IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received February 28, 2020, accepted March 26, 2020, date of publication March 31, 2020, date of current version April 16, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2984538

Wheelchair Free Hands Navigation Using Robust DWT_AR Features Extraction Method With Muscle Brain Signals

ZAINEB M. ALHAKEEM[1], **(Member, IEEE), RAMZY S. ALI**[1], **(Senior Member, IEEE), AND RAED A. ABD-ALHAMEED**[1,2], **(Senior Member, IEEE)**
[1]Department of Electrical Engineering, University of Basrah, Basrah 00964, Iraq
[2]Department of Biomedical and Electronic Engineering, Faculty of Engineering, University of Bradford, Bradford BD7 1DP, U.K.

Corresponding author: Zaineb M. Alhakeem (zm_engineer@yahoo.com)

ABSTRACT Researchers try to help disabled people by introducing some innovative applications to support and assess their life. The Brain-Computer Interface (BCI) application that covers both hardware and software models, is considered in this work. BCI is implemented based on brain signals to be converted to commands. To increase the number of commands, non-brain source signals are used, such as eye-blinking, teeth clenching, jaw squeezing, and other movements. This paper introduced a low dimensions robust method to detect the eye-blinks and jaw squeezing; so that the method can be applied to drive a wheelchair by using five commands. Our approach is used Discrete Wavelet Transform with Autoregressive to extract the signal's features. These features are classified by using a linear Support Vector Machine (SVM) classifier. The present method detects every testing sample using a small training set to test and drive a powered wheelchair. The proposed method is fully implemented practically based on binary-coded commands.

INDEX TERMS Autoregressive, brain-computer interface, coded commands, discrete wavelet transform, driving a wheelchair, electroencephalography, eye-blinks, jaw squeeze, muscle brain signals, robust features extraction method, signals detection.

I. INTRODUCTION

Brain-Computer Interface (BCI) is a control protocol that converts the brain activity signals to some useful commands to an external device. This system can be classified into two categories: invasive and noninvasive systems, depending on the place of the electrodes that will record the signals from an object.

Invasive BCI system needs the electrodes to be inside the patient body that will add surgical risks, meanwhile, the noninvasive BCI system does not need any risky operations. Non-invasive BCI systems use electrodes distributed over specific areas of the scalp [1]. These electrodes are either dry or wet [2]. The recorded signals are classified depending on the biophysical nature of the signal source [1]: Metabolic measured by techniques like (near-infrared spectroscopy (NIRS), Electrophysiological that measured by electroencephalography (EEG), Magnetic signal that measured by magnetoencephalography (MEG) [3] and [4].

The associate editor coordinating the review of this manuscript and approving it for publication was Mohammad Zia Ur Rahman.

BCI used in many fields of live entertainment, live enhancement (smart home) and the most important use can be found in medical applications. BCI can be used in rehabilitation of stroke patients [5], and to make life easy to disable people who are locked in their bodies (have good mental activity with very low muscle activity) [3] and [6]. The features are the measurable property of a process by which the classifier could separate and recognize classes. There are many types of feature extraction methods such asAutoregressive (AR), Wavelet Transform (WT), Common Spatial Pattern (CSP), Independent Component Analysis (ICA), Statistical Features such as Mean, standard deviation, Maximum, Minimum, etc.; or a combination of features [3], [7]. There are several types of classifiers each one has its advantages and disadvantages. Depending on the features characteristics, the nature of the dataset and the number of classes the classifier should be chosen [8]. To distinguish between classes, classification methods should be applied on the feature vector such as Linear discriminant Analysis (LDA), Support vector machine (SVM), K-nearest neighbor (KNN), Neural networks Multilayer perceptron (MLP) [1], [9] and [10], Deep learning Convolutional Neural Network (CNN) [11] and many other types.

Some applications used direct muscle signals which are known as electromyography (EMG) signals, researchers introduced methods to classify such signals, usually, these applications are multiclass classification applications [12].

In the applications of electroencephalography (EEG), the involuntarily eye-blinks which can be considered as artifacts that should be removed or rejected to clean up the brain signals. The muscle activity in EEG signals is considered as artifacts or noise while they are used as main signals in direct muscle applications. The healthy natural person blinks every five seconds involuntarily. The signals that collected using EEG are different than those collected by Electrooculography (EOG) because of the positions of the channels. In EEG, the frontal channels record the highest amplitude and intensity of the ocular signals than other channels [13]. Tan *et al.* proposed a method to detect the eye-blinks in an image by tracking the iris [14]. Krishnaveni *et al.* used an adaptive thresholding method to remove the ocular artifacts from the EEG signals [15]. Herrero *et al.* presented a method to clean the EEG signals from multiple channels by using Second Order Blind Identification (SOBI) to separate the source of the artifacts [16]. Okada *et al.* proposed a method using independent Component Analysis (ICA) to identify and remove the eye-blinks from Magnetoencephalography (MEG) multi-channels [17]. While Hoffman *et al.* used the same method with the EEG signal [18]. Vialatte *et al.* used the wavelet transform to reject this artifact [19]. Sovierzoski *et al.* developed an eye-blink identifier using time-domain features and MLP [13]. Krolak *et al.* proposed a method to detect the face firstly in an image then detect the eye using the Canonical Correlation Analysis (CCA) method [20]. Kong *et al.* proposed a method to identify and remove the eye-blinks using ICA and Correlation Based Index (CBI) [21]. Matiko *et al.* used Morphological Component Analysis (MCA) in single-channel EEG real-time applications [22]. Rihana *et al.* used Max, Min, and Kurtosis of the EEG signals with MLP classifier [23]. Roy *et al.* recognize the blinks to estimate the fatigue statue using source separation with six features [24]. Tibdewal et.al. detect whether there is eye-blink in the signal segment or not using DWT and ANN [25]. Chang *et al.* proposed an unsupervised method based on the individual threshold level of the eye-blinks to detect the artifact. This method is useful in real-time applications [26]. As eye-blinks sometimes consider as artifacts, other times, the voluntarily eye-blinks are used to create commands. Al-Gawwam *et al.* proposed a technique to detect the eye-blink from a video frame [27]. Other facial movements could also be used, such as (teeth clenching, eyebrow movement, smile, jawing, and other movements). Lawhern *et al.* proposed a method to classify the previous movements using autoregressive and SVM [28]. Wang *et al.* proposed a hybrid BCI system used brain signals with eye-blinks to control the stopping of a wheelchair. They used (CCA) to identify the eye-blinks [29]. Lin *et al.* used DWT to recognize a facial action among six others. This method recognize teeth clenching 100% [30]. Heo *et al.* developed a novel wearable which is used in driving a wheelchair [31]. Gao *et al.* used teeth clenching as a command in a hybrid system to write a word using a robot hand. They used DWT to extract alpha and beta bands of the EEG signal [32]. Khosham *et al.* presented a method to detect jaw movements. Power spectrum density (PSD) is calculated for the signal, and the frequency threshold is found in the range 20Hz and above. For each band, a clenching index is calculated to detect the jaw clenching [33]. Palaniappan et.al. they proposed a coding method inspired from Morse code alphabets, the dashes and dots by using two different mental tasks, the math task and figure rotation task [34]. Jiang *et al.* proposed Morse based coding to control robot arm, this coding used two MI signals left and right hands imagination [35].

II. METHODS AND MATERIALS

The following Section contains the recording paradigm and the datasets.

A. DATASETS AND RECORDING PARADIGM

Two different datasets are recorded using 16 dry channels of OPENBCI headset from three different subjects. The channels labels are listed in Table 1. Each subject sat on an armed-chair comfortably in front of a laptop screen. A phrase appeared on the screen for 5 seconds told the participant to blink his/her eyes or squeeze his/her jaw five times for each movement.

TABLE 1. Channels labels in OPENBCI.

Number	Channel label	Number	Channel label
1	FP1	9	F7
2	FP2	10	F8
3	C3	11	F3
4	C4	12	F4
5	P7	13	T7
6	P8	14	T8
7	O1	15	P3
8	O2	16	P4

1. Dataset1: Three different subjects (suba, subh, and subz) recorded this dataset in fifteen sessions each one contained 90 samples (43 samples for jaw squeezing and 47 samples for eye-blinks).

2. Dataset2: Single-subject (subz) recorded five sessions each session of 100 samples for eye-blinks and jaw squeezes 50 for each movement.

Figure 1 shows the OPENBCI_GUI window that indicates eye-blinks and jaw squeeze signals. Figure (2-a and b) shows the intensity of the EEG signal in the head plot for both

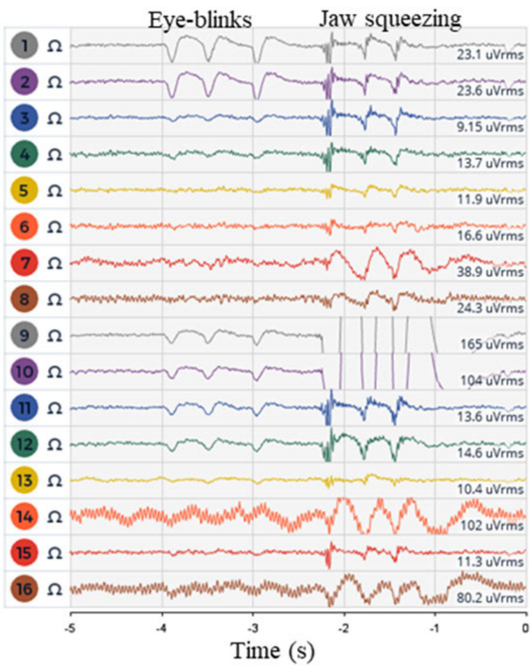
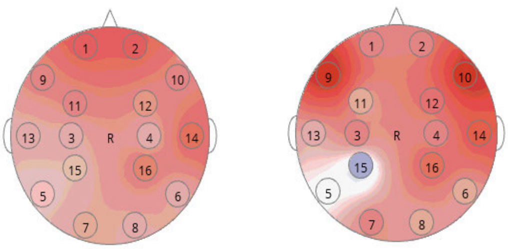


FIGURE 1. **Eye-blink and jaw squeezing signals recorded by 16 channels using OPENBCI headset.**



a: eye-blinking b: jaw squeezing

FIGURE 2. **The intensity channels during a) eye-blinking and b) jaw squeezing.**

eye-blinks and jaw squeeze, respectively. From these figures, it is evident that Ch1 and Ch2 read very high-intensity for eye-blinks signals, while Ch9 and Ch10 are the best channels for jaw squeezing.

B. THE USED METHODS AND THE PROPOSED METHOD

The following Section contains the features extraction and classification.

1) DISCRETE WAVELET TRANSFORM (DWT)

DWT is a fast linear algorithm for machine computation [19], [23]. The applications of DWT in biomedical signals are compression, de-noising, and feature extraction in the time-frequency domain. The decomposition process is a convolution of the input signal $S(n)$ with two analysis filters, low pass filter labeled as H and high pass filter labeled as G, as seen in Figures 3. The low-frequency component is decomposed into high and low frequencies as illustrated in

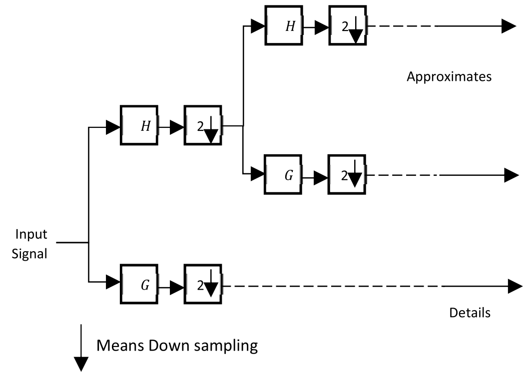


FIGURE 3. **DWT decomposition.**

following Eqns.1 and 2:

$$a(k) = \sum_{n} S(n)\, H\,(2\text{k-n}) \tag{1}$$

$$d(k) = \sum_{n} S(n)\, G(2k-n) \tag{2}$$

2) AUTOREGRESSIVE COMPONENTS (AR)

AR is a spectral estimation method for signal modeling [15], [19]. AR aims to obtain the filter coefficients. Mathematically AR model of P order describes the signal $s(t)$ as follows:

$$s(t) = a_1 s(t-1) + a_2 s(t-2) + \ldots + a_p s\,(\text{t-p}) \tag{3}$$

where a_i is the ith order AR component.

3) THE PROPOSED METHOD DWT_AR

For this present work, we proposed a robust low dimension method to detect the eye-blinks and jaw squeeze signals using EEG data. DWT is used to extract the frequency ranges of both movements using single-level decomposition. The approximate signal represents the eye-blinks in the time-frequency domain, and the details signal represents the jaw squeezing. To reduce the size of the produced signals, AR with low order is used (p=1 for single user and p=3 for multiuser detection). The pseudo-code of the training phase for the proposed method is as follows:

Chan=The selected channels
Time=duration of the trail
Order=BPF order
P=AR order
Type=BPF type
Num_trails=1
While (Num_trails<=Max_num_trails)
Data=Training_data(chan,time,num_trails);
Filtered_data=BPF(Data,Type,Order);
[approximat,details]=DWT(Fitered_data,db1)
Features1=AR(approximate,P)
Features2=AR(details,P)
Features=[Features1,Features2]
*Classifier_model=SVM(Features, Training_*

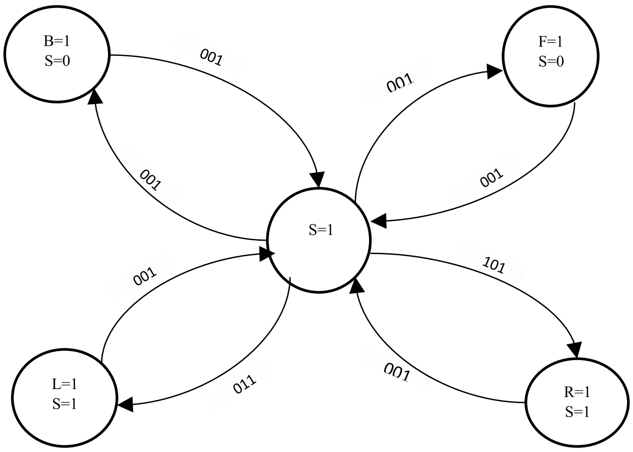


FIGURE 4. Finite state machine of the system.

Labels, linear_kernal)
End While
SVM=save(Classifier_model)

The pseudo-code of the Testing Phase for DWT_AR as the following:

Do
Data=Read 2 sec of data
Filtered_data=BPF(Data,Type,Order)
[approximat,details]=DWT(Fitered_data,db1)
Features1=AR(approximate,P)
Features2=AR(details,P)
Features=[Features1,Features2]
Desition=SVM(Features)
Until stop

4) SUPPORT VECTOR MACHINE CLASSIFIER (SVM)

The SVM classifier is a simple, fast one that is used to maximize the margins between two hyperplanes [7], [9]. These hyperplanes used to separate binary classes. This classifier could be linear or non-linear, depending on its kernel. This classifier is insensitive to over-fitting. As thi classifier is fast, it is suitable to be used in practical implementations.

5) BCI SYSTEM BASED ON DWT_AR METHOD

This method is implemented in the BCI system to drive a powered wheelchair. Subz data are used to train the SVM classifier model that will be used in online testing. As there is only two status, we have proposed a binary code to refer to each command that is used in the BCI system. Table 2 shows the codes and the commands of this system. The zeros represent eye-blinks, and the ones represent jaw squeeze. One of the commands that are shown in this table is ''Keep Moving'' which is meant to stay forward, backward, or stay in stopping statues. After turning to left or right, the stop command automatically activated to prevent the user from turning in loops. Table 3 shows the commands depending on the previous states. Figure 4 shows the finite state machine of the system.

TABLE 2. The coded commands for the system.

Code	Command
000	Keep moving
001	Forward/Backward/Stop
010	Keep moving
011	Turn left
100	Keep moving
101	Turn right
110	Keep moving
111	Keep moving

TABLE 3. Previous statuses conditions and the commands.

$Status_0$	$Status_1$	Code	Command
Backward/Left/Right	Stop	001	Forward
Forward	Stop	001	Backward
Forward	Stop	011	Turn Left
Forward	Stop	101	Turn Right

III. RESULTS

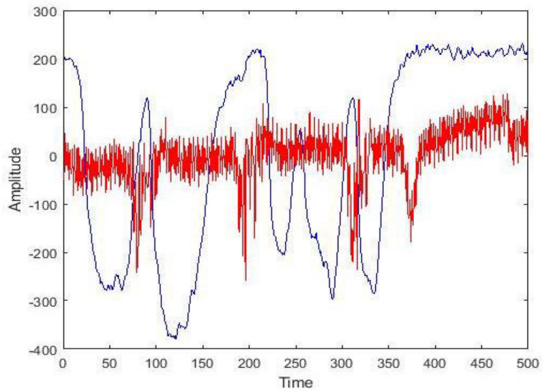
A. OFFLINE TESTS RESULTS

Two datasets are used to train and test the proposed method. The first dataset is multi-subject (suba, subh, and subz) data, and the second is a single subject data subz. Fourth-order Butterworth bandpass filter is used to preprocess the data for removing artifacts. DWT_AR is used to extract features using a single level (db1) to extract the frequency range (0.1-15) Hz for eye-blinks and from the frequencies (15-30) Hz for jaw squeezing. Figure 5 shows the time-series signals of both jaw squeeze and eyeblinks. Figures 6 and 7 shows the FFT of these signals, in these figures the irrelevant channels eliminated to ensure the clearance of the signals. As we use the approximates and details of DWT, see Figures (8-15).

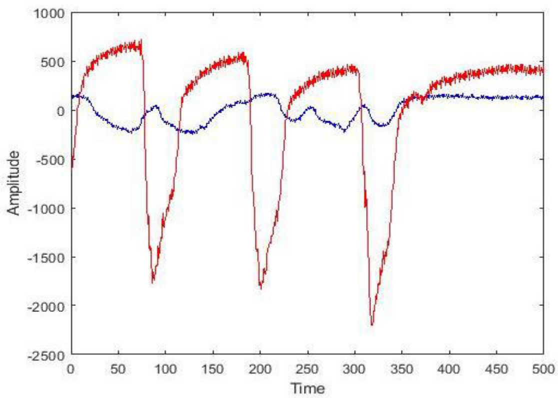
Multiple sizes of training sets are used to extract the features from 16 channels using multiple AR orders, as shown in Figures 16 and 17. Figure 18 shows the accuracy of both datasets using single, double, and four channels, which 100% using only two channels (Ch1=FP1 and Ch9=F7). As shown in Figure 19, DWT_AR can be selected as the best method among the other four methods.

A comparison is made with four methods in the literature. The first method is the threshold in which no training is needed and very efficient in real-time applications when no previous data is available. The individual threshold is known [26]. AR is used in [28], while the last method is time-domain features (Max, Min, and Kurtosis) used in [23].

Both CSP and DWT_AR give 100% detection for single-user data with small features vector, the long features



a) Signals recorded from channel 1



b) Signals recorded from channel 9

FIGURE 5. Time series of multiple jaw squeezes (red signal) and five eyeblinks (blue signal).

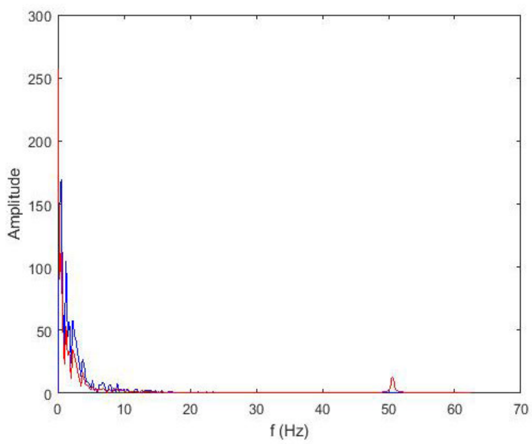


FIGURE 6. FFT of multiple jaw squeezes (both ch1 and ch9).

vectors increase the processing time and could sometimes insert irrelevant features that may reduce the performance so it is better to choose the most relevant features and keep the features vector small. Table 4 shows a comparison according to the features vector size of the methods used in this work. The CSP method provides outstanding results when the training data is enough and used for the single-user application. For multi-user and single-user applications, DWT_AR give

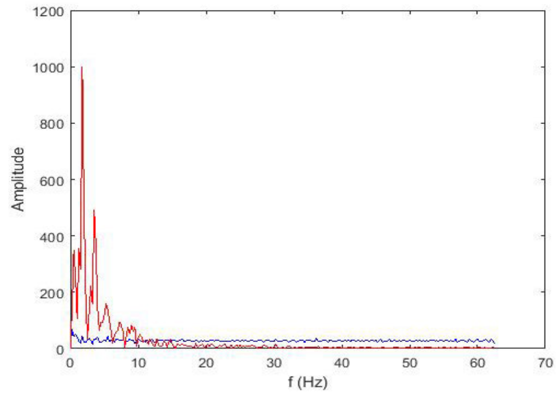


FIGURE 7. FFT of multiple eyeblinks(both ch1 and ch9).

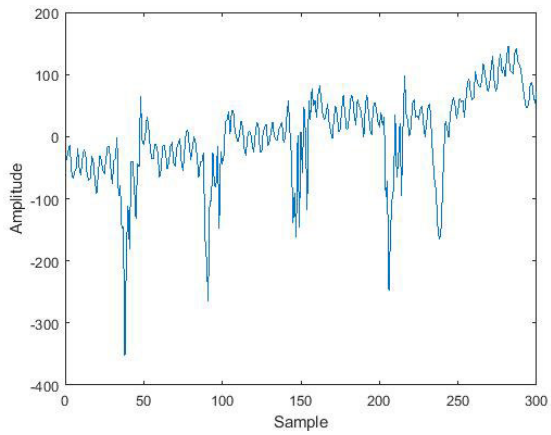


FIGURE 8. The DWT approximate coefficients of multiple jaw squeeze (only ch1).

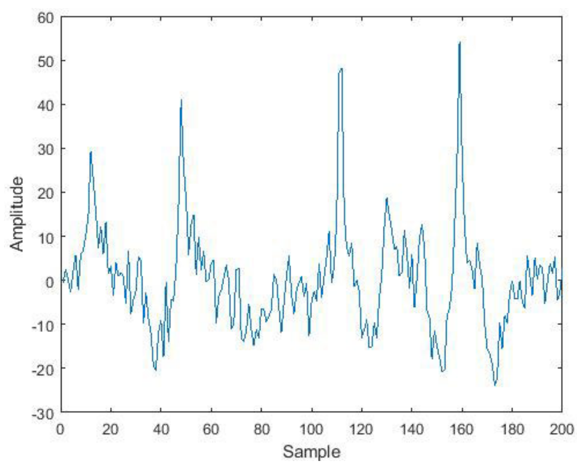


FIGURE 9. The DWT approximate coefficients of multiple eyeblinks (only ch1).

100% detection using small training data set and small features vector.

B. REAL-TIME TESTS AND ONLINE RESULT

Subz is trained on the system and tested it both offline and online for both the straight and not straight paths, as shown

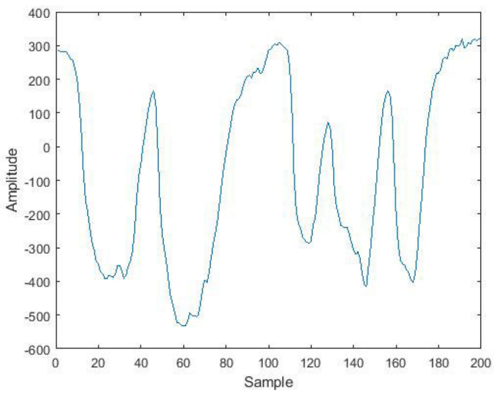


FIGURE 10. The DWT details coefficients of multiple jaw squeeze (only ch1).

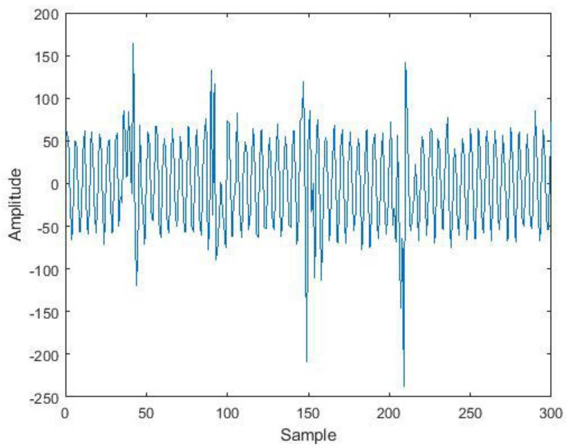


FIGURE 11. The DWT details coefficients of multiple eyeblinks (only ch1).

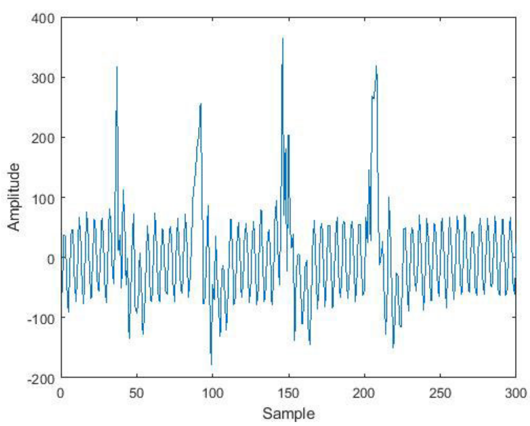


FIGURE 12. The DWT approximate coefficients of multiple jaw squeeze (only ch9).

in Figure 20. Table 5 shows the commands that virtually represent the curved path in Figure 15-b that used for training on the computer.

Every person has his/her own timing in blinking and other actions as we want to convert blinks and jaw squeezes to commands, each command has its own timing to be recognized.

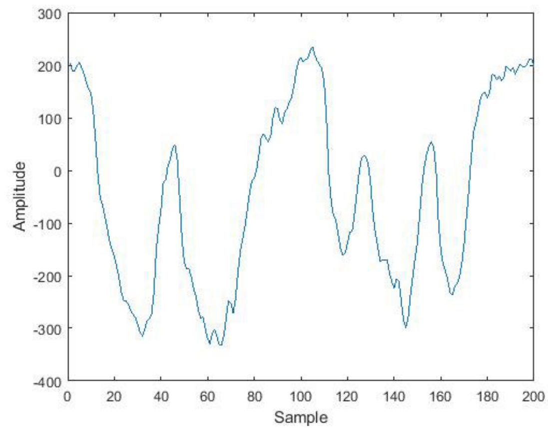


FIGURE 13. The DWT approximate coefficients of multiple eyeblinks (only ch1).

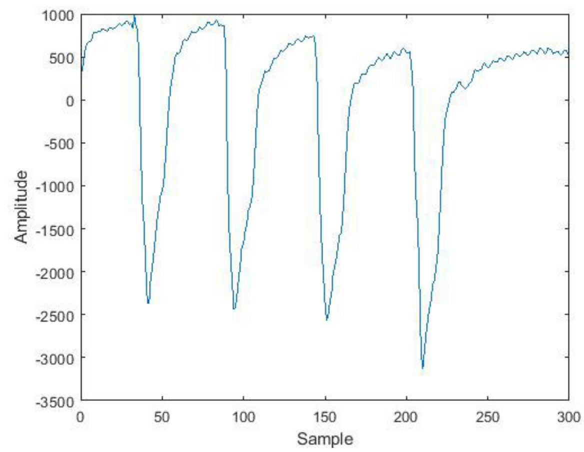


FIGURE 14. The DWT details coefficients of multiple jaw squeeze (only ch9).

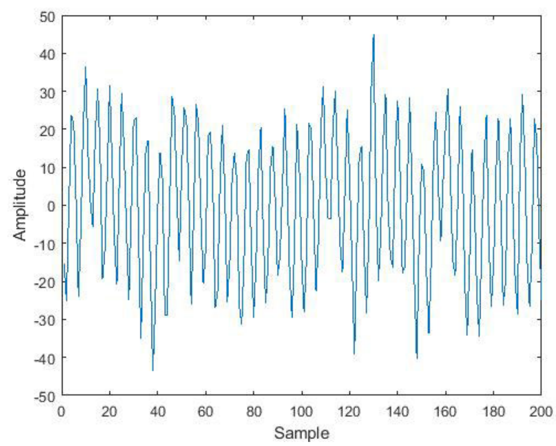


FIGURE 15. The DWT details coefficients of multiple eyeblinks (only ch9).

In the training phase, Subz finds out the timing of each command, so the best number of movements for each command is chosen (i.e., how many jaws squeeze that the user should do, to get logic one and how many eyes blinks the user should

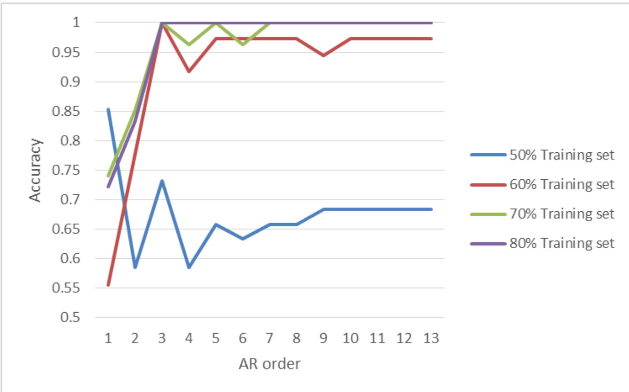


FIGURE 16. Accuracy VS AR order for different sizes of training datasets (Dataset1).

FIGURE 17. Accuracy VS AR order for different sizes of training datasets (Dataset2).

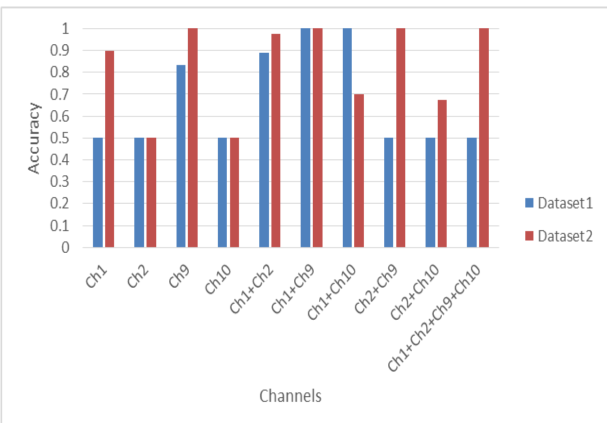


FIGURE 18. Accuracy VS different channels for two datasets (Dataset1 and Dataset2).

do to get logic zero). Subz tests the system in real-time first using a virtual path on the computer. Table 6 shows the offline straight path results.

The training makes the subject able to know how many movements he/she should do to get the right command. For subz, two jaw squeeze equals to logic one, and four eye-blinks equal to logic zero. Table 7 shows the results of training subz on the virtual control path, as in Table 5. The powered wheelchair is driven using two DC motors, one to the left and the other to the right for the back wheels of the chair. Figure 21 shows the real picture of the wheelchair and the interface circuit. To go forward, both motors are operated in the forward direction, and for backward movement also both motors are operated, but in the reverse direction. When we want to drive the chair to turn to left or right the opposite motor should be on and the motor of the same direction is off so the chair could turn to the desired direction. Two high current H-bridge motor drivers are used to interface the motor of the chair to the microcontroller one for each motor. As we already need the computer to inform the user to give a command, Matlab is used for processing the data streams from the headset, as shown in Figure 22.

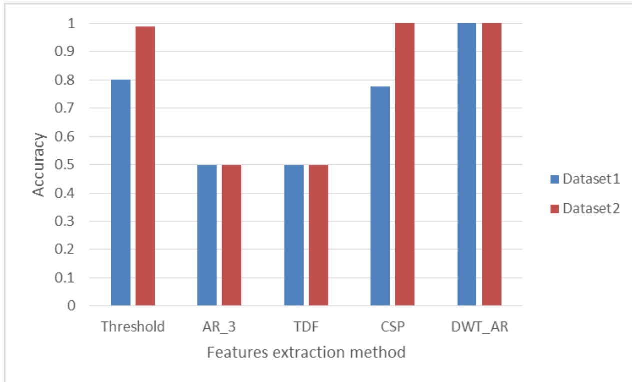


FIGURE 19. Comparison of different methods.

TABLE 4. Features vector size.

Method	Features Vector size
Threshold	No features
AR_3	48
TDF	48
CSP	16
DWT_AR	3 or 6 depending on the number of channels(1 or 2)

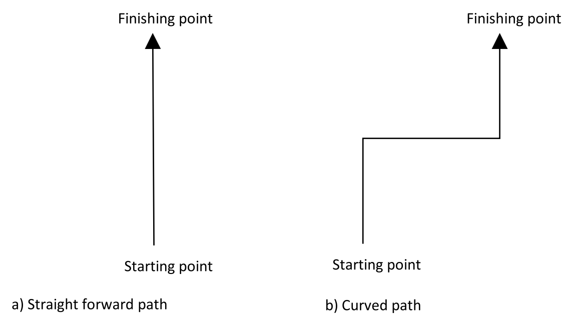


FIGURE 20. The paths that used to train subz.

TABLE 5. The commands of the curved path.

Command	Number of repeating times
Forward	1
Keep moving	3
Stop	1
Right	1
Forward	1
Keep moving	2
Stop	1
Left	1
Forward	1
Keep moving	1
Stop	1
Total	**14**

TABLE 6. The accuracy of training Subz on the virtual straight forward path.

Training day	No. of correct commands	Accuracy
Day1	15	75%
Day2	17	85%
Day3	15	75%
Day4	18	90%
Day5	16	80%
Day6	15	75%
Average accuracy		80%

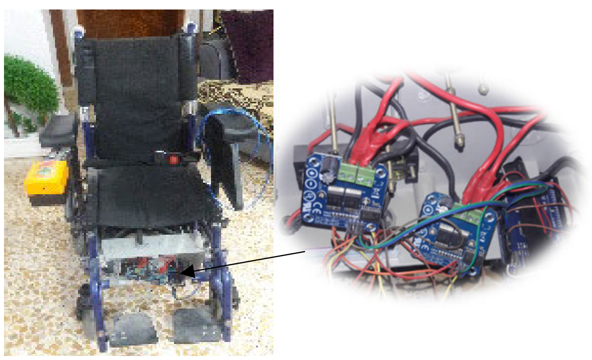
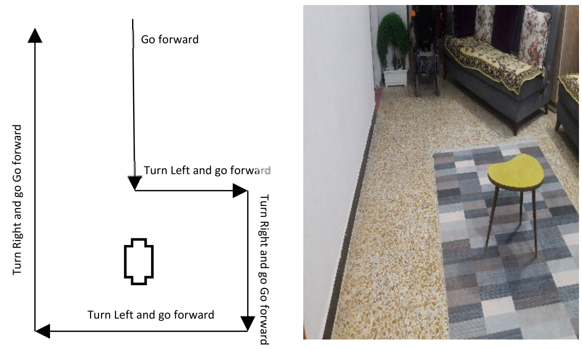


FIGURE 21. The real picture of the wheelchair and the interfacin.

The pseudo-code of the practical implementation of wheelchair navigation is as follows

```
FW=0;BW=1;S=0;L=0;R=0;
Do
    Data=Read a data segment of 4.5 sec length
```



a) Scheme of the real path b) The real path

FIGURE 22. The real path tested by sub.

TABLE 7. The accuracy of training Subz on the virtual curved path.

Training day	No. of correct commands	Accuracy
Day1	7	50%
Day2	9	64%
Day3	9	64%
Day4	12	85%
Day5	10	71%
Day6	12	85%
Average accuracy		70%

```
Filtered_data=BPF(Data,Type,Order)
For segment=1 to 3
    Seg= Filtered_data(segment*1.5 sec)
    [approximat,details]=DWT(seg,db1)
    Features1=AR(approximate,P)
    Features2=AR(details,P)
    Features=[Features1,Features2]
    Code(segment)=SVM(Features)
End for
If Code=001 then
    If (BW=1 and S=1) or (S=1 and L=1) or (S=1
    and R=1)
        {FW=1;BW=0;S=0;L=0;R=0;
        Go Forward}
    Else if (FW=1 and S=1)
        {Go Backward
        FW=0;BW=1;S=0;L=0;R=0;}
    Else if (S=0 and FW=1)) or (S=0 and BW=1)
        {Stop
```

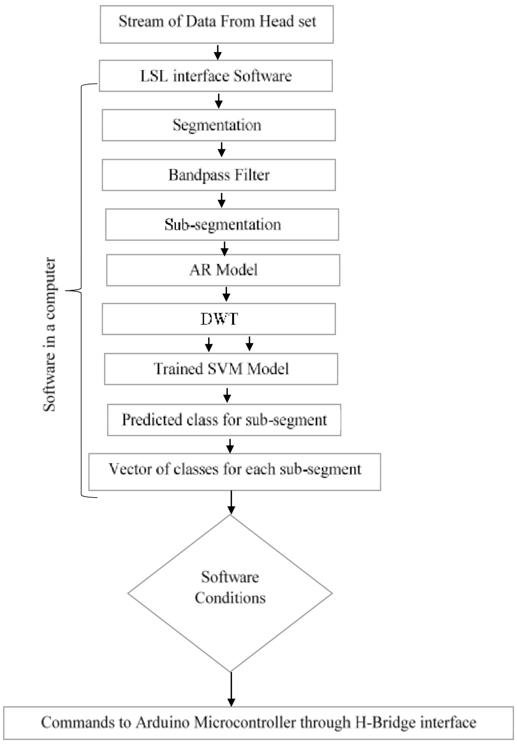
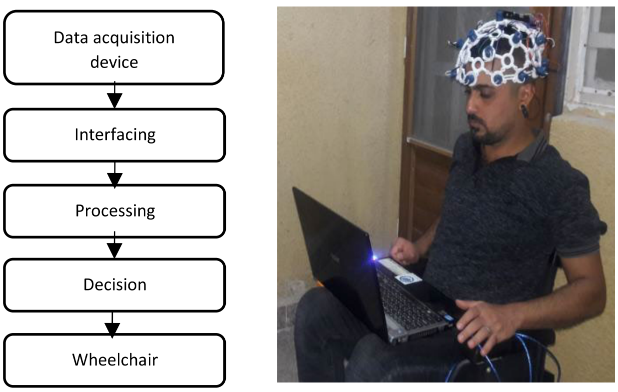
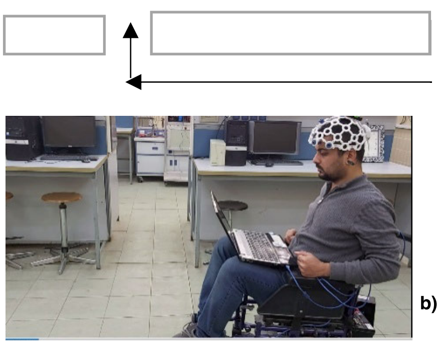


FIGURE 23. The detailed steps of the syste.



a) Steps of the system b) Real system with suba

FIGURE 24. The steps of the system.



a) Scheme of the real path

b) The real path

FIGURE 25. The real path tested by sub.

S=1;}
End if
Else if (S=1 and BW=0)

TABLE 8. The real-time testing information of subz for a straight path.

Training day	Attempt statues	Comments
Day1	success	Stop after a while
Day2	Success	
Day3	Fail	Did not stop
Day4	Fail	Did not stop
Day5	Fail	Did not stop
Day6	Success	
Day7	Success	
Day8	Success	
Day9	Success	

TABLE 9. The real-time testing information of suba for a curvedpath.

Training day	Attempt statues	Comments
Day1	Fail	Stop after a while
Day2	Success	
Day3	Success	
Day4	Fail	Did not stop in the correct place
Day5	Fail	Did not stop in the correct place
Day6	Success	
Day7	Success	
Day8	Success	

If Code =011
{Turn Left
FW=0;BW=0;S=1;L=1;R=0;}
Else if Code =101
{Turn Right
FW=0;BW=0;S=1;L=0;R=1;}
Else
Keep Moving
End if

TABLE 10. A comparison among different applications according to the number of actions versus the number of commands.

Ref.	No. commands	No. actions	Signal type	Features extraction method	Classifier	Application
[36]	2	3	MI	Adaptive Gaussian Mixture	LDA	Game
[37]	7	5	MI/P300/Eye-blink	CSP/CCA	SVM	Wheelchair navigation
[38]	8	7	MI/SSVEP	CSP/CCA	SVM	Wheelchair navigation
[32]	7	7	MI/SSVEP/Teeth clench	Bands power/CCA/DWT	Thresholds	Robot arm writing
[39]	4	2	MI	CSP/Bands Power	LDA	Robot navigation
Our system	5	2	Eye-blink/Jaw squeeze	DWT_AR	SVM	Wheelchair navigation

Else
Keep moving
End if
Until finishing the path
Stop moving

Figure 23 shows the detailed steps of the wheelchair BCI system. This flowchart contains a software conditions block, this block depends on the conditions of the pseudo opcode. Figure 24 shows the brief steps of the system and the real picture of suba.

Two attempts each day are recorded for training on the straight forward path, and each attempt consists of ten collections of commands (Forward, Backward, and Stop) taking some rest between the attempts. Table 8 shows the results of subz testing the straight forward path on the computer Path= (Forward, Keep moving, and stop). In this table, the results of the subject show that after five days of training on the path the subject could generate the correct commands in the current time. In the third, fourth and fifth days, the subject could not generate the stop command at the right time or could not generate it at all (the subject could not tune the exact timing of the command). After the training of the subject virtually a real-time testing takes place using a powered wheelchair, the wheelchair speed is 8 cm/sec to make the subject able to give the right command. An ordinary room is used for testing; the room is 12 m^2. Two tests are implemented, one with an obstacle and the other without, as shown in Figure 25.

The total path length is 8m from the starting point to the finish point, which is the same place. The first test was driving the wheelchair without an obstacle. This test takes 20 minutes and 50 commands. In this test, the turn left/right takes place for 1 sec. Then an automatic stop takes place if the subject wants to turn with a wider angle. Another command should be issued. This path finished with fifty commands, including the go back to avoid the walls. Suba tests the forward path without offline test just follow the instructions of subz he could drive the chair after the third attempt.

The second test is done with an obstacle in the middle of the room, and the turning command takes place for 3 sec instead of 1, so the turning angle is wider using a single command. Here the number of commands to finish the path is less than the first test, which is thirty-five only. The path finished in 20 minutes because of the obstacle.

The third test is done to drive the chair between two obisticals in a lab environment as shown in Figure 25, Table 9 shows the results of suba.

In [39] the authors used the MI signal. This signal is recorded by 64 wet channels, while our system only uses two dry channels. The other works used hybrid signals and a high number of actions to create commands.

Table 10 shows a comparison with different previous works that implement different applications, the only work has almost the same number of commands to actions is [39].

IV. CONCLUSION

A robust method to develop a hybrid BCI application that detects the eye-blinks and jaw squeezing has been presented. The proposed method was based on DWT_AR that was capable to separate the frequency ranges of eye-blinks and jaw squeezing while the AR reduces the features vector to obtain a low dimensional vector that includes five features.

DWT_AR was compared against four other methods, the present work was able to recognize both movements with an accuracy of 100% for both single subject training and multi-subject training using a small training set that is around less than 70 samples. It was concluded that single action could be used to create five commands based on binary codes. The prototype was worked on two movements to create five active commands that include: forward, backward, stop, turn left and turn right all that using only jaw squeeze and eye-blink signals. The test results were approved to drive the chair successfully.

REFERENCES

[1] N. Alamdari, A. Haider, R. Arefin, A. K. Verma, K. Tavakolian, and R. Fazel-Rezai, "A review of methods and applications of brain computer interface systems," in *Proc. IEEE Int. Conf. Electro Inf. Technol. (EIT)*, May 2016, pp. 345–350.

[2] P. Shende and V. Jabade, "Literature review of brain computer interface (BCI) using Electroencephalogram signal," in *Proc. Int. Conf. Pervasive Comput. (ICPC)*, 2015, pp. 1–5.

[3] L. Nicolas-Alonso and J. Gomez-Gil, "Brain computer interfaces, a review," *Sensors*, vol. 12, pp. 1211–1279, Feb. 2012.
[4] W. Tatum, A. Husain, S. Benbadis, and P. Kaplan, *Handbook of EEG Interpretation*, 1st ed. New York, NY, USA: Demos Medical, 2008.
[5] L. van Dokkum, T. Ward, and I. Laffont, "Brain computer interfaces for neurorehabilitation–its current status as a rehabilitation strategy post-stroke," *Ann. Phys. Rehabil. Med.*, vol. 58, pp. 3–8, 2015.
[6] P. Diez, *Smart Wheelchairs and Brain-Computer Interfaces*, vol. 9780128128930, 1st ed. Amsterdam, The Netherlands: Elsevier, 2018.
[7] S. Sanei and J. Chambers, *EEG Signal Processing*. Hoboken, NJ, USA: Wiley, 2007.
[8] A. Tanwani, J. Afridi, M. Shafiq, and M. Farooq, "Guidelines to select machine learning scheme for classification of biomedical datasets," in *Proc. EvoBIO*, in Lecture Notes in Computer Science, vol. 5483. Berlin, Germany: Springer-Verlag, pp. 128–139, 2009.
[9] F. Lotte, M. Congedo, A. Lécuyer, F. Lamarche, and B. Arnaldi, "A review of classification algorithms for EEG-based brain–computer interfaces," *J. Neural Eng.*, vol. 4, no. 2, pp. 1–24, 2007.
[10] F. Lotte, L. Bougrain, A. Cichocki, M. Clerc, M. Congedo, A. Rakotomamonjy, and F. Yger, "A review of classification algorithms for EEG-based brain–computer interfaces: A 10 year update," *J. Neural Eng.*, vol. 15, no. 3, Jun. 2018, Art. no. 031005.
[11] N. Aloysius and M. Geetha, "A review on deep convolutional neural networks," in *Proc. Int. Conf. Commun. Signal Process.*, 2017, pp. 0588–0592.
[12] H. M. Rashid and H. A. Jaber, "HD-sEMG gestures recognition by SVM classifier for controlling prosthesis," *Iraqi J. Comput., Commun., Control Syst. Eng.*, vol. 19, no. 1, pp. 10–19, 2019.
[13] M. Sovierzoski, F. Argoud, and F. de Azevedo, "Identifying eye blinks in EEG signal analysis," in *Proc. 5th Int. Conf. Inf. Technol. Appl. Biomed., Conjunct. 2nd Int. Symp. Summer School Biomed. Health Eng.*, Shenzhen, China, May 2008, pp. 10–19.
[14] H. Tan and Y.-J. Zhang, "Detecting eye blink states by tracking iris and eyelids," *Pattern Recognit. Lett.*, vol. 27, no. 6, pp. 667–675, Apr. 2006.
[15] V. Krishnaveni, S. Jayaraman, L. Anitha, and K. Ramadoss, "Removal of ocular artifacts from EEG using adaptive thresholding of wavelet coefficients," *J. Neural Eng.*, vol. 3, no. 4, pp. 338–346, Dec. 2006.
[16] G. Gomez-Herrero, W. De Clercq, H. Anwar, O. Kara, K. Egiazarian, S. Van Huffel, and W. Van Paesschen, "Automatic removal of ocular artifacts in the EEG without an EOG reference channel," in *Proc. 7th Nordic Signal Process. Symp. (NORSIG)*, Jun. 2006.
[17] Y. Okada, J. Jung, and T. Kobayashi, "An automatic identification and removal method for eye-blink artifacts in event-related magnetoencephalographic measurements," *Physiol. Meas.*, vol. 28, no. 12, pp. 1523–1532, Dec. 2007.
[18] S. Hoffmann and M. Falkenstein, "The correction of eye blink artefacts in the EEG: A comparison of two prominent methods," *PLoS ONE*, vol. 3, no. 8, p. e3004, 2008.
[19] F.-B. Vialatte, J. Solé-Casals, and A. Cichocki, "EEG windowed statistical wavelet scoring for evaluation and discrimination of muscular artifacts," *Physiol. Meas.*, vol. 29, no. 12, pp. 1435–1452, Dec. 2008.
[20] A. Królak and P. Strumiłło, "Eye-blink detection system for human–computer interaction," *Inf. Soc.*, vol. 11, no. 4, pp. 409–419, 2012.
[21] W. Kong, Z. Zhou, S. Hu, J. Zhang, F. Babiloni, and G. Dai, "Automatic and direct identification of blink components from scalp EEG," *Sensors*, vol. 13, no. 8, pp. 10783–10801, 2013.
[22] J. W. Matiko, S. Beeby, and J. Tudor, "Real time eye blink noise removal from EEG signals using morphological component analysis," in *Proc. 35th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, Jul. 2013, pp. 13–16.
[23] S. Rihana, P. Damien, and T. Moujaess, *EEG-Eye Blink Detection System for Brain-Computer Interface*. Berlin, Germany: Springer, 2013.
[24] R. N. Roy, S. Charbonnier, and S. Bonnet, "Eye blink characterization from frontal EEG electrodes using source separation and pattern recognition algorithms," *Biomed. Signal Process. Control*, vol. 14, pp. 256–264, Nov. 2014.
[25] M. N. Tibdewal, R. R. Fate, M. Mahadevappa, and A. Ray, "Detection and classification of eye blink artifact in electroencephalogram through discrete wavelet transform and neural network," in *Proc. Int. Conf. Pervas. Comput. (ICPC)*, Jan. 2015, pp. 1–6.
[26] W.-D. Chang, J.-H. Lim, and C.-H. Im, "An unsupervised eye blink artifact detection method for real-time electroencephalogram processing," *Physiol. Meas.*, vol. 37, no. 3, pp. 401–417, Mar. 2016.
[27] S. Al-gawwam and M. Benaissa, "Robust eye blink detection based on eye landmarks and Savitzky–Golay filtering," *Information*, vol. 9, no. 4, p. 93, 2018.
[28] V. Lawhern, W. D. Hairston, K. McDowell, M. Westerfield, and K. Robbins, "Detection and classification of subject-generated artifacts in EEG signals using autoregressive models," *J. Neurosci. Methods*, vol. 208, no. 2, pp. 181–189, Jul. 2012.
[29] H. Wang, Y. Li, J. Long, T. Yu, and Z. Gu, "An asynchronous wheelchair control by hybrid EEG–EOG brain–computer interface," *Cogn. Neurodyn.*, 2014.
[30] J. Lin and Z. Jiang, "An EEG-based BCI system to facial action recognition," *Wireless Pers. Commun.*, 2016.
[31] J. Heo, H. Yoon, and K. Park, "A novel wearable forehead EOG measurement system for human computer interfaces," *Sensors*, vol. 17, no. 7, p. 1485, 2017.
[32] Q. Gao, L. Dou, A. N. Belkacem, and C. Chen, "Noninvasive electroencephalogram based control of a robotic arm for writing task using hybrid BCI system," *BioMed Res. Int.*, vol. 2017, pp. 1–8, 2017, Art. no. 8196485.
[33] M. Khoshnam, E. Kuatsjah, X. Zhang, and C. Menon, *Hands-Free EEG-Based Control of a Computer Interface Based on Online Detection of Clenching of Jaw*. Springer, 2017.
[34] R. Palaniappan, R. Paramesran, S. Nishida, and N. Saiwaki, "A new brain-computer interface design using fuzzy ARTMAP," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 10, no. 3, pp. 140–148, Sep. 2002.
[35] J. Jiang, Z. Zhou, E. Yin, Y. Yu, Y. Liu, and D. Hu, "A novel Morse code-inspired method for multiclass motor imagery brain–computer interface (BCI) design," *Comput. Biol. Med.*, vol. 66, pp. 11–19, Nov. 2015.
[36] B. A. S. Hasan and J. Q. Gan, "Hangman BCI: An unsupervised adaptive self-paced brain–computer interface for playing games," *Comput. Biol. Med.*, vol. 42, no. 5, pp. 598–606, May 2012.
[37] H. Wang, Y. Li, J. Long, T. Yu, and Z. Gu, "An asynchronous wheelchair control by hybrid EEG–EOG brain–computer interface," *Cognit. Neurodynamics*, vol. 8, no. 5, pp. 399–409, Oct. 2014.
[38] L. Cao, J. Li, H. Ji, and C. Jiang, "A hybrid brain computer interface system based on the neurophysiological protocol and brain-actuated switch for wheelchair control," *J. Neurosci. Methods*, vol. 229, pp. 33–43, May 2014.
[39] M. Aljalal, R. Djemal, and S. Ibrahim, "Robot navigation using a brain computer interface based on motor imagery," *J. Med. Biol. Eng.*, vol. 39, no. 4, pp. 508–522, Aug. 2019.

ZAINEB M. ALHAKEEM (Member, IEEE) was born in Basrah, Iraq, in 1983. She received the B.S. and M.S. degrees in computer engineering from the University of Basrah, in 2005 and 2008, respectively, where she is currently pursuing the Ph.D. degree with the Electrical Engineering Department.

Her research interests include biomedical engineering, digital signal processing, and computer engineering.

RAMZY S. ALI (Senior Member, IEEE) received the B.Sc. and M.Sc. degrees in electrical engineering and control and computers engineering from the University of Basrah, Basrah, Iraq, in 1985 and 1989, respectively, and the Ph.D. degree in control and systems from the Saint-Petersburg State Polytechnic University, Russia, in 2003.

He is currently an Assistant Professor with the University of Basrah. His teaching interests include intelligent control systems, robust control systems, microprocessor and microcontrollers, and industrial automation. His research interests include intelligent control of robotics, computational intelligence, computational methods and optimization, chaos and nonlinear dynamics, renewable electrical energy systems, and PLC applications in industrial and engineering education. He currently serves as a Co-Editor of the *Basrah Journal for Engineering Sciences*.

RAED A. ABD-ALHAMEED (Senior Member, IEEE) is currently a Professor of electromagnetic and radiofrequency engineering at the University of Bradford, U.K. He has long years' research experience in the areas of radio frequency, signal processing, propagations, antennas, and electromagnetic computational techniques, and has published more than 400 academic journal and conference papers. Besides, he is a coauthor of three books and several book chapters. He is currently the leader of radiofrequency, propagation, sensor design, and signal processing; in addition to leading the Communications Research Group for years within the School of Engineering and Informatics, Bradford University, U.K. He is a Principal Investigator of several funded applications to EPSRCs and the Leader of several successful knowledge Transfer Programmes such as with Pace plc, YW plc, ITEG ltd, Datong plc, Emkay ltd, and Two limited. He has also been a co-investigator in several funded research projects, including Nonlinear and Demodulation Mechanisms in Biological Tissue (Department of Health, Mobile Telecommunications and Health Research Programme; and Assessment of the Potential Direct Effects of Cellular Phones on the Nervous System (EU: collaboration with six other major research organisations across Europe). His research interests include computational methods and optimizations, wireless and mobile communications, sensor design, EMC, beam steering antennas, energy-efficient PAs, and RF predistorter design applications. He is a Fellow of the Institution of Engineering and Technology, a Fellow of the Higher Education Academy, and a Chartered Engineer. Recently, he was awarded the Business Innovation Award for his successful KTP with Pace on the design and implementation of MIMO sensor systems. He is the Chair of several successful workshops on Energy Efficient and Reconfigurable Transceivers (EERT): Approach towards Energy Conservation and CO2 Reduction that addresses the biggest challenges for future wireless systems. He has also been appointed as a Guest Editor for the *IET Science, Measurements and Technology Journal*, since 2009 and in 2012, respectively. He has also been a Research Visitor for Wrexham University, Wales, since September 2009, covering the wireless and communications research areas.