American Journal of Engineering Research (AJER)
e-ISSN: 2320-0847 p-ISSN : 2320-0936
Volume-10, Issue-6, pp-123-130
www.ajer.org
Research Paper Open Access

Building an EEG-based BCI systems for Remote Device Control using CNN network

Hanh Phuc Nong[1], Mai Thuong Duong Thi[2]
1 Thai Nguyen University - Lao Cai Campus, Lao Cai, Vietnam
2 Thai Nguyen University of Information and Communication Technology (ICTU)

ABSTRACT :*Remote device control via electroencephalogram (EEG) signal is a problem with great applicability in life. This problem is not only applied to people with disabilities (loss of mobility) but also to ordinary people when direct contact becomes unsafe (toxic environment, Covid-19 epidemic). To accomplish this problem, we must apply high-precision identification techniques. Currently, many practical applications have proven that convolutional neural networks (CNN) are an effective deep learning tool in improving the quality of recognition systems. In this paper, the authors will present the application of CNN to classify EEG signals to build a remote device control system. The system has been tested in practice and gives high accuracy performance results.*

KEYWORDS *Electroencephalogram, Brain Computer Interface system, Convolutional Neural Network, Remote device control.*

Date of Submission: 09-05-2021 Date of acceptance: 09-06-2021

I. INTRODUCTION

Currently, with the explosive development of modern technologies, remote device control has become a familiar solution to help improve people's quality of life. The key point of this problem is to understand the user's "idea" from which to control the desired devices in a reasonable way. This idea is usually expressed through a remote control (by Bluetooth or radio waves), a smart phone (via a mobile environment or the Internet) [10], gestures or the user's voice [8].

In our lives, there are many people who have completely lost the ability to move their limbs and lose the ability to communicate due to congenital defects or after having a stroke or an accident. All their activities including personal hygiene are completely dependent on the care of their family or their doctor. However, the inability to move and use language creates a difficult barrier for caregivers to understand and care for the patient. This process can take many years, causing a lot of inconvenience and fatigue for caregivers. Therefore, understanding the “patient's idea" through "inner" signals such as electroencephalography (EEG) becomes necessary and important to help them reintegrate into the community. This need has led to the steady growth of Brain Computer Interface (BCI) applications [1].

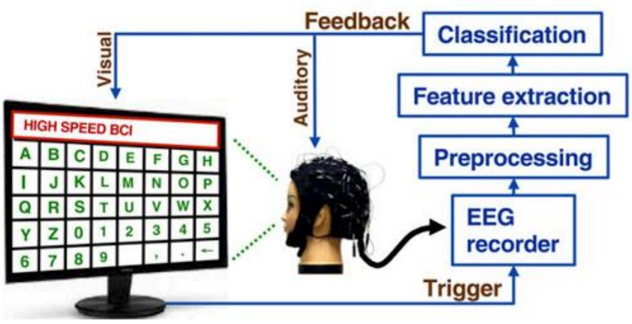

Figure 1. Basic block diagram of a BCI system [12]

Up to now, in order to develop recognition applications based on EEG signals, there have been many published works and different approaches for solving the problem of brain wave recognition [6], [2]. All efforts are focused on the task of building a fast acting and highly accurate BCI system depicted as Figure 1 [12]. To accomplish this, all published studies focus on improving the quality of three main steps in the BCI system: preprocessing (signal amplification and digitization), feature extraction, and digitization. signal, signal classification.

Previously, with the aid of machine learning techniques such as neural networks, support vector machine ... we can build a few practical applications such as remote device control, identification some needs of patients... However, the results achieved by the above applications are still limited. This shows that further improving the performance of BCI systems is still a challenge for scientists [3], [5].

Recently, in the field of recognition, a tool of deep learning has been proposed. That is a convolutional neural network (CNN). This tool has a higher recognition efficiency than traditional machine learning algorithms and has been used a lot in real recognition problems [1].

However, the performance of a CNN network depends greatly on the determination of the network's architectural parameters. Therefore, in order to use CNN for remote device control problem, the challenge is to find the appropriate internal architecture (number of convolutional layers, architectural inside each convolutional layer…) so that when we apply it to the problem of specific EEG signal recognition, it will give the best results [7].

II. HARDWARE DESIGN

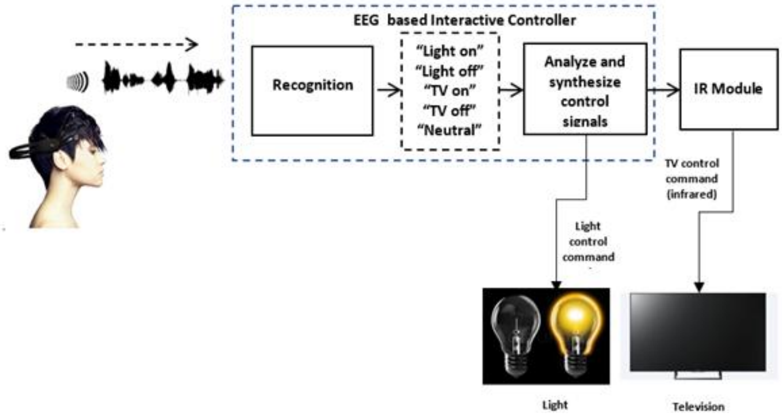

Figure 2. Block diagram of remote device control system via EEG signal

Figure 2 depicts the block diagram of the device control system through brain waves. In which, the hardware structure of the devices is as follows:

2.1. Caps for EEG signal acquisition and preprocessing

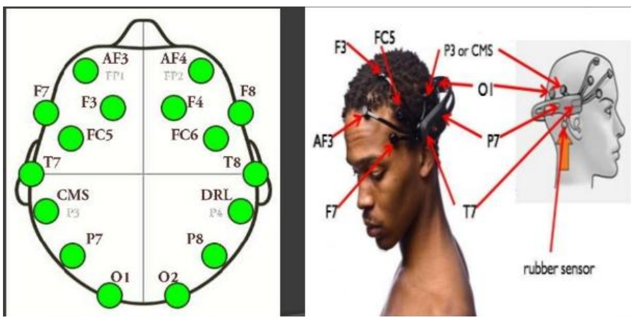

Figure 3. Emotiv Epoc+ cap and 16 electrode positions

To collect EEG signals, we use Emotiv EPOC+ wireless caps. This cap has 16 electrodes. These electrodes are arranged based on the international standard for 10-20 electrode system. Figure 3 depicts the Emotiv Epoc+ cap and the locations of the 16 electrodes. In which, there are 14 electrodes located at positions AF3, F7, F3, FC5, T7, P7, O1, O2, P8, T8, FC6, F4, F8, AF4 and 2 reference electrodes. EPOC+ caps need to be moistened with a special solution to improve conductivity.

2.2. The central controller

In terms of hardware, this controller can be a personal computer or a Raspberry pi 3 microcontroller depending on the stage of use. The PC is used in the training phase of the CNN network. After the training is complete, in the implementation phase, the CNN Network is installed on the Raspberry pi 3 microcontroller. This microcontroller will receive the EEG signal from the Emotiv Epoc+ hat, classifying the signal based on the CNN network trained in pre-phase, encode classes into code and pass to output communication modules. The output modules will control the lights and TV according to the received command codes. Raspberry Pi 3 is a miniature embedded computer. Therefore, its use will contribute to the simplicity of deploying open source applications that communicate with I/O devices.

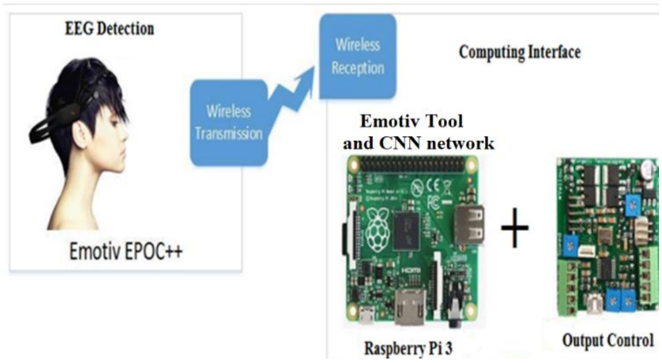

Figure 4. Central controller and I/O interface

2.3. Output communication modules

- *Infrared communication module*

Figure 5. Pairing Raspberry and infrared transceiver module for TV control

This module receives the command code from the recognition module, generates an infrared signal and transmits it to the corresponding device to control the TV on and off.

- *Light control module*

To turn on/off the light, the Raspberry Pi will issue a control command through the GPIO port to the relay module. We will use the relay module as shown in Figure 6. This module consists of 8 small relays for backup purposes.

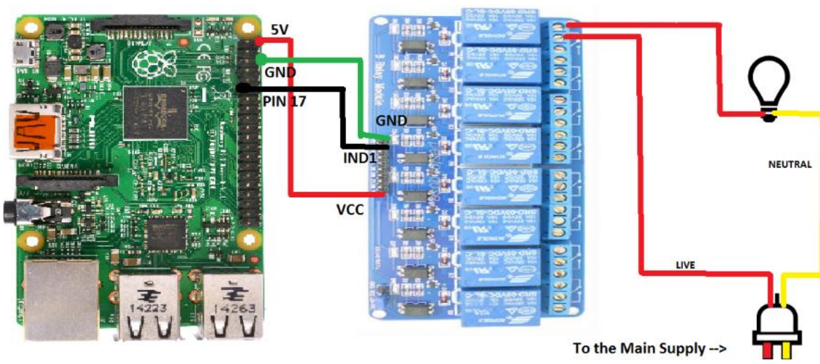

Figure 6. Communication between Raspberry Pi 3 and Relay Module for light control

III. SOFTWARE DESIGN

3.1. EEG signal acquisition module

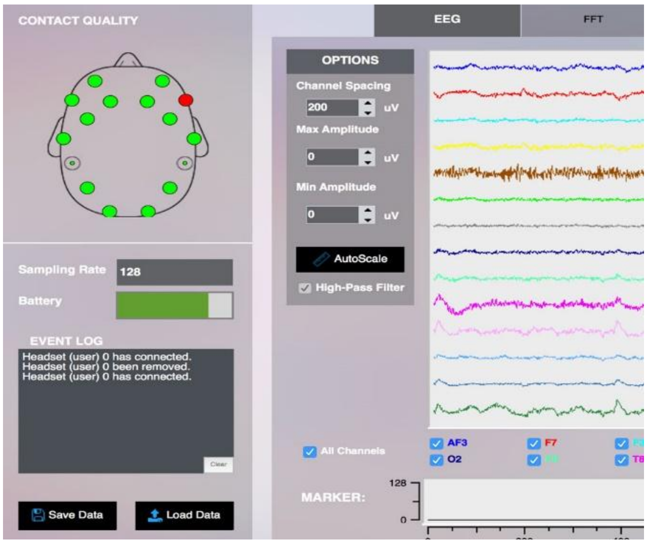

Figure 7. Acquisition of EEG signal using Emotiv SDK software

To get the raw EEG data from the Emotiv Epoc+ hat, we used the Emotiv SDK software module provided by Emotiv company. This module is installed on MATLAB 2020 platform to support CNN network training on PC. We also use this tool but install it on Python to perform EEG signal recognition on Raspberry PI after training the CNN network. This tool is useful for experiment design, media preparation and configuration. The tool also helps to collect EEG data in a structured and systematic way, which can be connected to the CNN network training tool and other programs that control peripheral devices.

3.2. EEG signal recognition module using CNN network

- *Introduction to CNN*

Among the types of neural networks, convolutional neural network (CNN) is one of the models for image recognition and classification. Technically, the CNN model is used to train as well as test the image recognition performance. Each input image will be passed through a series of convolutional layers with filters, aggregating fully connected layers, and applying SoftMax function to classify objects with probability value between 0 and 1. Figure 8 depicts the entire CNN process for input image processing and classification of image objects based on 2D pixel space values.

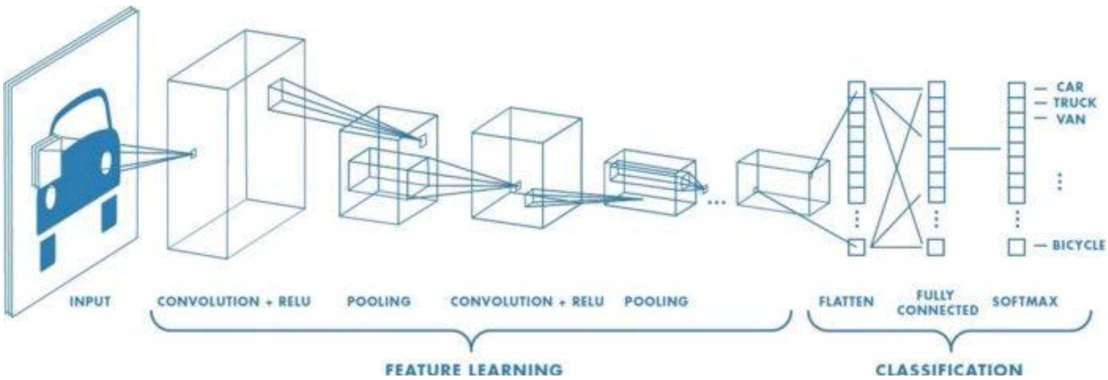

Figure 8. CNN network architecture for image recognition [13]

- *CNN network for EEG signal recognition*

The remote device control system is expected to allow users to use 5 control commands via brain waves. Therefore, to classify the control commands, we use the neural network CNN with the architecture as shown in Figure 9.

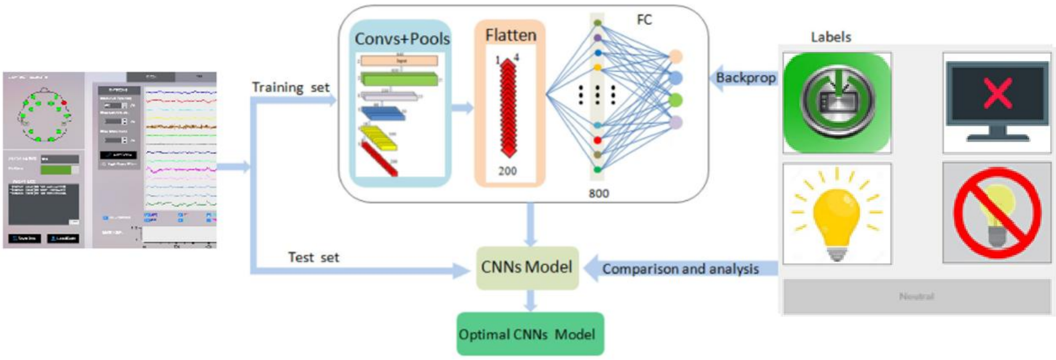

Figure 9. CNN network architecture for training device control commands

During the network training phase, the labeled EEG data (i.e. data from a known operator request) is recorded and sent to the CNN neural network which is installed on the PC to train the model.

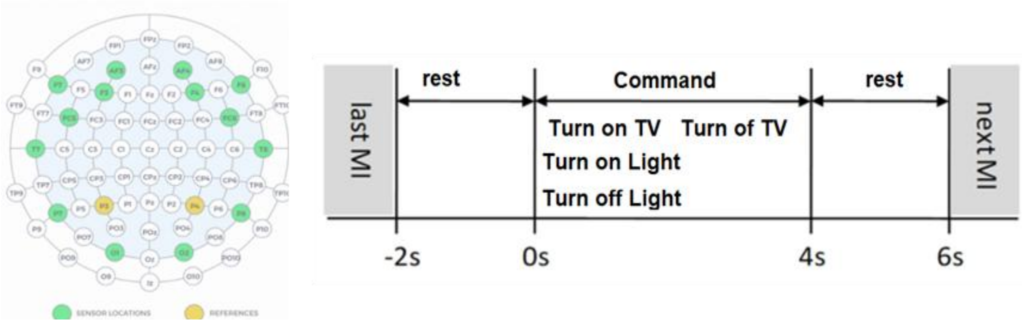

Figure 10. EEG signal sampling time allocation

To prepare data samples for network training and to evaluate the recognition results of the CNN network, users are required to look at the images on the interface, focus their thoughts for 8 seconds so that the software has time to collect EEG signals. During these 8s, the first 2s and the last 2s are used for preparation and rest. The actual data for network training is only recorded in the 4 middle seconds (Figure 10).

The following samples are also tested for correlation with the previous samples. This calculation is intended to assist the user in deciding whether or not to choose as a suitable sample data for network training.

Based on Figure 10 we can see that the Emotiv cap with 14 electrodes is used. With an electrode, we need to collect the variation of the (digital) signal over time. The signal sampling frequency as described in (Figure 9) is 128 Hz. The signal sampling is done in 4s. Thus, each electrode will also supply 512 samples. Therefore, each data sample fed into the CNN network will be a two-dimensional array of size 14 x 512.

During the recognition phase, unknown EEG data is fed into a trained CNN network (installed on Raspberry pi 3) to make a decision on the most appropriate command.

Each of the five thought commands (television on, tv off, light on, light off, rest) were practiced 250 times on the same person. The recorded data is divided into sets of 80% for network training and 20% for testing. Accordingly, for each instruction, there will be 200 samples for network training and 50 samples for checking the accuracy of recognition. Thus, the sum of the samples for training the CNN network is 1000. The total of the test samples is 250.

- *Searching for optimal CNN neural network structure for EEG signal recognition*

The performance of the CNN network depends on the number of convolutional layers, the size of the Local Receiving Field (LRF), and the Feature Maps (FM). In fact, depending on the specific application, we choose different parameters. In this paper, we have built the software on MATLAB 2020 as described in Figure 11. According to the "trial/error" method, we change these parameters in turn. Finally, we find the CNN network to achieve the best performance with 98.73% prediction accuracy in 27.08 minutes using three convolutional layers, each with 7x7 LRF and 9 FM at C1, 18 FM in C2 and 36 FM in C3.

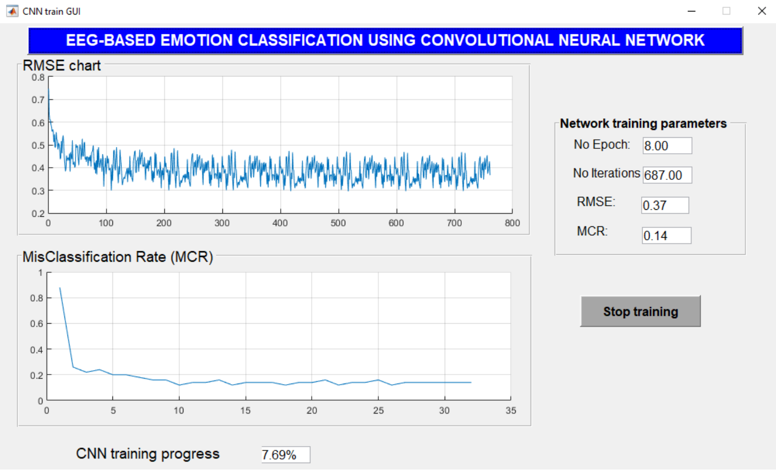

Figure 11. The interface for training CNN network to find the optimal parameter

3.3. Building application software

On the basis of the CNN network with the optimal parameters searched by the "trial and error" method described above, we proceed to build the application software. This software is responsible for illustrating the operation of a remote device control system based on brain waves. Figure 12 depicts the main interface of the application program. On this interface, the user can choose the training function to generate sample data for the training of the CNN network and the control function to check the actual performance of the system.

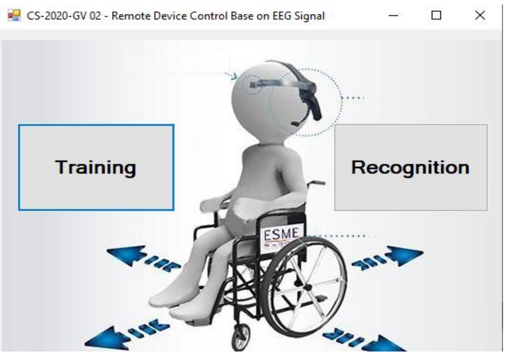

Figure 12. Main interface of the application program

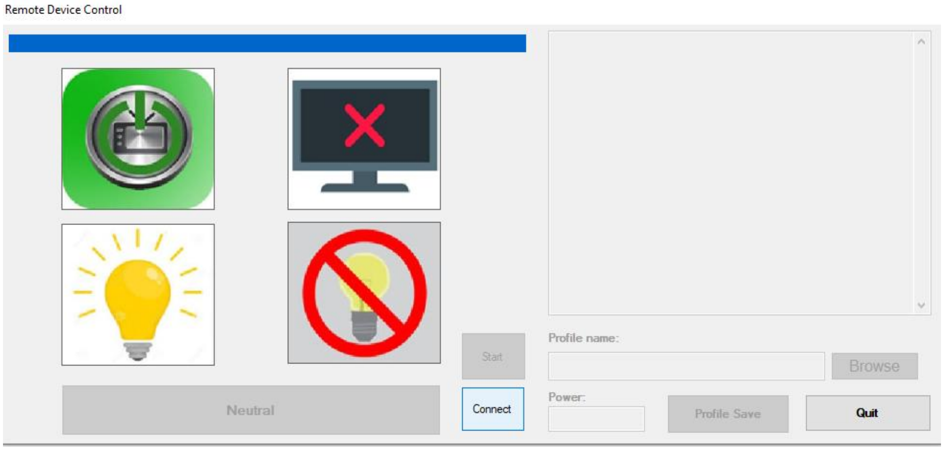

Figure 13. Interface of command training and device control software

Figure 13 illustrates the interface of the device control and instruction program. In the process of commands training, the screen will show illustrations for 5 commands to learn (4 commands to control the device and 1 command to rest). Users click on the corresponding command to focus their thoughts on that photo in the middle 4 seconds of 8 seconds (2 seconds before and the last 2 seconds are for preparation). During the test stage, the user focuses on which photo the corresponding command will be executed.

IV. SOME EXPERIMENTAL RESULTS

As mentioned above, we have built hardware and software for the device control system through EEG signal recognition. The system allows to identify 5 basic requirements of the operator such as turning on the lights, turning off the lights, turning on the TV, turning off the TV, and neutral.

To create a sample dataset for model training, for each command, the user will look at an image related to that statement on the PC interface. In order for the network training process to be accurate, users need to prepare carefully and focus their thoughts in 8 seconds. After enough practice with 200 samples for each command, the user will check the accuracy over 50 times and evaluate the results

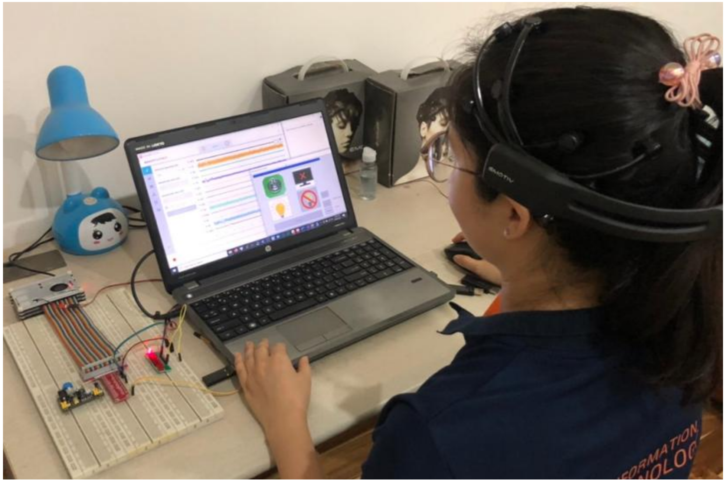

Figure 10. The actual architecture of the system

The performance results on 50 test data samples for each command are shown in Table 1. Thereby, it can be seen that the system works with an accuracy of over 92%. This shows great potential when developing this problem in practice.

Table 1. Some test results

Command	Number of test samples	Number of samples with correct results	Accurate recognition rate
Turn on the light	50	47	94
Turn off the light	50	46	92
Turn on the TV	50	48	96
Turn off the TV	50	46	92
Neutral	50	48	96

The system works with high precision. However, these are only the results tested on the same person. For others, we have to perform sampling and retraining the model. This will affect the performance of the system when applied to the patients because it will be difficult to perform the network training well when the user is sick. To overcome this problem, in the future, we will sample and train on many people with different circumstances so that the CNN network derives thinking features that are common to everyone. Besides, during the test, we found that the system still had difficulty with the delay of the control command.

V. CONCLUSION

It can be seen that building a device control system based on EEG signals with high accuracy is a very complicated problem. However, the implementation of the problem is completely feasible if we know how to apply research achievements in the fields of signal processing, mathematics, artificial intelligence, control systems. Especially, the application of CNN network will give very good recognition efficiency.

In this paper, we describe how to build a system (both hardware and software) for controlling the device via brain waves. In terms of hardware, the system collects EEG data using the Emotiv Epoc+ hat, sends it to the PC during network training and sends it to the Raspberry microcontroller during the identification process. The software installed on the microcontroller will apply the CNN network to recognize the patient's thoughts, transfer the command code to the actuator to turn on and off the lights and television. The system works with a high accuracy of over 92% and can be completely deployed in other practical applications such as controlling devices for smart homes, controlling electric wheelchairs, etc.

ACKNOWLEDGEMENT

This research was supported by the Thai Nguyen University - Lao Cai Campus research grant (under project # CS-2020-GV-02). We thank our colleagues from Thai Nguyen University - Lao Cai Campus, ICTU who provided insight and expertise that greatly assisted the research.

REFERENCES

[1]. Abiri, Reza & Borhani, Soheil & Sellers, Eric & Jiang, Yang & Zhao, Xiaopeng. (2018). A comprehensive review of EEG-based brain-computer interface paradigms. Journal of Neural Engineering. 16. 10.1088/1741-2552/aaf12e.

[2]. Cao, Zehong. (2020). A review of artificial intelligence for EEG- based brain−computer interfaces and applications. Brain Science Advances. 6. 162-170. 10.26599/BSA.2020.9050017.

[3]. Chamola, Vinay et al. "Brain-Computer Interface-Based Humanoid Control: A Review." Sensors (Basel, Switzerland) vol. 20,13 3620. 27 Jun. 2020, doi:10.3390/s20133620

[4]. Mohammad J. M. Zedan, Ali I. Abduljabbar, Fahad Layth Malallah, Mustafa Ghanem Saeed, "Controlling Embedded Systems Remotely via Internet-of-Things Based on Emotional Recognition", Advances in Human-Computer Interaction, vol. 2020, Article ID 8895176, 10 pages, 2020. https://doi.org/10.1155/2020/8895176

[5]. Nguyen, T. H., T. Duong, Toan Luu Van and P. Nguyen. "Building a brain wave system for patient supports." (2019).

[6]. Phuong Huy Nguyen, Thu May Duong, Thu Huong Nguyen, Thi Mai Thuong Duong, Combination of Wavelet and MLP Neural Network for Emotion Recognition System, International Journal on Future Revolution in Computer Science & Communication Engineering 4 (2018) 105-109

[7]. Rakhmatulin, Ildar. (2020). Review of Neural Networks in the EEG Signal Recognition. 10.20944/preprints 202011.0152.v1.

[8]. Ram, Sri. (2020). Hand Gesture and Voice Controlled Smart Vehicle. 164-167.

[9]. Rashid, M., Sulaiman, N., P P Abdul Majeed, A., Musa, R. M., Ab Nasir, A. F., Bari, B. S., & Khatun, S. (2020). Current Status, Challenges, and Possible Solutions of EEG-Based Brain-Computer Interface: A Comprehensive Review. Frontiers in neurorobotics, 14, 25. https://doi.org/10.3389/fnbot.2020.00025

[10]. Rizwan Majeed, Nurul Azma Abdullah, Imran Ashraf, Yousaf Bin Zikria, Muhammad Faheem Mushtaq, Muhammad Umer, "An Intelligent, Secure, and Smart Home Automation System", Scientific Programming, vol. 2020, Article ID 4579291, 14 pages, 2020. https://doi.org/10.1155/2020/4579291

[11]. Shilpa Bharti, Sukhman Preet Singh, An Enhanced Feature Extraction Method and Classification Method of EEG Signals using Artificial Intelligence, International Journal of Computer Applications (0975 – 8887), Volume 126 – No.10, September 2015.

[12]. https://medicalxpress.com/news/2015-10-fast-noninvasive-brain-computer-interface.html

[13]. https://pbcquoc.github.io/cnn/

Hanh Phuc Nong, et. al. "Building an EEG-based BCI systems for Remote Device Control using CNN network."*American Journal of Engineering Research (AJER)*, vol. 10(6), 2021, pp. 123-130.