Pamukkale Üniversitesi Mühendislik Bilimleri Dergisi

Pamukkale University Journal of Engineering Sciences

Investigating effects of Levenshtein distance algorithm on accuracy performance in EEG based communication system

EEG tabanlı haberleşme sisteminde Levenshtein mesafe algoritmasının doğruluk performansına etkisinin incelenmesi

Güneş EKİM[1], Nuri İKİZLER[1], Ayten ATASOY[2]*

[1]Electronics and Automation Department, Trabzon Vocational School, Karadeniz Technical University, Trabzon, Turkey.
gunesekim@ktu.edu.tr, nikizler@ktu.edu.tr

[2]Electrical-Electronics Engineering Department, Engineering Faculty, Karadeniz Technical University, Trabzon, Turkey.
ayten@ktu.edu.tr

Received/Geliş Tarihi: 03.07.2023 Revision/Düzeltme Tarihi: 14.07.2023 doi: 10.5505/pajes.2024.66263
Accepted/Kabul Tarihi: 03.09.2024 Research Article/Araştırma Makalesi

Abstract

In our study, we investigated the effects of the Levenshtein distance algorithm on the eye-blink communication system that we developed based on EEG signals for people with severe motor disabilities, such as Amyotrophic Lateral Sclerosis, stroke, and locked-in syndrome. The developed system analyzes eye-blink signals to extract information and vocalize it. EEG signals were obtained from an electrode above the left eye using a NeuroSky MindWave Mobile device. Morse-coded eye-blink words were input to the system and feature vectors were extracted using the Wavelet Transform method. Support vector machines were trained with these vectors and the Levenshtein distance algorithm was used to reduce classification errors. Finally, the system was completed with a text-to-speech synthesis algorithm. The experiments, which used 20 words for self-expression, obtained highly successful results.

Keywords: EEG, Brain-computer interface, Wavelet transform, Support vector machine, Levenshtein distance algorithm, Eye-blink detection

Öz

Çalışmamızda, amyotrofik lateral skleroz, felç ve kilitli kalma sendromu gibi motor engeli olan kişiler için, EEG sinyalleri kullanarak geliştirdiğimiz göz-kırpma iletişim sistemi üzerinde Levenshtein mesafe algoritmasının etkilerini araştırdık. Sistem, göz-kırpma sinyallerini analiz ederek bilgi çıkarmakta ve seslendirmektedir. EEG sinyalleri, kablosuz NeuroSky MindWave cihazı kullanılarak sol gözün üzerine yerleştirilmiş bir elektrottan elde edilmiştir. Göz-kırpmalarla oluşturulan Mors kodlu kelimeler sisteme giriş olarak verilmiş ve Dalgacık Dönüşümü kullanılarak öznitelik vektörleri çıkarılmıştır. Bu vektörlerle Destek Vektör Makineleri eğitilmiş ve sınıflandırma hatalarını azaltmak için Levenshtein mesafe algoritması kullanılmıştır. Son olarak, Metin-Konuşma sentezi algoritması ile sistem tamamlanmıştır. Kendini ifade etmek için 20 kelimenin kullanıldığı deneyler oldukça başarılı sonuçlar vermiştir.

Anahtar kelimeler: EEG, Beyin-bilgisayar arayüzü, Dalgacık dönüşümü, Destek vektör makineleri, Levenshtein mesafe algoritması, göz-kırpma tespiti

1 Introduction

Communication is a vital component of our daily lives, allowing us to express ourselves, convey our thoughts and feelings, and connect with others. However, individuals with conditions such as Amyotrophic Lateral Sclerosis (ALS) or stroke may experience severe motor disabilities that limit their ability to communicate effectively [1]. In recent years, researchers have explored the potential of brain-computer interface (BCI) systems to enable communication for these individuals. This is a technology that allows for direct communication between the brain and an external device, such as a computer. EEG-based eye blink communication systems can be considered a type of BCI. These systems are an emerging technology that enables individuals to communicate and interact with the world around them by measuring and interpreting electrical signals generated by the brain [2], [3]. These systems require algorithms to be trained to recognize different patterns of eye blinks by detecting electrical signals picked up by electrodes placed on the scalp. Eye blinks are a popular choice in BCI systems since they are a common and easily detectable event in EEG signals [4], [5]. By allowing algorithms to recognize different patterns of eye blinks, individuals can control a variety of devices, such as computers, wheelchairs, and prosthetic limbs, using their brain signals [6]. The use of EEG-based eye blink communication systems has great potential to benefit individuals with disabilities, such as those with paralysis or motor neuron diseases, by enabling them to control devices using only their brain signals and providing greater independence and improving overall quality of life. However, there are still many challenges to be addressed in the development of EEG-based eye blink communication systems, such as improving the accuracy and reliability of signal detection, reducing training time, and addressing issues related to user fatigue and discomfort.

These systems used dry electrodes and wireless transmission technology, making them more comfortable and accessible for users. In recent years, researchers have continued to refine and improve EEG-based eye-blink communication systems, focusing on improving accuracy and reducing error rates [7]. There has also been increasing interest in integrating these systems with other assistive technologies, such as speech synthesis and text-to-speech conversion [8]. Overall, the historical development of EEG-based eye-blink communication systems has been driven by advancements in technology and the growing recognition of the importance of providing

* Corresponding author/Yazışılan yazar

alternative communication methods for individuals with severe motor disabilities.

The conversion of eye blinks into speech is currently available in developed systems using alternative methods [9], [10]. Electrooculography (EOG) measures the electrical potential between the cornea and the retina. By attaching electrodes around the user's eyes, the system can detect eye movements, including blinks. This technology can then be used to convert those blinks into a form of communication, such as Morse code or text [11], [12]. Computer vision-based blink detection method uses computer vision algorithms to detect eye blinks by analyzing the video stream of a camera pointed at the user's face [13], [14]. This technology can then be used to convert those eye blinks into a form of communication, such as speech or text. Accelerometer-based blink detection method uses an accelerometer sensor to detect the user's head movements, which can be used to infer eye blinks [15]. This technology can then be used to convert those eye blinks into a form of communication, such as speech or text.

Eye tracking, as a technology for detecting eye movements, can be used in conjunction with some of the previously mentioned techniques for converting eye blinks into communication signals [16], [17]. For example, eye tracking can be used to determine when a person is looking at a specific location or object, and then a separate technique (such as EOG or video-based methods) can be used to detect eye blinks and convert them into communication signals. Alternatively, some eye tracking systems can also be used to detect eye blinks directly, based on changes in the pupil size or eye shape that occur during a blink [18]. However, these methods may not be as accurate or reliable as other techniques specifically designed for detecting eye blinks, especially in situations where there is a lot of eye movement or other sources of interference.

In this study, we present a novel EEG-based eye-blink communication system that uses wavelet analysis and support vector machines to convert Morse-coded eye-blink signals into speech. Our system utilizes a single-channel EEG device to record eye-blink signals and processes them using sophisticated algorithms to produce accurate and meaningful output. Moreover, to improve the system's accuracy, we propose the use of the Levenshtein distance algorithm that never tested in other studies as seen in Table 8. This algorithm can significantly reduce classification errors, enhancing the system's reliability and usefulness for individuals with severe motor disabilities.

Finally, we discuss the development and implementation of this innovative communication system and highlight the potential impact of our research on improving the quality of life for individuals with motor disabilities.

2 Materials and methods

2.1 Data acquisition

Data acquisition is one of the main steps in Brain Computer Interface (BCI) systems, which involves measuring brain activity and digitizing it for processing on a computer. Different methods can be used to measure brain activity, and Electroencephalogram (EEG) is generally preferred for BCI systems due to its practicality. EEG is a painless, easy-to-apply, low-cost, and non-invasive procedure that has no harmful effects on the person. Brain signals that are traditionally used in existing BCI systems were not directly used in our implemented system. Instead, eye-blink signals, which are often considered as a noise signal, were utilized. Although it is possible to acquire eye-blink signals using devices such as EOG and EMG instead of EEG, we chose to use a single-channel EEG device in our study to prioritize patient comfort and cost-effectiveness. The NeuroSky Mind Wave Mobile (NSMWM) device wirelessly transmits electrical activity to a computer via Bluetooth [19]. NeuroSky Mind Wave Mobile device senses the signals from the human brain, filters out extraneous noise and 50/60 Hz electrical interference The device has two sensors, one that touches the forehead above the eye and the other that is attached to the earlobe. As seen in Figure 3(b), the EEG electrode is in the FP1 position and detects electrical signals emanating from the frontal lobe of the brain. The sensor in the ear clip acts as a ground. This single-channel device has a sampling frequency of 512 Hz, does not cause any discomfort, and is very affordable.

The block diagram and the flow chart of the developed system are given in Figure 1 and Figure 2, respectively. Figure 3 (a) shows the single-channel wire-less NSMWM device, 3 (b) shows the electrode layout of the device, and 3 (c) shows the recording session of the system.

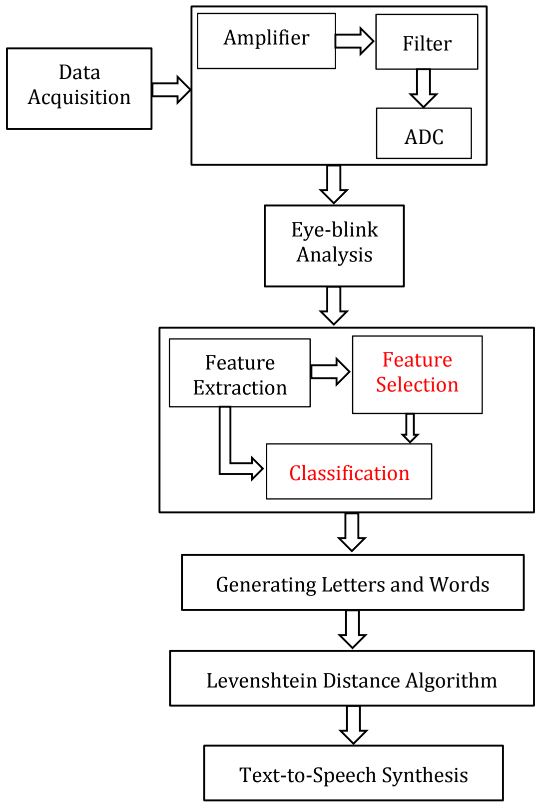

Figure 1. Block diagram of system developed in the study.

2.2 Morse coded signal input

In this study, the Morse coding technique was used for encoding eye-blink signals [20]. This technique represents short lines with a single eye-blink and long lines with double eye-blinks. For example, the letter A, as shown in Table 1, consists of one short line and one long line in Morse code, which is encoded as a single eye-blink followed by double eye-blinks.

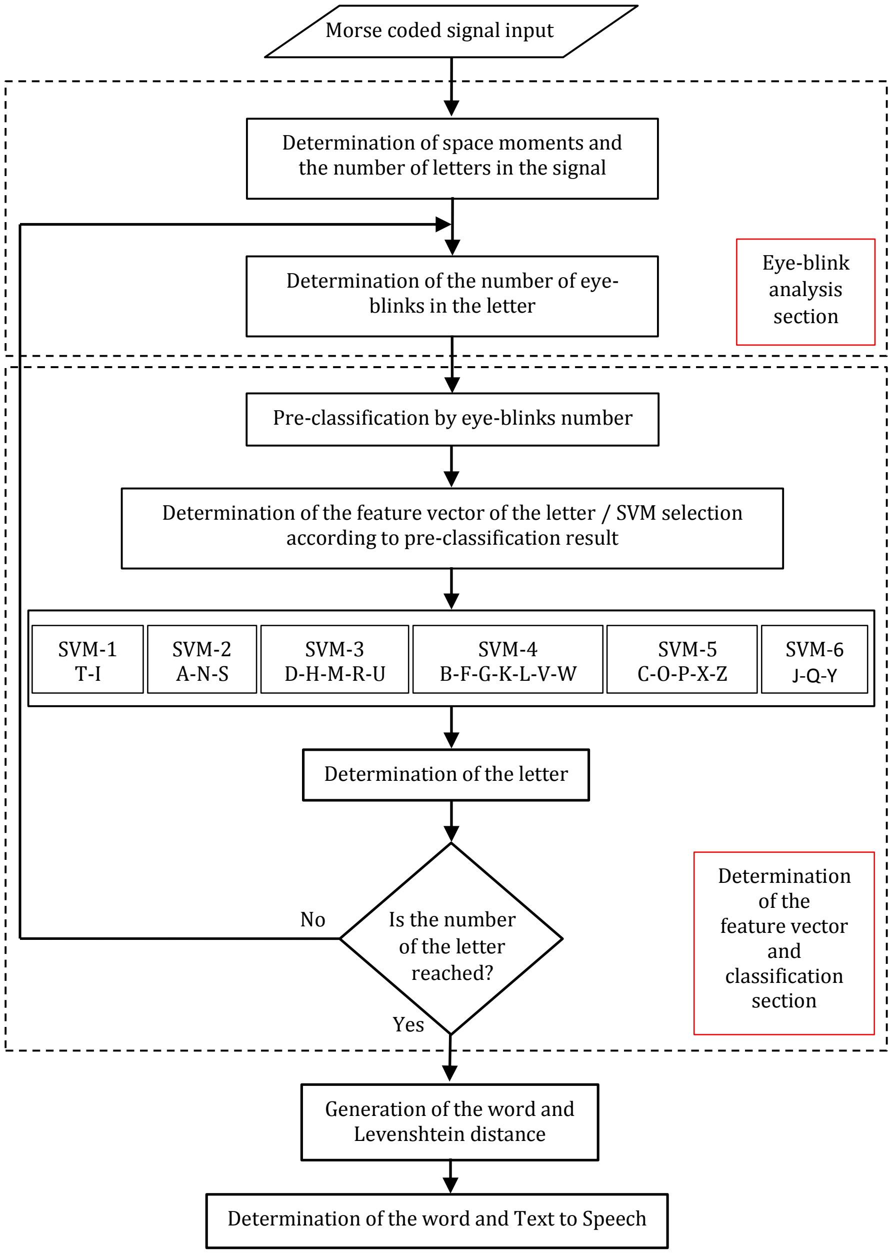

Figure 2. Flow diagram of algorithm that used for study.

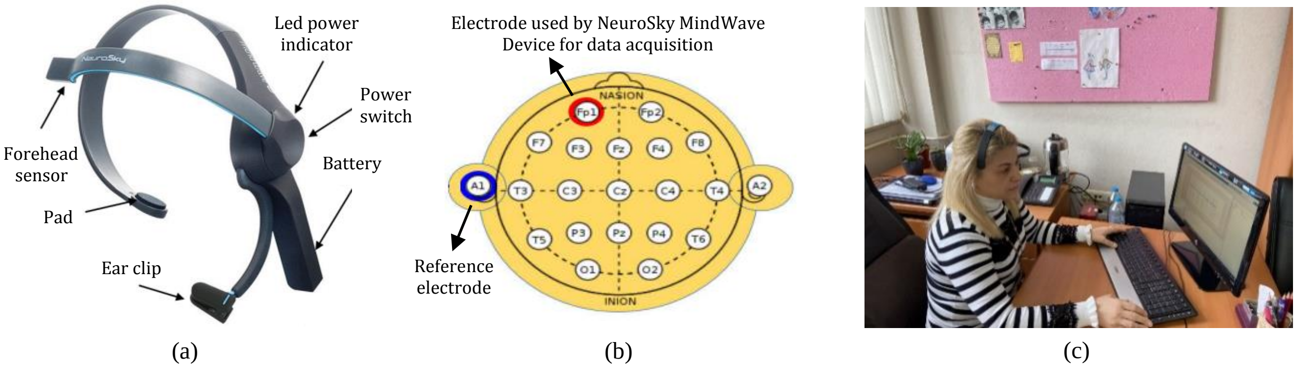

Figure 3. (a) NSMWM device (b) Electrode positions using for data acquisition (c) Recording session of a volunteer.

Table 1. Morse codes of word "WATER" and their eye-blink sequences as seen in Figure 4.

Letter	Code	Eye-blink sequence
A	· —	Single eye-blink + Delay duration + Double eye-blinks
E	·	Single eye-blink
R	· — ·	Single eye-blink + Delay duration + Double eye-blinks + Delay duration + Single eye-blink
T	—	Double eye-blinks
W	· — —	Single eye-blink + Delay duration + Double eye-blinks + Delay duration + Double eye-blinks

2.3 Eye-blink analysis

During the experiments, eye-blink signals were obtained from a total of 20 healthy volunteers aged 20-55 years. A total of 400 recordings were obtained from volunteers for 20 words, 200 of which were used in the training phase for letters and 200 were used in the testing phase for the system.

The signals generated by eye movements have higher amplitudes compared to signals produced by the brain. When the eyelids are closed, the eyeball moves upwards, and this movement is recorded as a positive amplitude signal. When the eyelids are opened, the opposite happens, and this movement is recorded as a negative amplitude signal [21]. Eye-blinks are often considered as artifacts that need to be removed from EEG signals due to their disruptive effect on the signal. While designing Brain-Computer Interfaces, it is not generally considered to take advantage of this artifact, which is generally considered as a muscle movement, and efforts are made to eliminate it.

The first step in eye-blink analysis is to determine the letter count in a word. Therefore, it is necessary to correctly identify the spaces between the letters in the entered word. Figure 4 shows the word "WATER" formed by eye-blinks. In Figure 4, the parameters T_1, T_2, T_3, and T_4 are defined as follows:

T_1: Duration of a single eye-blink

T_2: Duration of double eye-blinks

T_3: Duration between single and double eye-blinks (Delay duration)

T_4: Space duration between two letters

$$T_3 < T_4 \quad (1)$$

The fact that the duration of T_4 is greater than the duration of T_3 indicates the number of letters in the word. If the number of T_4 durations in a word is expressed as T_{4N}, the number of letters in the word L_N can be found using equation (2).

$$L_N = T_{4N} + 1 \quad (2)$$

There are no restrictions on the durations of T_1 and T_2. The T_1 duration is always shorter than the T_2 duration in a natural eye-blink movement. During word input, volunteers were asked to leave only a short space at the end of each letter. All volunteers freely determined the interval between a single eye-blink and a double eye-blink. However, the duration of this interval must be shorter than the duration at the end of each letter.

After finding the T_{4N} value, the second step in the eye-blink analysis is to determine the number of eye-blinks in a letter.

As seen in Figure 4, there are positive and negative peaks in the signal. During an eye-blink, a positive peak is followed by a negative peak. This is an electrical signal that occurs during the closing and opening of an eyelid. The intensity of the positive peaks that occur during the closing of the eye while blinking can vary from person to person and may even differ within the same person. However, the process of opening the eyelid produces an electrical signal of almost the same intensity.

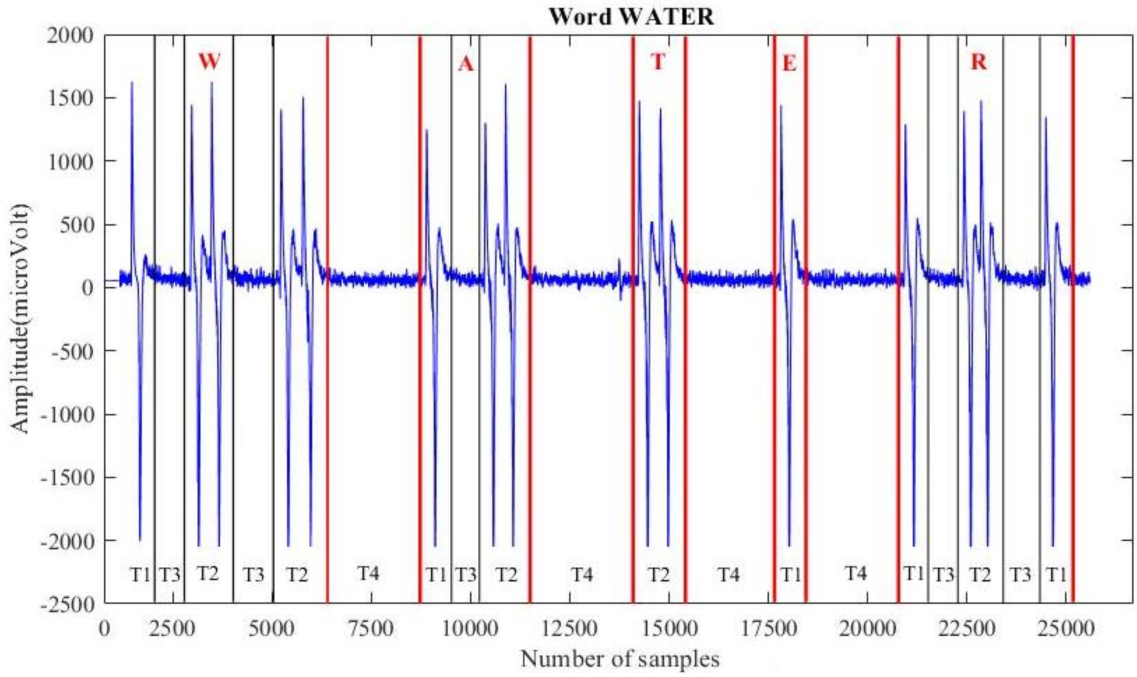

Figure 4. Graphic of the word WATER obtained by eye-blinks as seen in Table 1.

If the positive peak is expressed as $Y_{pos}(k)$ and the negative peak as $Y_{neg}(k)$, equation (3) will be applicable to all eye-blink signals. It was observed in all experiments that $|Y_{pos}|$ and $|Y_{neg}|$ have values between 1300-1600 µV and 2000-2100 µV, respectively. Additionally, the positive peak amplitude values are lower than the negative peak amplitude values in all results. Although the presence of positive peaks provides accurate results in detecting eye-blinks, it is a more accurate approach to use negative peaks for eye-blink detection due to any potential noise effects. For this reason, the detection of the presence of negative peaks gives definite results in terms of determining whether an eye-blink has occurred or not.

$$\left|Y_{pos}(k)\right| < \left|Y_{neg}(k)\right| \tag{3}$$

The peak values of the Y(i) signal can be found as given in the equation (4).

$$\begin{gathered} |Y(i)| > |Y(i+1)| \\ |Y(i)| > |Y(i-1)| \\ |Y(i-1)| = |Y(i+1)| \end{gathered} \tag{4}$$

To calculate negative peaks above a certain T threshold voltage level, equation (5) is obtained from (4).

$$\begin{gathered} Y_{neg}(i) = 1, \quad \begin{matrix} |Y(i)| > |Y(i+1)| \\ |Y(i)| > |Y(i-1)| \\ |Y(i-1)| = |Y(i+1)| \end{matrix} \quad and\ |Y(i)| > T \\ Y_{neg}(i) = 0, \quad \begin{matrix} |Y(i)| < |Y(i+1)| \\ |Y(i)| < |Y(i-1)| \\ |Y(i-1)| \neq |Y(i+1)| \end{matrix} \quad or\ |Y(i)| < T \end{gathered} \tag{5}$$

Y_{neg} has the same length as the Y signal. Here, $Y_{neg}(i)=1$ indicates the presence of a negative peak, and $Y_{neg}(i)=0$ indicates the absence of a negative peak. The number of $Y_{neg}>0$ values included in the resulting Y_{neg} signal will show how many negative peaks are present in a letter. Therefore, it will give eye-blink count between two T_4 periods. On the other hand, the i values that provide the $Y_{neg}(i)>0$, also gives the location of the eye-blinks that occur in the time axis [7].

2.4 Feature vector extraction and classification

2.4.1 Pre-classification

The 26 letters in English can be divided into 7 subclasses according to the number of eye-blinks they contain. When examining Morse code, it is observed that the letters E, T-I, A-N-S, D-H-M-R-U, B-F-G-K-L-V-W, C-O-P-X-Z, J-Q-Y have the same number of eye-blinks (1, 2, 3, 4, 5, 6, 7, respectively). For example, if 4 eye-blinks are detected between two T_4 durations, the letter in this interval may be one of the letters D-H-M-R-U.

2.4.2 Wavelet transform

Wavelet transform is used to analyze the features of signals at different scales [22], [23]. This process is different from Fourier transform, which breaks down signals based on their frequencies. Wavelet transform uses a type of analysis-filtering process to break down the signal into different frequency bands and then analyzes the different features of the signal in each band. Wavelet coefficients are numerical values obtained as a result of the wavelet transform. These values are obtained by multiplying a signal with a wavelet function at different scales and times. The calculation of wavelet coefficients involves several stages of the wavelet transform:

1. Decomposition: First, the signal is multiplied with a certain wavelet function (e.g., Daubechies, Symlets, Coiflets, etc.) to obtain the different parts of the signal at different scales. This process is called decomposition, which separates the signal into low-frequency and high-frequency components.
2. Down-sampling: After the decomposition process, the low-frequency components are subjected to a down-sampling process, which reduces the sampling rate of the signal and shortens the processing time.
3. Continued decomposition: The process continues with the further decomposition of the low-frequency components. This process is repeated to analyze the low-frequency components of the signal at different scales.
4. Calculation of coefficients: After all the decomposition processes are completed, the wavelet coefficients obtained from the wavelet transform of the signal are calculated. These coefficients represent the features of the signal at different scales and time intervals.

When analyzing signals using the wavelet transform, it is important to choose a suitable wavelet and determine the number of decomposition levels based on the dominant frequency components. The classification accuracy depends on the wavelet type and degree used in practice [24]. The choice of wavelet and decomposition level is usually based on the researchers' experience and the signal's shape. The more similar the main wavelet is to the studied signal, the higher the wavelet coefficients obtained, which means it better represents the signal. The number of levels for applying the discrete wavelet transform depends on the signal's sampling frequency and the total number of samples. If the decomposition level number is chosen excessively, the contents at the last levels may no longer represent the signal accurately.

In this study, the number of decomposition levels was determined as 5. Thus, EEG signals are divided into detail sub-bands D1–D5 and approximation sub-band A5. Since the wavelet coefficients D3, D4, D5, and A5 contain important information about the signal, they are considered as feature vectors. Then, the feature vector dimensions were reduced, and some statistical operations were performed on the wavelet coefficients. These operations are: Maximum value of coefficients for each sub-band, minimum value of coefficients for each sub-band, standard deviation of the coefficients for each sub band and average value of coefficients for each sub-band. Thus, a feature vector consisting of 16 features was obtained for each record.

2.4.3 Support Vector Machines

Support Vector Machines (SVM) is a classification algorithm that determines the most suitable hyperplane (or surface) to separate data points [25]. The hyper-plane is determined by maximizing the distance between the closest points of the classes. The points located on the boundaries are called support vectors and they determine the boundary of the class they belong to. SVM can work with both linear and non-linear datasets [26]. A linear SVM attempts to separate data points in a plane, while a non-linear SVM attempts to separate data points in a high-dimensional space. To work with SVM, the dataset must first be divided into two classes. Then, the hyperplane that maximizes the distance between the closest points of the classes is determined. This hyperplane provides the best separation between the classes. The support vectors located on the boundaries determine the hyperplane and are optimized to provide the best separation between the classes. Finally, to classify a new data point, SVM checks on which side of the hyper-plane it is located. If the point is on one side of the hyperplane, it is classified as one class, and if it is on the other side, it is classified as the other class.

SVM can be expressed mathematically as follows:

Let the dataset be denoted as $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$ where y_i in {-1, 1} is the class label of the i-th data point. SVM uses a hyperplane to separate the data points into two classes. This hyperplane is determined to provide the best separation between the data points.

For a linear SVM, the hyperplane can be expressed as:

$$w \cdot x - b = 0 \tag{6}$$

Here, w is a vector parallel to the hyperplane and b is a constant. SVM tries to find the values of w and b that optimize the separation between the classes. SVM works in two stages: training and testing. In the training stage, SVM finds the best hyperplane using the data. This is the hyperplane that provides the best separation between the classes. In the testing stage, when a new data point is given, SVM checks on which side of the hyperplane it is located to classify it.

SVM solves an optimization problem to determine the hyperplane [27], [28]. The goal of this problem is to find the hyperplane that provides the best separation between the classes. The solution to this problem is based on finding the closest points between the classes, which are called support vectors and determine the hyperplane.

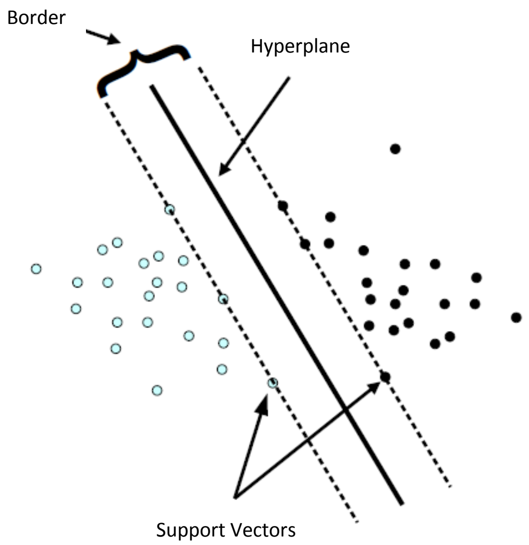

Figure 5. Linear separable case classification in SVM [29].

The designed system includes a total of 6 SVMs, which are determined by the pre-classification process. For instance, if there are three eye-blinks between two T_4 periods, the system will use the SVM-2 shown in Figure 2, and classify the signal based on the three letters A-N-S only. Table 2 shows the number of eye-blinks for each letter: T-I includes 2 eye-blinks, A-N-S includes 3 eye-blinks, and D-H-M-R-U includes 4 eye-blinks, B-F-G-K-L-V-W includes 5 eye-blinks, C-O-P-X-Z includes 6 eye-blinks, and J-Q-Y includes 7 eye-blinks. The feature vector consists of 16 parameters, which are given as input to the appropriate support vector machine and separated according to the number of eye-blinks. However, since the letter E contains only a single eye-blink, it is not included in any class, and thus, a support vector machine is not defined for it.

Table 1. Number of eye-blinks in letters.

Letters	Eye-blinks
E	1
T-I	2
A-N-S	3
D-H-M-R-U	4
B-F-G-K-L-V-W	5
C-O-P-X-Z	6
J-Q-Y	7

2.5 Levenshtein distance algorithm

Levenshtein distance is an algorithm used to measure the similarity between two different strings. It works by comparing the two words letter by letter and calculating the number of changes required to transform one word into the other. These changes include deletions, insertions, and substitutions, and each transaction has a cost of 1 [30]. Mathematically, Levenshtein distance defines the distance between two words a and b as follows:

$$lev_{a,b}(i,j) = \begin{cases} max(i,j) & if \quad min(i,j) = 0, \\ min \begin{cases} lev_{a,b}(i-1,j)+1 \\ lev_{a,b}(i,j-1)+1 \\ lev_{a,b}(i-1,j-1)+1_{(a_i \neq b_j)} \end{cases} & otherwise \end{cases} \tag{7}$$

Here 1 (ai ≠ bj) is the indicator function. It is equal to 0 when ai = bj and 1 otherwise. $lev_{a,b}$ (i, j) is the distance between the first letter i of a and the first letter j of b.

Table 3 shows a two-dimensional matrix used to calculate the change value for different letters in the words. The edit distance between the substring of A of length i and the substring of B of length j is represented by D [i, j]. The first column and row, D [i, 0] and D [0, j], represent the distance of the subarrays to the empty array. The smallest edit distance (D [i, j]) between two characters has two possibilities, depending on the last characters in the two strings:

- If the last characters compared are equal, the edit distance is the value in the left diagonal. So, D [i, j] = D [i-1, j-1].
- If the last characters compared are different, the edit distance is found by adding 1 to the smallest of the cell values to the left, top and left diagonal of the cell.

  So, D [i, j] = 1 + min (D [i, j -1], D [i-1, j], D [i-1, j-1]).

Table 3. Distance matrix.

			Word B			
			B0	B1	B2	B3
		0	1	2	3	4
Word A	A0	1				
	A1	2				
	A2	3				
	A3	4	D[i-1, j-1]	D[i-1, j]		
	A4	5	D[i, j-1]	D[i, j]		
	A5	6				
	A6	7				Total Dist.

The value held in the last cell in the lower right corner of the matrix represents the edit distance required to transform the source word into the target word. The result of the Levenshtein distance algorithm is a single integer value that indicates the number of transformations required to convert one of the two words into the other. The lower the number of changes required, the higher the similarity between the two words. Conversely, increasing the Levenshtein distance indicates increasing differences between the two words. Moreover, a Levenshtein distance of zero implies that the two compared words are the same [31].

3 Results

In this study, to reduce the effect of artifacts, volunteers were asked not to move their heads left and right while recording, and not to move their eyes except during blinking moments. To detect eye-blinks from the signal received by the wireless NSMWM device, first, the spaces in the eye-blink sequence of the word were detected and the number of letters in the word was determined. Then, the eye-blink analysis was completed by determining how many eye-blink signals each letter contained.

Subsequently, pre-classification was performed based on the number of eye-blinks. After these processes, the letter signal's feature vector was extracted using Wavelet transform. Then, these vectors were given as input to a suitable SVM according to pre-classification results. For example, if the number of eye-blinks within a specified time period in a test signal is 3, the pre-classification result concludes that this eye-blink sequence belongs to class A-N-S shown in Table 2. For this reason, only the SVM provided in Figure 5 is used during the testing process. This process was repeated for each letter in a word. Finally, the word was obtained by completing the letters.

However, the large size of the data made it difficult to process the recorded data effectively and quickly in terms of time and processing load. To address this issue, the dimensions of the feature vectors were reduced by performing statistical operations such as mean, standard deviation, maximum and minimum value on the wavelet coefficients. In this study, the 4th order Daubechies (db4) wavelet was chosen as the main wavelet for obtaining feature vectors in the designed system, and discrete wavelet transform was applied to the signal using level 5 coefficients.

As seen in Table 4, the classification accuracy rate is highest when using the maximum, minimum, standard deviation, and average value for the D3, D4, D5, and A5 sub-bands. Therefore, these statistical features are used to create feature vectors for these sub-bands. In addition, Table 5 shows the feature vector calculated for three different letters A. The accuracy percentages given in Table 6 were obtained for each SVM in tests conducted using records that were not used in training. Systems implemented for 10 and 20 individuals were tested, and their performances were calculated with and without using the Levenshtein distance algorithm and given in Table 7.

Table 2. Classification accuracy variation with different statistical values.

Sub-bands	Statistical Values	Number of Attributes	Number of Classes	Classification Accuracy for SVM (%)
D3-D4-D5-A5	Mean, Std, Max, Min, Mod, Median, Kurtosis, Skewness	32	3	88.8889
D4-D5-A5	Mean, Std, Max, Min, Mod, Median, Kurtosis, Skewness	24	3	88.8889
D5-A5	Mean, Std, Max, Min, Mod, Median, Kurtosis, Skewness	16	3	88.8889
D3-D4-D5-A5	Mean, Std, Max, Min	16	3	91.1111

Table 5. Feature vector calculated for 3 letters A.

Feature Vector	A1	A2	A3	Feature Vector	A1	A2	A3
Max(D3)	0.78	0.71	0.75	Max(D5)	219.11	190.29	205.91
Min(D3)	-0.60	-0.60	-0.59	Min(D5)	-163.36	-146.25	-180.33
Std(D3)	0.17	0.16	0.17	Std(D5)	52.87	48.05	52.36
Mean(D3)	0.00	0.00	0.00	Mean(D5)	0.01	-0.02	-0.02
Max(D4)	15.81	14.11	14.45	Max(A5)	3878.93	3837.27	3921.41
Min(D4)	-12.10	-11.43	-11.22	Min(A5)	-5404.40	-5261.02	-5369.67
Std(D4)	3.46	3.14	3.43	Std(A5)	1490.92	1517.69	1459.94
Mean(D4)	-0.00	0.00	0.00	Mean(A5)	320.14	291.77	336.82

Table 6. Classification results for all letters using SVM1-SVM6 trained with 20 volunteers.

Classification Method		Class	Recall	Precision	Accuracy (%)	Time (s)
SVM1	Polynomial	T-I	1.000	1.000	100	0.02
	Linear		1.000	1.000	100	0.01
SVM2	Polynomial	A-N-S	0.903	0.906	90.3226	0.26
	Linear		0.903	0.903	90.3226	0.31
SVM3	Polynomial	J-Q-Y	0.867	0.862	86.6667	1.51
	Linear		0.878	0.878	87.7778	1.42
SVM4	Polynomial	D-H-M-R-U	0.824	0.828	82.4314	0.63
	Linear		0.835	0.837	83.4706	0.77
SVM5	Polynomial	C-O-P-X-Z	0.814	0.817	81.4	2.39
	Linear		0.793	0.796	79.3333	2.14
SVM6	Polynomial	B-F-G-K-L-V-W	0.798	0.794	79.7887	3.22
	Linear		0.798	0.790	79.7619	3.46

Table 7. Classification accuracy based on words in the dataset with different number of volunteers.

Word	SVM Accuracy 1 Volunteer (%)	SVM Accuracy 10 Volunteers (%)	SVM Accuracy 20 Volunteers (%)	SVM + Levenshtein Accuracy 1 Volunteer (%)	SVM + Levenshtein Accuracy 10 Volunteers (%)	SVM + Levenshtein Accuracy 20 Volunteers (%)
CLOSE	100	96.02	94.11	100	100	100
COFFEE	100	97.56	95.91	100	100	100
COME	100	98.12	94.99	100	100	100
END	98.88	91.21	89.27	100	94.78	92.84
ENOUGH	93.95	91.46	89.86	100	100	100
FEW	99.19	96.44	92.85	100	100	100
FRUIT	93.28	89.69	87.75	100	100	100
FULL	90.55	86.61	84.21	100	100	97.74
LATER	94.52	93.21	91.99	100	96.85	95.63
LEFT	100	99.23	97.29	100	100	100
NO	100	96.75	91.99	100	100	100
ONLY	88.77	86.97	84.44	99.08	97.34	94.81
OPEN	100	98.09	95.45	100	100	100
OUT	95.31	91.44	87.87	100	100	100
READ	95.99	91.41	89.46	99.50	94.92	92.97
SICK	84.22	79.32	76.75	100	99.24	96.67
TEA	100	98.23	96.55	100	100	100
TURN	88.71	84.88	80.47	99.86	96.03	91.62
WATER	93.47	91.12	89.99	99.03	96.68	95.55
YES	100	90.12	86.66	100	100	100

In the last section of the designed system, the sounds of 20 words in the database were recorded. During the test phase, the result obtained at the system output is voiced using the sound records of the corresponding word in the sound database.

4 Discussion

The designed system has a complex structure, considering the wavelet analysis, eye-blink detection, 6 SVM classifiers, and the Levenshtein distance algorithm used in it. Initially, a single SVM that classified single blink, double blink, and space signals did not yield successful results due to the recorded data shifting too much in the time axis. Eye blinks can occur at very different time instants from person to person, even for the same person, which directly affects classification success. Therefore, it was preferred to determine the number of eye-blinks and classify the letters according to the Morse Alphabet.

When previous literature studies are examined in Table 8, it is observed that the Levenshtein distance algorithm was not used in the implemented systems. Although the accuracy rate of our system appears to be lower compared to other systems, it is observed that there is a significant increase in the accuracy percentage after the inclusion of the Levenshtein distance algorithm to the system.

Although a single SVM structure with a feature vector of 16 statistical values as input and 26 letters in English as output seems appropriate, the large number of letters to be classified would decrease the classification accuracy of the SVM. For this reason, 6 SVMs were used for letter classification. After the pre-classification process according to the number of eye-blinks, the appropriate SVM was used. Thus, the designed system provided more satisfactory results and became more adaptable for daily use.

As can be seen in Figure 2, Levenshtein distance algorithm compares the word given by the system as a result with the words in the database, and if there is an incorrect word, it selects the word closest to the incorrect word as output. If the Levenshtein distance algorithm is removed from the system, it will no longer be connected to a specific database and will allow for an unlimited number of words to be used. However, in this case, any error that may occur in any letter of the words will decrease the overall accuracy of the system. When the system is designed as a person-dependent system, it has been observed to have a very high accuracy in experiments. For the 20 words in the database, an average accuracy rate of 99.87% was obtained with the Levenshtein distance algorithm, and 95.84% without using the Levenshtein distance algorithm. When the system is considered as a person-independent system, the accuracy rate decreases. In a study performed with 10 people, an accuracy of 98.79% was obtained with the Levenshtein distance algorithm, and 92.39% without using the Levenshtein distance algorithm. In trials with 20 volunteers, an accuracy of 97.89% was found with the Levenshtein distance algorithm, and 89.90% without using the Levenshtein distance algorithm. When Levenshtein distance algorithm is used, the accuracy of the system increases up to 8%. However, the increase in the number of words in the database negatively affects the performance of the Levenshtein distance algorithm.

It is debatable whether an unlimited database is necessary. Long blinking periods during a conversation may tire the patient, so it may be more reasonable to consider a limited database only for the patient's basic needs. Despite this disadvantage, the system is quite satisfactory in terms of cost and accuracy for the basic needs of an ALS patient.

The use of a single-channel EEG in the implemented system has significantly reduced the cost compared to other studies. Eye trackers and EOG devices are costly, and using a single-channel EEG device has reduced the cost considerably. This has also reduced the patient's discomfort during data input. The data input rate is one of the most important factors affecting the system speed. Considering the health status of ALS patients, it is clear that the data input rate will be below the desired value. The data input rate of the eye-blinks is a parameter that is entirely related to the physical condition of the patient. Experimental results with healthy individuals will not give a clear result in terms of data input speed. The data input rate of the system depends entirely on the duration of the patient's blinking without getting tired. When the data from healthy volunteers was examined, it was observed that the number of eye-blinks in the Morse codes forming the letter and the space duration between the letters had a significant effect on the total data input duration. For example, the word "NO" has an input time of about 19.5 seconds, while this duration is 17 seconds for the word "TEA" and about 30 seconds for the word "YES". These values were obtained from healthy volunteers, and it can be said that these values will increase for ALS patients.

5 Conclusions

This study can be considered as a multidisciplinary study that covers many subtopics and algorithms. The aim of the study was to investigate the effects of the Levenshtein distance algorithm on a system that blends these topics and algorithms in a harmonious manner, enabling it to communicate with the environment of people with neurological diseases such as amyotrophic lateral sclerosis, stroke, and lock-in syndrome at low cost and with high accuracy. In our study demonstrated that the Levenshtein distance algorithm significantly improved the accuracy of the eye-blink communication system we developed for individuals with severe motor disabilities. By analyzing eye-blink signals using the NeuroSky MindWave Mobile device and Wavelet Transform method, we were able to extract feature vectors that could be used for classification. The support vector machines trained on these vectors and the Levenshtein distance algorithm helped to reduce classification errors and improve system performance. Overall, our experiments showed that the developed system could successfully recognize and vocalize 20 words for self-expression, which has the potential to greatly enhance the communication abilities of individuals with severe motor disabilities.

Future studies could focus on several areas to further improve the eye-blink communication system. For instance, the system could be tested with a larger sample size of individuals with different types and severity levels of motor disabilities to evaluate its effectiveness across a wider range of users. Additionally, more words and phrases could be added to the system to expand its vocabulary and increase its practical use.

Another potential area of future research is exploring the use of alternative input methods, such as facial expressions or head movements, to increase the flexibility and ease of use of the communication system. Moreover, different machine learning algorithms could be evaluated to compare their performance with the Levenshtein distance algorithm used in this study.

This system, which is designed for people with neurological diseases such as amyotrophic lateral sclerosis paralysis, and lock-in syndrome, could also potentially be used in fields such as the military, especially for target orientation in quiet environments, by using a limited database.

6 Author contribution statements

In the study, Author 1 and Author 2 contributed to forming the idea, making the design, reviewing the literature, evaluating the results, supplying the materials used and examining the results. Author 3 contributed to spelling and checking the article in terms of content.

Table 8. Comparison of the studies in the literature.

Device	Cost ($)	Speed	Accuracy (%)	Accuracy with Levensthein (%)
EEG device [32]	High-priced	2.3 character per minute	95	Not tested
EEG device [33]	High-priced	50.5 bits per minute	Over 90	Not tested
Neuroscan Synamps [34]	2,500	47.26 bits per minute	79	Not tested
64-channel EEG amplifier [35]	11,000	11.11 character per minute	93	Not tested
32-channel Brain Amp MR [36]	High-priced	1.67 minute per word	Not specified	Not tested
Cleve Med Bio Radio device [37]	400-500	60-78 seconds for 1 character	84.76	Not tested
Eye tracking system [38]	High-priced	8.58 words per minute	96.6	Not tested
Eye tracking device [39]	High-priced	9.89 words per minute	98.8	Not tested
2-channel EOG [40]	250-300	“WATER” was typed in 24.5 sec.	95	Not tested
EyeTech eye tracker [41]	High-priced	4.8 words per minute	97	Not tested
Cellular phones [42]	Low-priced	20 character per minute	99.8	Not tested
Tobii eye tracker [43]	20,000	6.8 words per minute	Not specified	Not tested
Data acquisition device [44]	750	Not specified	50-100	Not tested
EOG device [45]	Low-priced	Not specified	87.38	Not tested
EOG Biosignal Amplifier [46]	1875	27.9 character per minute	95	Not tested
EOG device [47]	250-300	17.5 words per minute	92.5	Not tested
Emotiv Epoc [48]	850	Not specified	95	Not tested
IR sensor [49]	20-25	1 character for 3 seconds	Not specified	Not tested
NSM Mobile device [7]	110	“TEA” was typed in 17 sec.	99.2	Not tested
NSM Mobile device [8]	110	“EVET” was typed in 32.8 sec.	94-99.5	Not tested
NSM Mobile device [50]	110	A predefined sentence was typed in 12-14 sec.	98.75	Not tested
NSM Mobile device	110	“TEA” was typed in 17 sec.	89.89	97.89

7 Ethics committee approval and conflict of interest statement

The dataset comes from volunteers. So, there is no need to obtain permission from the ethics committee for the article prepared.

There is no conflict of interest with any person / institution in the article prepared.

8 References

[1] Arthur KC, Calvo A, Price TR, Geiger JT, Chio A, Traynor B. J. "Projected increase in amyotrophic lateral sclerosis from 2015 to 2040". *Nature Communications*, 7(1), 1-6, 2016.

[2] Soman S, Murthy BK. "Using brain computer interface for synthesized speech communication for the physically disabled". *Procedia Computer Science*, 46, 292-298, 2015.

[3] García L, Ron-Angevin R, Loubière B, Renault L, Le Masson G, Lespinet-Najib V, André JM. "A comparison of a brain-computer interface and an eye tracker: Is there a more appropriate technology for controlling a virtual keyboard in an ALS patient?". *In Advances in Computational Intelligence: 14th International Work-Conference on Artificial Neural Networks, IWANN*, Cadiz, Spain, 14-16 June 2017.

[4] de Lima Medeiros PA, da Silva GVS, dos Santos Fernandes FR, Sánchez-Gendriz I, Lins HWC, da Silva Barros DM, de Medeiros Valentim RA. "Efficient machine learning approach for volunteer eye-blink detection in real-time using webcam". *Expert Systems with Applications*, 188, 2022.

[5] Królak A, Strumiłło P. "Eye-blink detection system for human–computer interaction". *Universal Access in the Information Society*, 11, 409-419, 2012.

[6] Francis WC, Umayal C, Kanimozhi G. "Brain-computer interfacing for wheelchair control by detecting voluntary eye blinks". *Indonesian Journal of Electrical Engineering and Informatics (IJEEI)*, 9(2), 521-537, 2021.

[7] Ekim G, Atasoy A, İkizler N. "A New Approach for Eye-Blink to Speech Conversion by Dynamic Time Warping". *Traitement du Signal*, 38(2), 369-377, 2021.

[8] Ekim G, İkizler N, Atasoy A. "A Study on Eye-Blink Detection-Based Communication System by Using K-Nearest Neighbors Classifier". *AECE Advances in Electrical and Computer Engineering*, 23(1), 71-78, 2023.

[9] Kaur A. "Wheelchair control for disabled patients using EMG/EOG based human ma-chine interface: A review". *Journal of Medical Engineering & Technology*, 61-74, 2020.

[10] Tamura H, Yan M, Sakurai K, Tanno K. "EOG-sEMG Human Interface for Communication". *Computational Intelligence and Neuroscience, Special Issue*, 2016.

[11] Mukherjee K, Chatterjee D. "Augmentative and Alternative Communication device based on eye-blink detection and conversion to Morse-code to aid paralyzed individuals". *In 2015 International Conference on Communication, Information & Computing Technology (ICCICT)*, 1-5, 2015.

[12] Singh J, Miglani R, Goel C. "Encrypted Morse Communication using Eye Blinks and Light Bulb". *Proceedings of the International Conference on Innovative Computing & Communication (ICICC)*, 2021.

[13] Chareonsuk W, Kanhaun S, Khawkam K, Wongsawang D. "Face and Eyes mouse for ALS Patients". *In 2016 Fifth ICT international student project conference (ICT-ISPC)*, 77-80, 2016.

[14] Rashid M, Sulaiman N, Mustafa M, Bari BS, Sadeque MG, Hasan MJ. "Wink based facial expression classification using machine learning approach". *SN Applied Sciences*, 2, 1-9, 2020.

[15] Groll MD, Hablani S, Vojtech JM, Stepp CE. "Cursor Click Modality in an Accelerometer-Based Computer Access Device". *In IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 28(7), 1566-1572, July 2020.

[16] Wang J, Xu S, Dai Y, Gao S. "An Eye Tracking and Brain-computer Interface Based Human-environment Interactive System for Amyotrophic Lateral Sclerosis Patients". *IEEE Sensors Journal*, 2022.

[17] Cipresso P, Meriggi P, Carelli L, Solca F, Meazzi D, Poletti B, Lulé D, Ludolph A, Riva G, Silani V. "The combined use of Brain Computer Interface and Eye-Tracking technology for cognitive assessment in Amyotrophic Lateral Sclerosis". *5th International Conference on Pervasive Computing Technologies for Healthcare (Pervasive Health) and Workshops*, Ireland, 320-324, 2011.

[18] Ma X, Yao Z, Wang Y, Pei W, Chen H. "Combining brain-computer interface and eye tracking for high-speed text entry in virtual reality". *In 23rd international conference on intelligent user interfaces*, 263-267, March 2018.

[19] Awais MA, Yusoff MZ, Yahya N, Ahmed SZ, Qamar MU. "Brain controlled wheelchair: A smart prototype". *In Journal of Physics: Conference Series, IOP Publishing*, 1529(4), April 2020.

[20] Carron LP. "Morse code: the essential language (No. 69)" Amer Radio Relay League, 1991.

[21] Fisch BJ. *Spehlmann's EEG primer*. Elsevier Science Limited, 1991.

[22] Subaşı A. "EEG signal classification using wavelet feature extraction and a mixture of expert model". *Expert Systems with Applications*, 32(4), 1084-1093, 2007.

[23] Übeyli ED, Güler İ. "Feature extraction for diagnostic systems using wavelet transform: State analysis of internal carotid artery Doppler signals". *ASYU-INISTA*, 2004.

[24] Ekim G, İkizler N, Atasoy A. "The effects of different wavelet degrees on epileptic seizure detection from EEG signals". *In 2017 IEEE International Conference on INnovations in Intelligent SysTems and Applications (INISTA)*, 316-321, 3-5 July 2017.

[25] Chang CC, Lin CJ. "LIBSVM: A Library for Support Vector Machines". *ACM transactions on intelligent systems and technology (TIST)*, 2(3), 1-27, 2011.

[26] Ekim G, Atasoy A, İkizler N. "Statistical comparison of classification methods in EEG signals". *In 2017 25th Signal Processing and Communications Applications Conference (SIU)*, 1-4, IEEE, May 2017.

[27] Günay M, Alkan A, "Classifications of EMG Signals by Spectral Methods and SVM Classifier", Kahramanmaras Sutcu Imam University Journal of Engineering Sciences, Volume 13-2, 63-80, 2010.

[28] Tuncer S, A, Alkan A, "Classifications of EMG Signals taken from arm with hybrid CNN-SVM architecture", Concurrency and Computation Practice and Experience, Volume 34-5, 2022.

[29] Meyer D, Wien FT. "Support vector machines". *The Interface to libsvm in package e1071*, 28(20), 597, 2015.

[30] Levenshtein VI. "Binary codes capable of correcting deletions, insertions, and reversals". *In Soviet physics doklady*, 10(8), 707-710, 1966.

[31] Heeringa WJ. Measuring Dialect Pronunciation Differences Using Levenshtein Distance. PhD Thesis, University of Groningen, Groningen, Holland, 2004.
[32] Farwell LA, Donchin E. "Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials". *Electroencephalography and Clinical Neurophysiology*, 70(6), 510-523, 1998.
[33] Meinicke P, Kaper M, Hoppe F, Heumann M, Ritter H. "Improving transfer rates in brain computer interfacing: A case study". *In Advances in Neural Information Processing Systems*, 1131-1138, 2003.
[34] Kaper M, Ritter H. "Generalizing to new subjects in brain-computer interfacing". *The 26th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 2, IEEE*, 4363-4366, 2004.
[35] Amcalar A, Cetin M. "A Brain-Computer Interface System for Online Spelling", *IEEE 18th Signal Processing and Communication Applications Congress*, Diyarbakir, 196-199, 2010.
[36] Akram F, Seung Moo H, Tae-Seong K. "An efficient word typing P300-BCI system using a modified T9 interface and random forest classifier", *Computers in Biology and Medicine* 56, 30-36, 2015.
[37] Oralhan Z. "The Effect of Interstitial Time and Stimulus Structure on Performance in P300 Based Brain Computer Interface Systems". *Düzce University Journal of Science and Technology*, 7.3: 1834-1846, 2019.
[38] Miniotas D, Spakov O, Evreinov GE. "Symbol Creator: An Alternative Eye-based text entry technique with low demand for screen space". *In INTERAC,* 3, 137-143, 2003.
[39] Majaranta P, Aula A, Raiha KJ. "Effects of feedback on eye typing with a short dwell time", *In Proceedings of the 2004 Symposium on Eye tracking Research & Applications*, 139-146, 2004.
[40] Usakli AB, Gurkan S. "Design of a novel efficient human–computer interface: An electrooculogram based virtual keyboard". *IEEE Transactions on Instrumentation and Measurement*, 59(8), 2099-2108, 2009.
[41] MacKenzie IS, Ashtiani B. "BlinkWrite: efficient text entry using eye blinks", *Universal Access in the Information Society*, 10(1), 69-80, 2011.
[42] Zhang C, Yao R, Cai J." Efficient eye typing with 9-direction gaze estimation", *Multimedia Tools, and Applications*, 77(15), 19679-19696, 2018.
[43] Porta M, Turina M. "Eye-S: a full-screen input modality for pure eye-based communication". *In Proceedings of the 2008 Symposium on Eye Tracking Research & Applications*, 27-34, 2008.
[44] Tsai J, Lee C, Wu C, Wu J, Kao K. "A feasibility study of an eye-writing system based on electro-oculography", *Journal of Medical and Biological Engineering*, 28(1), 39, 2008.
[45] Lee KR, Chang WD, Kim S, Im CH. "Real-time "eye-writing" recognition using electrooculogram". *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 25(1), 37-48, 2016.
[46] Fang F, Shinozaki T. "Electrooculography-based continuous eye-writing recognition system for efficient assistive communication systems". *PloS one*, 13(2), 2018.
[47] Ozbek Ulkutas H. Development of computer-based eye-writing system by using EOG. M.Sc. Thesis, Department of Biomedical Engineering, Başkent University, Ankara, Turkey, 2015.
[48] Soman S, Murthy BK. "Using brain computer interface for synthesized speech communication for the physically disabled". *Procedia Computer Science*, 46, 292-298, 2015.
[49] Mukherjee K, Chatterjee D. "Augmentative and alternative communication device based on eye-blink detection and conversion to Morse-code to aid paralyzed individuals". *In 2015 International Conference on Communication, Information & Computing Technology (ICCICT), IEEE*, 1-5, 2015.
[50] İkizler N, Ekim G, Atasoy A. "A Novel Approach on Converting Eye Blink Signals in EEG to Speech with Cross Correlation Technique". *AECE Advances in Electrical & Computer Engineering*, 23(2), 29-38, 2023.

"