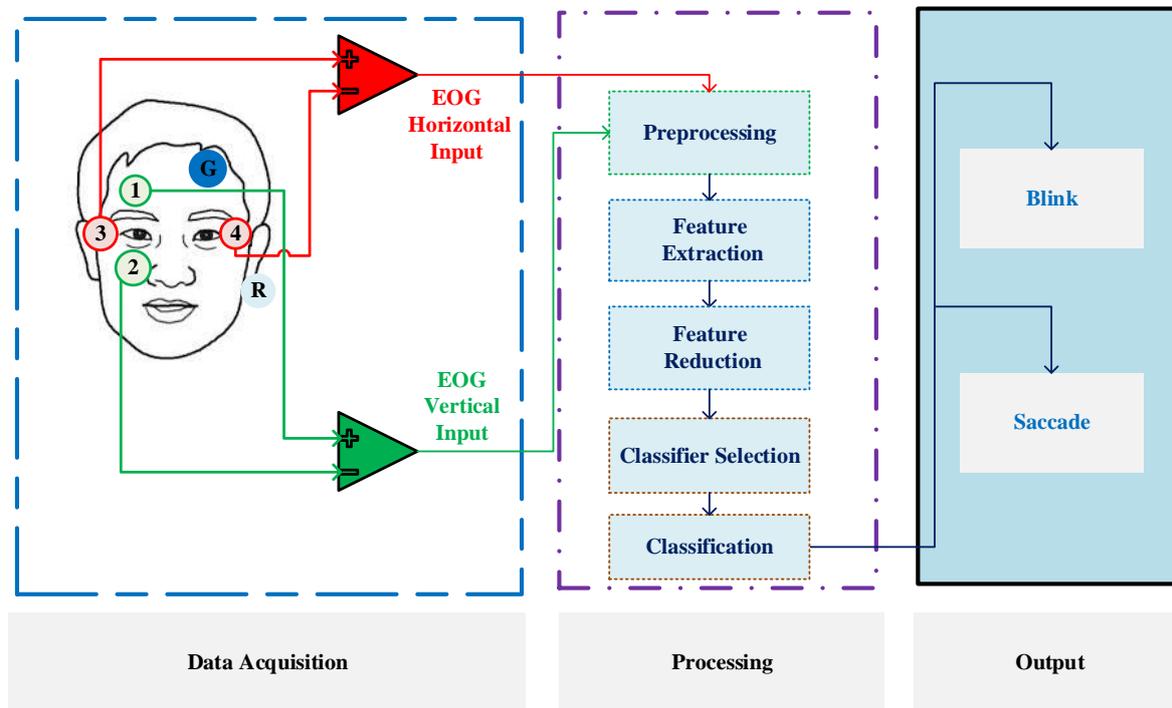


Figure 1. Simplified block diagram of the complete system.

3.2. Simulation

In this work, the EOG processor is first simulated in MATLAB. The processing is done in three stages: preprocessing, feature extraction, and classification. For efficient detection of the eye blinks, various approaches, such as variable mode extraction (VME) [35] and moving standard deviation (MSD) [36] can be adopted in the software backend. However, for fulfilling the aim of implementing the software design in hardware, features, and classifiers with simple structures are suggested to be used [37].

3.2.1. Preprocessing

Preprocessing is the first stage of EOG processor design. It is necessary to prepare the data before sending it to the application stage as the raw data suffers from some unwanted noises. In this work, the obtained raw EOG data is preprocessed using two finite impulse response (FIR) filters: a high pass filter (HPF) with 0.03 Hz cutoff frequency, and a low pass filter (LPF) with 30 Hz cutoff frequency as shown in Figure 1. The HPF is a constrained-least-square filter with an order of 26. Constrained-least-square filters are chosen as this type of filtering technique is feasible to filter out the baseline wander noise [38]. The LPF is an equiripple filter with an order of 25. The equiripple method is chosen as it meets the specifications without overperforming [39]. This method shows equal ripple in the pass band and stop band. Therefore, in case of the lowpass filter, the equiripple filter is adopted. The filters employed are not zero-phase filters. The orders of the filters are chosen with minimum order to obtain the desired frequency response. The high pass filter is capable of baseline drift mitigation [34]. The low pass filter denoised the raw signal to achieve the useful EOG component, which lies below 30 Hz. The magnitude response of the high pass filter and the low pass filter shows satisfactory results per the requirements [10].

3.2.2. Feature Extraction

The preprocessed horizontal and vertical EOG data are utilized for selecting features. The preprocessed data of six subjects are segmented into small epochs. These epochs either carry saccade or blink movement. Features of three types: time-domain features, frequency-domain features, and time-frequency hybrid features can be extracted from electrophysiological signals. As this work opts for hardware-software codesign of blink detection from EOG signals, the features that require fewer hardware resources for implementations need to be chosen [40]. Thus, only time-domain features, i.e., simple statistical features are used. At first, for feature selection, several statistical features, such as mean, mean absolute deviation, standard deviation, root mean square, kurtosis, and skewness for both channels' EOG signals have been checked. Then, we have chosen root mean square (RMS) and standard deviation (STD) as our desired features using the MRMR algorithm [41].

The root-mean-square is a statistical measure used to find the average value of a set of values. It is often used in the context of signal processing, statistics, and mathematics. It is particularly useful for calculating the root average of a set of values, especially when dealing with varying magnitudes. The root-mean-square level of a vector x containing N scalar observations, can be found using Equation (3).

$$x_{RMS} = \sqrt{\frac{1}{N} \sum_{i=1}^N |x_i|^2} \quad (3)$$

The standard deviation is a statistical measure of the amount of variation or dispersion in a set of values. It quantifies how spread out or how much the values in a data set differ from the mean (average) of that data set. It informs how much individual data points deviate from the mean. For a random variable vector x containing N scalar observations, the standard deviation is defined as Equation (4).

$$S = \sqrt{\frac{1}{N-1} \sum_{i=1}^N |x_i - \mu|^2} \quad (4)$$

where μ is the mean of x as given in Equation (5).

$$\mu = \frac{1}{N} \sum_{i=1}^N x_i \quad (5)$$

The above-mentioned features are calculated from preprocessed epoch data of all six subjects. These features are then utilized for classification.

3.2.3. Classification

In this work, binary classification is used for detecting blinks and saccades. In this work, the Support vector machine (SVM) algorithm is utilized for classification. Linear SVM has been chosen for its efficient behavior by providing the desired accuracy that outperforms other known classifiers [42]. In hardware implementation, linear SVM needs fewer resources than other popular classifiers, such as the tree classifier and the KNN classifier. Thus, the SVM algorithm is selected for classification. SVM is one of the most efficient machine learning algorithms, which has been mostly used for pattern recognition [43]. It works relatively well when there is a clear margin of separation between the two classes. SVM works by mapping data to a high-dimensional feature space so that training data points can be categorized, even when the data are not linearly separable. A separator between the categories of the training data is found, and then the data are transformed so that the separator can be drawn as a hyperplane. The features of test data with assigned labeling can be used to predict the group of it. Here, in this work, the calculated features

are used in the linear SVM classifier. We have used the 80:20 training: testing method to train the classifier.

The scoring function utilized in the SVM binary classification algorithm is given in Equation (6).

$$f(x) = x'\beta + b \tag{6}$$

where: x is an observation. The vector β contains the coefficients that define an orthogonal vector to the hyperplane, and b represents the bias term. Here, the training features are used as vector x , and the weight values, β and bias term are found from the trained model in MATLAB. The classifier then provides the output either as +1 or -1. According to the assigned labeling, the classifier detects the eye movement either as blink or saccade.

3.3. FPGA Implementation

Our proposed EOG processor design is realized in hardware using ZedBoard. It is a low-cost development board that uses Xilinx Artix-7 FPGAs [27]. The block diagram of the FPGA-implemented system is depicted in Figure 2. It takes horizontal EOG signal and vertical EOG signal as input and provides blink/saccade information after processing. The FPGA-implemented system contains three subsystems.

- Preprocessing Subsystem
- Feature Extraction Subsystem
- Classification Subsystem

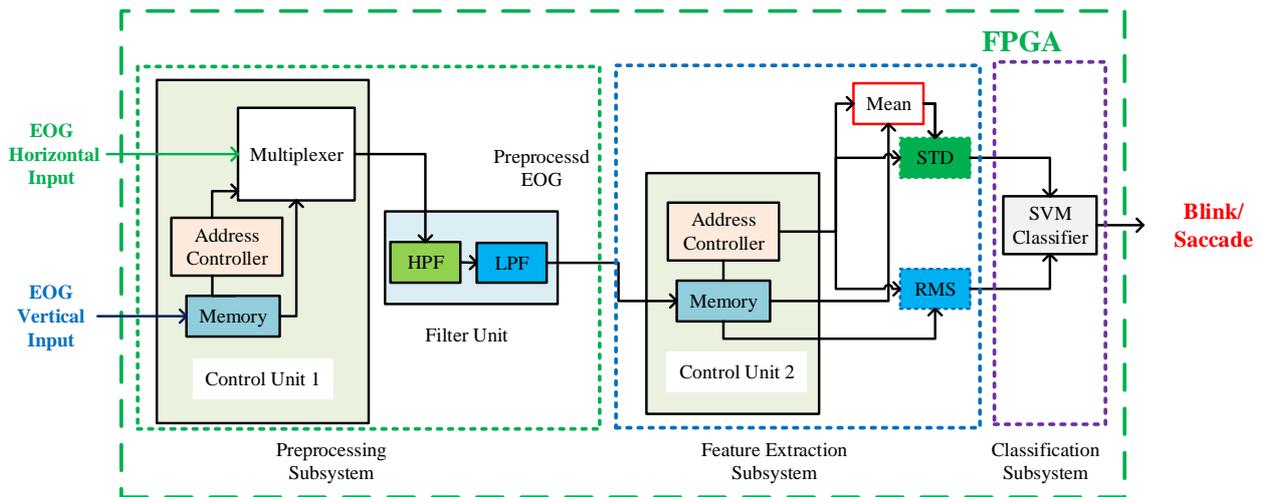


Figure 2. Block diagram of the FPGA implemented system.

3.3.1. Preprocessing Subsystem

Figure 3 depicts the block diagram of the preprocessing subsystem. It consists of a control unit (red) and a filter unit (black). The control unit takes horizontal data $EOG_h(i)$ and vertical data $EOG_v(i)$ as input and passes sequentially to the filter unit using a multiplexer, Mux 1. The output of the control unit $Mux_out(i)$ is given as input for the filter unit. The control unit block has two up counters: Counter_Slow and Counter_Fast. These counters work at 256 Hz and 2.56 MHz, respectively. The random-access memory (RAM) block stores the vertical data while horizontal data is preprocessed in the filter unit. The vertical data stored in the RAM is passed sequentially. Multiplexer 2 generates addresses for the memory block, RAM.

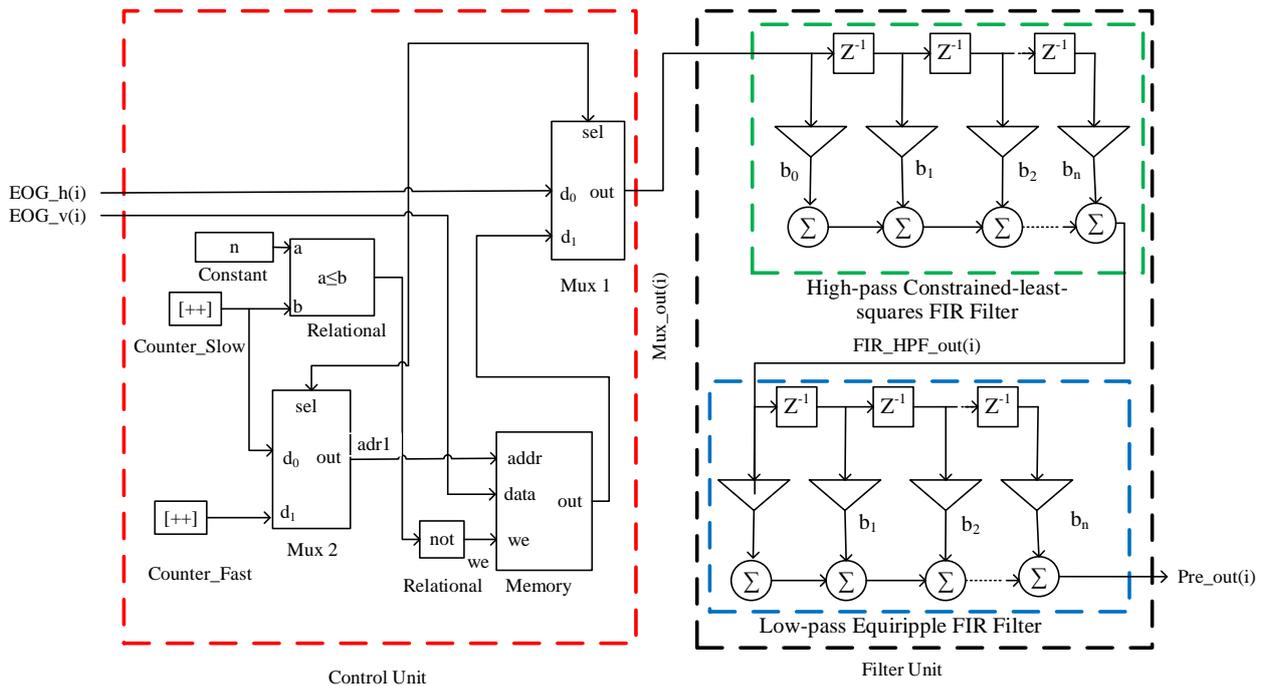


Figure 3. Block diagram of the preprocessing subsystem.

The filter unit contains two filters designed using Matlab FDATool. The first filter is a high pass FIR constrained-least-squares filter with a 0.03 Hz cutoff frequency. The second filter is a low pass FIR equiripple filter with a 30 Hz cutoff frequency. The order of the filters is set as 26 and 25 respectively as mentioned in Section 3.2. The preprocessing subsystem provides $Pre_out(i)$ as output.

3.3.2. Feature Extraction Subsystem

The $Pre_out(i)$ signal from the preprocessing subsystem is given as input in the feature extraction subsystem. In the feature extraction subsystem, we extract two features, root mean square (RMS) and standard deviation (STD).

Figure 4 demonstrates the internal formation of the RMS calculator. The RMS features can be implemented using Equation (7).

$$RMS = \sqrt{\frac{1}{N} \sum_{i=1}^N |Pre_out(i)|^2} \tag{7}$$

Here, $Pre_out(i)$ represents a preprocessed EOG epoch. N is the no. of observations in $Pre_out(i)$. In the RMS calculator block, the preprocessed EOG epoch, $Pre_out(i)$, is squared. Then, the average of the squared term is calculated consecutively using a multiplier, an accumulator, and a divisor circuit. The RMS is obtained by the square root of this average squared term. The RMS_H and RMS_V are calculated serially and stored in the register. In the $Pre_out(i)$ signal, both the horizontal and vertical data are present. The control signal $Ctrl_rms(i)$ helps in maintaining coherence in the process. It provides the signals to calculate RMS_H and RMS_V using the same unit. $en(1)$ and $en(2)$ signals helps to store the RMS_H and RMS_V features in two different registers. In this way, the features are extracted using one unit of RMS feature extractor, which saves resources.

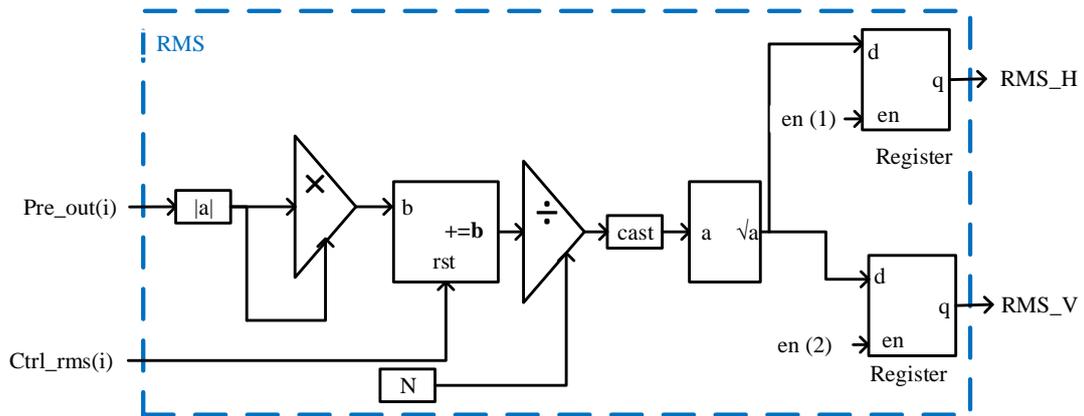


Figure 4. Block diagram of the RMS calculator in feature extraction subsystem.

Figure 5 illustrates the internal formation of the STD calculator. The STD features of all EOG epochs are extracted using Equation (8).

$$STD = \sqrt{\frac{1}{N-1} \sum_{i=1}^N |Pre_out(i) - \mu|^2} \tag{8}$$

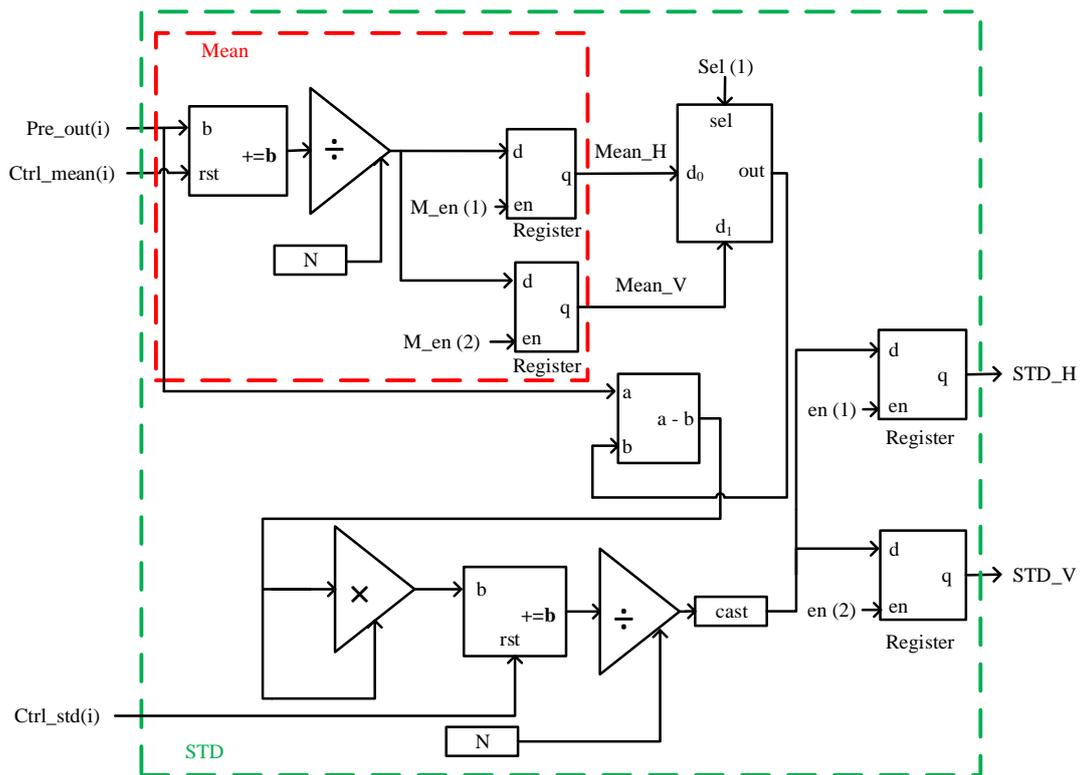


Figure 5. Block diagram of the STD calculator in feature extraction subsystem.

Here, μ represents the mean. For the hardware implementation, considering the absolute value of epoch amplitude, mean can be calculated using Equation (9).

$$Mean, \mu = \frac{1}{N} \sum_{i=1}^N Pre_out(i) \tag{9}$$

In the STD calculator block, we first determine the mean, μ . The absolute value of each preprocessed EOG epoch observation is accumulated and divided by the total number of observations to determine the mean, μ . Subtraction of the mean from $Pre_out(i)$ provides deviation. Then, the average of the squared deviation is calculated consecutively using a multiplier, an accumulator, and a divisor circuit. The STD is obtained by the square root of this average squared deviation. The STD_H and STD_V are determined serially. These values are stored in two registers. Here, the control signal, $Ctrl_std(i)$, maintains the coherency of the operation for horizontal and vertical EOG data.

3.3.3. Classification Subsystem

As mentioned in Section 3.2, we have chosen linear SVM for classification. SVM is a simple machine-learning technique that avoids complexity during hardware implementation.

Figure 6 shows the FPGA implementation of the SVM classifier. It can be translated in hardware using (10).

$$f(x) = \mathbf{x}'\mathbf{w} + b \tag{10}$$

where, \mathbf{x} contains the features. The vector \mathbf{w} contains the coefficients or weights that define an orthogonal vector to the hyperplane, and b is the bias term. In this case, \mathbf{x} contains four features. \mathbf{w} contains the weights $W1-W4$, and b is the constant bias. In the classifier block, at first, the features are multiplied by the weights. Then, weighted features are added with bias and results $f(x)$. Then, the $f(x)$ value is passed through the signum function. The function here simply gives the sign for the given values of $f(x)$ using Equation (11).

$$class = \frac{f(x)}{|f(x)|} \tag{11}$$

The signum function is realized using a divisor circuit. The output of this classifier subsystem of Figure 6 is class. For an $f(x)$ value greater than zero, the value of the class is +1 (blink), and for an $f(x)$ value lesser than zero, the value of the class is -1 (saccade).

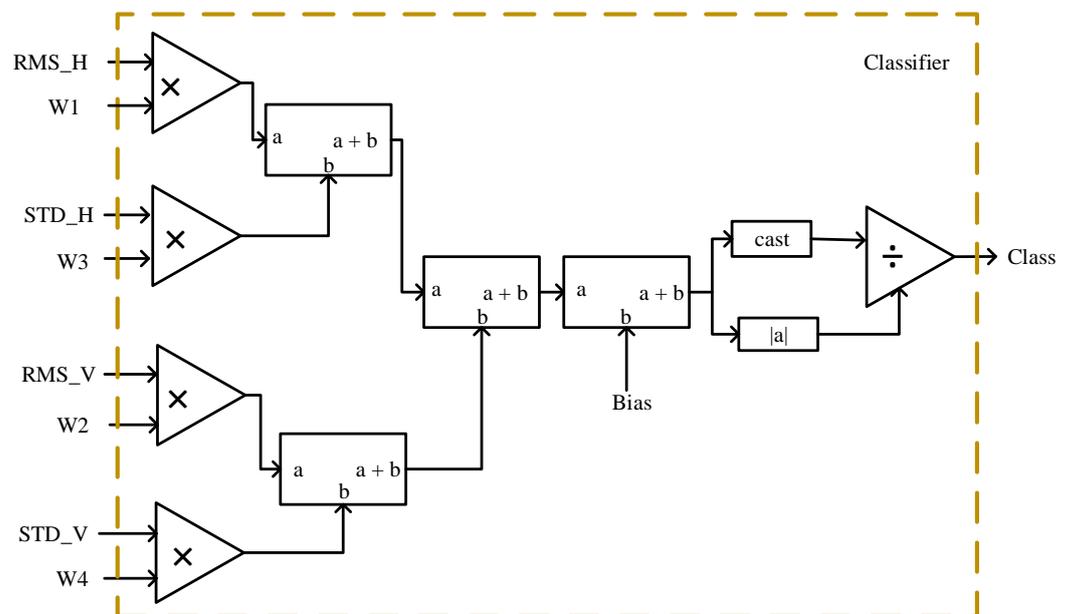


Figure 6. Block diagram of the classifier subsystem.

4. Result Analysis

This section demonstrates software and hardware results. Figures 7 and 8 show input and superimposed software-hardware results for only subject S_1 . In both figures, the data are represented in normalized form.

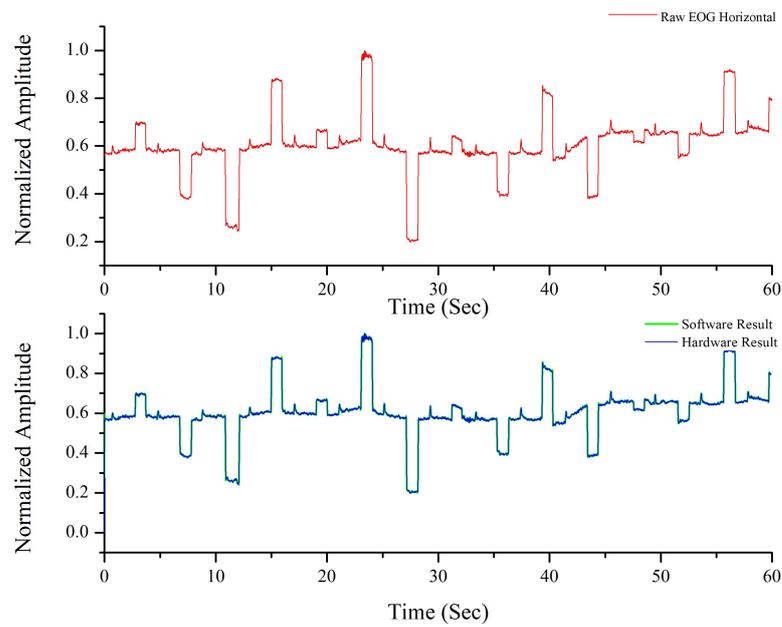


Figure 7. Raw and Preprocessed Horizontal EOG Signal.

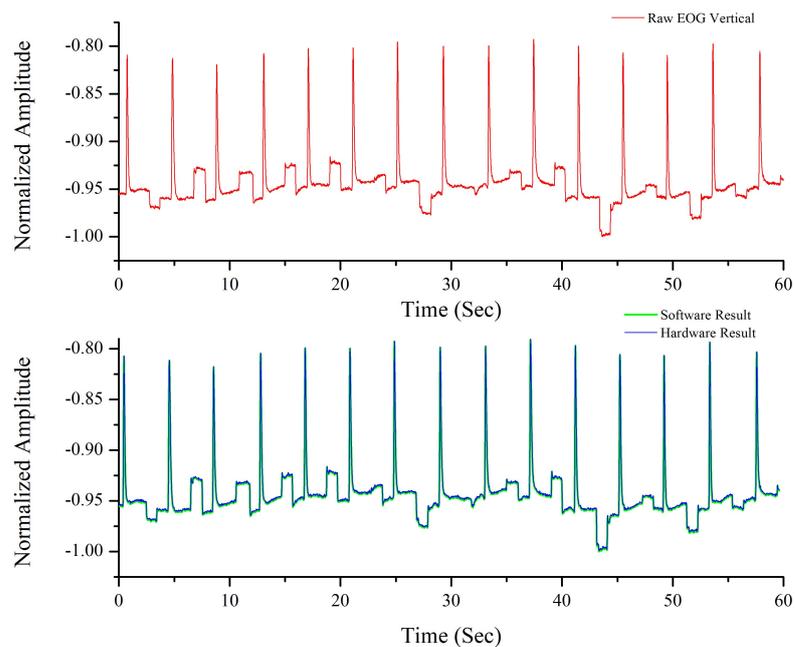


Figure 8. Raw and Preprocessed Vertical EOG Signal.

Figure 9 depicts the confusion matrix of the binary classifier. The number of true positive (TP), false positive (FP), true negative (TN), and false negative (FN) cases is 744, 27, 309, and 0, respectively. Here, TP represents the cases where a blink is detected as a blink, FN represents the cases where the blink is detected as a saccade, TN represents the cases where a saccade is detected as a saccade, and FP represents the cases where a saccade is detected as a blink.

Confusion Matrix

True Class	1	744	0
	-1	27	309
		1	-1
		Predicted Class	

Figure 9. Confusion Matrix for Binary Classifier of Blink Detection.

Several parameters, such as accuracy, precision, recall, and F1 score, are calculated to verify and analyze the system performance of the proposed classifier.

Accuracy is a measurement statistic that compares the proportion of accurate predictions made by a model to total predictions. It can be determined by Equation (12).

$$Accuracy = \frac{TN + TP}{TN + TP + FN + FP} * 100 \tag{12}$$

Sensitivity means how well a machine learning model can identify positive examples. It is also known as Recall. It can be expressed by Equation (13).

$$Sensitivity = \frac{TP}{TP + FN} * 100 \tag{13}$$

Specificity refers to how an algorithm or model can forecast a true negative for each possible category. It is defined by Equation (14).

$$Specificity = \frac{TN}{TN + FP} * 100 \tag{14}$$

Precision represents the quality of a positive prediction made by the mode It is defined by Equation (15).

$$Precision = \frac{TP}{TP + FP} * 100 \tag{15}$$

The F1 score can be calculated as the harmonic mean of the precision and recall scores as given in Equation (16).

$$F1\ score = \frac{TP}{TP + 0.5 * (FP + FN)} * 100 \tag{16}$$

The accuracy, sensitivity, specificity, precision, and F1 score found are 97.5%, 100%, 91.64%, 96.5%, and 98.22% respectively.

The proposed model is implemented on Xilinx Artix-7 FPGA. The accuracy of the implemented EOG processor is 95%. The eyeblink detection latency is 0.29 μs.

For qualitative analysis, the software and hardware results are compared using the Pearson correlation coefficient and root square error (RSE). These parameters can be used to verify the software-hardware agreement [37]. Pearson Correlation Coefficient, r , is calculated using the equation given in (17). Then, RSE is calculated as given in (18).

$$r = \frac{1}{n - 1} \left(\frac{\sum_X \sum_Y (X - \bar{X})(Y - \bar{Y})}{S_X S_Y} \right) \tag{17}$$

$$RSE = \sqrt{(X - Y)^2} \tag{18}$$

Here, n denotes the total number of paired data. X and Y are the simulation and hardware outputs, respectively. The averages of these corresponding data are presented by \bar{X} and \bar{Y} , respectively. S_X and S_Y denote the corresponding standard deviations.

The Pearson correlation coefficient is 0.98. The root mean square value is found in the 10^{-3} range. These values represent that our prototype agrees with the software model.

Table 1 shows the FPGA resource utilization for this design. Usage of Look Up Table (LUT), LUT-Random-Access Memory (LUTRAM), Flip-Flops (FF), Blocked Random-Access Memory (BRAM), Digital Signal Processing (DSP), Bonded Input/Output blocks (IO) and Global Buffer (BUFG) are 37%, 4%, 4%, 1%, 54%, 46%, and 3% respectively.

Table 1. FPGA Resource Utilization.

Resource	Utilization	Available	Percentage Utilization
LUT	19,684	53,200	37%
LUTRAM	696	17,400	4%
FF	4256	106,400	4%
BRAM	1	140	1%
DSP	119	220	54%
IO	92	200	46%
BUFG	1	32	3%

Power consumption in Zedboard consists of two major components. These are static power and dynamic power. Static or standby power refers to the total amount of power the device consumes when it is powered up but not actively performing any operation. Dynamic or active power refers to the total amount of power the device consumes when it is actively operating. As shown in Table 2, our prototype uses a total of 0.8 W power only. Dynamic power consumption is 0.684 W (86%). The dynamic power is utilized in Clocks (0.029 W), Signals (0.306 W), Logic (0.241 W), BRAM (0.007 W), DSP (0.076 W), and I/O (0.025 W). The static power consumption is 0.116 watts (14%).

Table 2. FPGA Power Consumption.

	Logic Operation	Power Consumption
Dynamic (86%)	Clocks	0.029 W (4%)
	Signals	0.306 W (45%)
	Logic	0.241 W (35%)
	BRAM	0.007 W (1%)
	DSP	0.076 W (11%)
	IO	0.025 W (4%)
Device Static (14%)	-	0.0116 W
Total On-Chip power		0.8 W

Table 3 compares this work and state-of-the-art works of EOG-based blink detection. Banerjee et al. [19] achieved 95.33% accuracy in blink detection with RBF kernel SVM. Ryu et al. [20] achieved 94.3% accuracy in detecting blinks, horizontal saccades, vertical saccades, and fixation with differential EOG signal based on a fixation curve (DOSbFC) method for baseline drift and noise removal. Molina-Cantero et al. [1] achieved 89.9% accuracy with adaptive K-means for classifying single blink, double blink, and long blink. Gundugonti and Narayanam used thresholding for the blink and saccade detection and implemented it in FPGA [22]. The first and second works of Table 3 were carried out at the software level. The third work adopted implementation in the discrete circuit only for the preprocessing stage. The fourth work implemented DWT and thresholding approach in FPGA.

Table 3. Comparative study of state-of-the-art works of EOG based blink detection.

References	Detection Approach	Accuracy	Implementation			Device
			Preprocessing	Feature Extraction	Classification	
Banerjee et al. [19]	RBF kernel SVM	95.33%	×	×	×	×
Ryu et al. [20]	DOSbFC and Thresholding	94.3%	×	×	×	×
Molina-Cantero et al. [1]	Adaptive K-means	89.9%	✓	×	×	Discrete
Gundugonti and Narayanam [22]	DWT and Thresholding	-	✓	×	✓	FPGA
Proposed work	Linear SVM	97.5% * and 95% **	✓	✓	✓	FPGA

* Software Accuracy, ** Hardware Accuracy.

The proposed design has better accuracy than all state-of-the-art works. This work demonstrates a machine learning based reconfigurable implementation of a dual-channel EOG processor incorporating preprocessing, feature extraction, and classification stages for blink detection.

5. Conclusions

In this paper, We propose an application-specific Electrooculogram (EOG) processor to detect blinks. The EOG signals are preprocessed using FIR filters of minimum order. Statistical features are extracted and then used for classification. The proposed architecture detects the saccades and blinks efficiently with software and hardware accuracies of 97.5% and 95%, respectively. Hardware resource utilization for different units and power consumption are presented. The on-chip power consumption for this design is only 0.8 watts. Experimental results demonstrate the designed EOG processor offers efficacy in terms of classification accuracy, implementation complexity, and power consumption. The prototype can be integrated into smart cars as a driver monitoring system. The detected blink from the proposed design can be employed to provide security as an access controller in IoT devices. This work can also be extended to real-life applications by focusing on reliability and cost-effectiveness. The system can be employed in online applications such as e-health systems that will benefit the health care of paralyzed people.

Author Contributions: Conceptualization, D.D. and M.H.C.; methodology, D.D. and M.H.C.; software, D.D. and K.H.; validation, D.D. and A.C.; formal analysis, D.D. and A.C.; investigation, D.D.; resources, R.C.C.C., Q.D.H. and M.H.C.; writing—original draft preparation, D.D., A.C. and K.H.; writing—review and editing, D.D., M.H.C., Q.D.H. and R.C.C.C.; visualization, D.D.; supervision, R.C.C.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Eye movement EOG dataset [34] has been used in this work.

Acknowledgments: CityU Architecture Lab for Arithmetic and Security (CALAS), City University of Hong Kong, Kowloon, Hong Kong has supported to implementation of this work.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Molina-Cantero, A.J.; Lebrato-Vazquez, C.; Merino-Monge, M.; Quesada-Tabares, R.; Castro-Garcia, J.A.; Gomez-Gonzalez, I.M. Communication technologies based on voluntary blinks: Assessment and design. *IEEE Access* **2019**, *7*, 70770–70798. [[CrossRef](#)]
2. Zhang, R.; He, S.; Yang, X.; Wang, X.; Li, K.; Huang, Q.; Yu, Z.; Zhang, X.; Tang, D.; Li, Y. An EOG-Based Human-Machine Interface to Control a Smart Home Environment for Patients with Severe Spinal Cord Injuries. *IEEE Trans. Biomed. Eng.* **2019**, *66*, 89–100. [[CrossRef](#)] [[PubMed](#)]
3. Hernández Pérez, S.N.; Pérez Reynoso, F.D.; Gutiérrez, C.A.G.; Cosío León, M.D.I.Á.; Ortega Palacios, R. EOG Signal Classification with Wavelet and Supervised Learning Algorithms KNN, SVM and DT. *Sensors* **2023**, *23*, 4553. [[CrossRef](#)] [[PubMed](#)]

4. He, S.; Zhou, Y.; Yu, T.; Zhang, R.; Huang, Q.; Chuai, L.; Mustafa, M.-U.; Gu, Z.; Yu, Z.L.; Tan, H. EEG- And EOG-Based Asynchronous Hybrid BCI: A System Integrating a Speller, a Web Browser, an E-Mail Client, and a File Explorer. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2020**, *28*, 519–530. [[CrossRef](#)] [[PubMed](#)]
5. Minati, L.; Yoshimura, N.; Koike, Y. Hybrid Control of a Vision-Guided Robot Arm by EOG, EMG, EEG Biosignals and Head Movement Acquired via a Consumer-Grade Wearable Device. *IEEE Access* **2016**, *4*, 9528–9541. [[CrossRef](#)]
6. Wu, J.F.; Ang, A.M.S.; Tsui, K.M.; Wu, H.C.; Hung, Y.S.; Hu, Y.; Mak, J.N.F.; Chan, S.C.; Zhang, Z.G. Efficient Implementation and Design of a New Single-Channel Electrooculography-Based Human–Machine Interface System. *IEEE Trans. Circuits Syst. II Express Briefs* **2015**, *62*, 179–183. [[CrossRef](#)]
7. Wu, S.-L.; Liao, L.-D.; Lu, S.-W.; Jiang, W.-L.S.; Chen, S.-A.; Lin, C.-T. Controlling a human–computer interface system with a novel classification method that uses electrooculography signals. *IEEE Trans. Biomed. Eng.* **2013**, *60*, 2133–2141. [[CrossRef](#)]
8. Bulling, A.; Ward, J.A.; Gellersen, H.; Tröster, G. Eye movement analysis for activity recognition using electrooculography. *IEEE Trans. Pattern. Anal. Mach. Intell.* **2011**, *33*, 741–753. [[CrossRef](#)]
9. López, A.; Ferrero, F.; Yangüela, D.; Álvarez, C.; Postolache, O. Development of a Computer Writing System Based on EOG. *Sensors* **2017**, *17*, 1505. [[CrossRef](#)]
10. Das, D.; Chowdhury, A.; Sanka, A.I.; Chowdhury, M.H. Design and Performance Evaluation of an FPGA based EOG Signal Preprocessor. In Proceedings of the 2023 International Conference on Electrical, Computer and Communication Engineering (ECCE), Chittagong, Bangladesh, 23–25 February 2023; pp. 1–6. [[CrossRef](#)]
11. Lin, C.-T.; King, J.-T.; Bharadwaj, P.; Chen, C.-H.; Gupta, A.; Ding, W.; Prasad, M. EOG-Based Eye Movement Classification and Application on HCI Baseball Game. *IEEE Access* **2019**, *7*, 96166–96176. [[CrossRef](#)]
12. Zhang, Y.; Zheng, X.; Xu, W.; Liu, H. RT-Blink: A Method Toward Real-Time Blink Detection from Single Frontal EEG Signal. *IEEE Sensors J.* **2023**, *23*, 2794–2802. [[CrossRef](#)]
13. Kołodziej, M.; Tarnowski, P.; Sawicki, D.J.; Majkowski, A.; Rak, R.J.; Bala, A.; Pluta, A. Fatigue Detection Caused by Office Work With the Use of EOG Signal. *IEEE Sensors J.* **2020**, *20*, 15213–15223. [[CrossRef](#)]
14. Hayawi, A.A.; Waleed, J. Driver’s drowsiness monitoring and alarming auto-system based on eog signals. In Proceedings of the 2019 2nd International Conference on Engineering Technology and its Applications (IICETA), Najaf, Iraq, 27–28 August 2019; pp. 214–218. [[CrossRef](#)]
15. Lapa, I.; Ferreira, S.; Mateus, C.; Rocha, N.; Rodrigues, M.A. Real-Time Blink Detection as an Indicator of Computer Vision Syndrome in Real-Life Settings: An Exploratory Study. *Int. J. Environ. Res. Public Health* **2023**, *20*, 4569. [[CrossRef](#)] [[PubMed](#)]
16. Shahbakhti, M.; Beiramvand, M.; Rejer, I.; Augustyniak, P.; Broniec-Wojcik, A.; Wierz, M.; Marozas, V. Simultaneous Eye Blink Characterization and Elimination from Low-Channel Prefrontal EEG Signals Enhances Driver Drowsiness Detection. *IEEE J. Biomed. Heal. Inform.* **2022**, *26*, 1001–1012. [[CrossRef](#)] [[PubMed](#)]
17. Ghosh, R.; Phadikar, S.; Deb, N.; Sinha, N.; Das, P.; Ghaderpour, E. Automatic Eyeblink and Muscular Artifact Detection and Removal From EEG Signals Using k-Nearest Neighbor Classifier and Long Short-Term Memory Networks. *IEEE Sensors J.* **2023**, *23*, 5422–5436. [[CrossRef](#)]
18. Maddirala, A.K.; Veluvolu, K.C. SSA with CWT and k-Means for Eye-Blink Artifact Removal from Single-Channel EEG Signals. *Sensors* **2022**, *22*, 931. [[CrossRef](#)] [[PubMed](#)]
19. Banerjee, A.; Pal, M.; Tibarewala, D.N.; Konar, A. Electrooculogram based blink detection to limit the risk of eye dystonia. In Proceedings of the 2015 Eighth International Conference on Advances in Pattern Recognition (ICAPR), Kolkata, India, 4–7 January 2015; pp. 1–6. [[CrossRef](#)]
20. Ryu, J.; Lee, M.; Kim, D.H. EOG-based eye tracking protocol using baseline drift removal algorithm for long-term eye movement detection. *Expert Syst. Appl.* **2019**, *131*, 275–287. [[CrossRef](#)]
21. Gundugonti, K.K.; Narayanam, B. Efficient Haar Wavelet Transform for Detecting Saccades and Blinks in Real-Time EOG Signal. *SN Comput. Sci.* **2021**, *2*, 156. [[CrossRef](#)]
22. Gundugonti, K.K.; Narayanam, B. FPGA implementation of eye movement detection algorithm. *Microprocess. Microsyst.* **2021**, *in press*. [[CrossRef](#)]
23. Prakashan, D.; Ramya, P.R.; Gandhi, S. A Systematic Review on the Advanced Techniques of Wearable Point-of-Care Devices and Their Futuristic Applications. *Diagnostics* **2023**, *13*, 916. [[CrossRef](#)]
24. Chowdhury, A.; Das, D.; Cheung, R.C.C.; Chowdhury, M.H. Hardware/Software Co-design of an ECG- PPG Preprocessor: A Qualitative & Quantitative Analysis. In Proceedings of the 2023 International Conference on Electrical, Computer and Communication Engineering (ECCE), Chittagong, Bangladesh, 23–25 February 2023; pp. 1–6. [[CrossRef](#)]
25. Lin, W.; Zhu, Y.; Arslan, T. DycSe: A Low-Power, Dynamic Reconfiguration Column Streaming-Based Convolution Engine for Resource-Aware Edge AI Accelerators. *J. Low Power Electron. Appl.* **2023**, *13*, 21. [[CrossRef](#)]
26. Giorgio, A.; Guaragnella, C.; Rizzi, M. FPGA-Based Decision Support System for ECG Analysis. *J. Low Power Electron. Appl.* **2023**, *13*, 6. [[CrossRef](#)]
27. Chowdhury, M.H.; Cheung, R.C.C. Reconfigurable Architecture for Multi-lead ECG Signal Compression with High-frequency Noise Reduction. *Sci. Rep.* **2019**, *9*, 17233. [[CrossRef](#)] [[PubMed](#)]
28. Conti, G.; Quintana, M.; Malagón, P.; Jiménez, D. An FPGA Based Tracking Implementation for Parkinson’s Patients. *Sensors* **2020**, *20*, 3189. [[CrossRef](#)] [[PubMed](#)]

29. Savastaer, E.F.; Tepe, C. Single Channel EOG Measurement System and Interface Design. In Proceedings of the ISMSIT 2021-5th International Symposium on Multidisciplinary Studies and Innovative Technologies, Ankara, Turkey, 21–23 October 2021; pp. 115–119. [[CrossRef](#)]
30. Pai, Y.S.; Bait, M.L.; Lee, J.; Xu, J.; Peiris, R.L.; Woo, W.; Billinghamurst, M.; Kunze, K. NapWell: An EOG-based Sleep Assistant Exploring the Effects of Virtual Reality on Sleep Onset. *Virtual Real.* **2022**, *26*, 437–451. [[CrossRef](#)]
31. Masai, K.; Kunze, K.; Sugimoto, M. Eye-based interaction using embedded optical sensors on an eyewear device for facial expression recognition. In Proceedings of the AHs '20: Proceedings of the Augmented Humans International Conference, Kaiserslautern, Germany, 16–17 March 2020. [[CrossRef](#)]
32. Li, L.; Xie, Y.; Xiong, J.; Hou, Z.; Zhang, Y.; We, Q.; Wang, F.; Fang, D.; Chen, X. Smartlens: Sensing eye activities using zero-power contact lens. In Proceedings of the Annual International Conference on Mobile Computing and Networking, MOBICOM, Sydney, Australia, 17–21 October 2022; pp. 473–486. [[CrossRef](#)]
33. Bolte, B.; Lappe, M. Subliminal reorientation and repositioning in immersive virtual environments using saccadic suppression. *IEEE Trans. Vis. Comput. Graph.* **2015**, *21*, 545–552. [[CrossRef](#)] [[PubMed](#)]
34. Barbara, N.; Camilleri, T.A.; Camilleri, K.P. A comparison of EOG baseline drift mitigation techniques. *Biomed. Signal Process. Control* **2020**, *57*, 101738. [[CrossRef](#)]
35. Shahbakhti, M.; Beiramvand, M.; Nazari, M.; Broniec-Wojcik, A.; Augustyniak, P.; Rodrigues, A.S.; Wierzchon, M.; Marozas, V. VME-DWT: An Efficient Algorithm for Detection and Elimination of Eye Blink From Short Segments of Single EEG Channel. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2021**, *29*, 408–417. [[CrossRef](#)]
36. Shahbakhti, M.; Beiramvand, M.; Nasiri, E.; Far, S.M.; Chen, W.; Sole-Casals, J.; Wierzchon, M.; Broniec-Wojcik, A.; Augustyniak, P.; Marozas, V. Fusion of EEG and Eye Blink Analysis for Detection of Driver Fatigue. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2023**, *31*, 2037–2046. [[CrossRef](#)]
37. Chowdhury, M.H.; Eldaly, A.B.M.; Agadagba, S.K.; Cheung, R.C.C.; Chan, L.L.H. Machine Learning Based Hardware Architecture for DOA Measurement from Mice EEG. *IEEE Trans. Biomed. Eng.* **2021**, *69*, 314–324. [[CrossRef](#)]
38. Egila, M.G.; El-Moursy, M.A.; El-Hennawy, A.E.; El-Simary, H.A.; Zaki, A. FPGA-based electrocardiography (ECG) signal analysis system using least-square linear phase finite impulse response (FIR) filter. *J. Electr. Syst. Inf. Technol.* **2016**, *3*, 513–526. [[CrossRef](#)]
39. Das, R.; Guha, A.; Bhattacharya, A. FPGA based higher order FIR filter using XILINX system generator. In Proceedings of the 2016 International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES), Paralakhemundi, India, 3–5 October 2016; pp. 111–115. [[CrossRef](#)]
40. Cicuttin, A.; Morales, I.R.; Crespo, M.L.; Carrato, S.; García, L.G.; Molina, R.S.; Valinoti, B.; Folla Kamdem, J. A Simplified Correlation Index for Fast Real-Time Pulse Shape Recognition. *Sensors* **2022**, *22*, 7697. [[CrossRef](#)] [[PubMed](#)]
41. Wang, G.; Lauri, F.; Hassani, A.H.E. Feature Selection by mRMR Method for Heart Disease Diagnosis. *IEEE Access* **2022**, *10*, 100786–100796. [[CrossRef](#)]
42. Afifi, S.; GholamHosseini, H.; Sinha, R. FPGA Implementations of SVM Classifiers: A Review. *SN Comput. Sci.* **2020**, *1*, 133. [[CrossRef](#)]
43. Zhang, Q.; Zhou, D. Machine Learning Electrocardiogram for Mobile Cardiac Pattern Extraction. *Sensors* **2023**, *23*, 5723. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.