

Article

Hand Movement Activity-Based Character Input System on a Virtual Keyboard

Md Abdur Rahim  and Jungpil Shin * 

School of Computer Science and Engineering, The University of Aizu, Aizuwakamatsu, Fukushima 965-8580, Japan; rahim_bds@yahoo.com

* Correspondence: jpshin@u-aizu.ac.jp

Received: 17 April 2020; Accepted: 2 May 2020; Published: 8 May 2020



Abstract: Nowadays, gesture-based technology is revolutionizing the world and lifestyles, and the users are comfortable and care about their needs, for example, in communication, information security, the convenience of day-to-day operations and so forth. In this case, hand movement information provides an alternative way for users to interact with people, machines or robots. Therefore, this paper presents a character input system using a virtual keyboard based on the analysis of hand movements. We analyzed the signals of the accelerometer, gyroscope, and electromyography (EMG) for movement activity. We explored potential features of removing noise from input signals through the wavelet denoising technique. The envelope spectrum is used for the analysis of the accelerometer and gyroscope and cepstrum for the EMG signal. Furthermore, the support vector machine (SVM) is used to train and detect the signal to perform character input. In order to validate the proposed model, signal information is obtained from predefined gestures, that is, “double-tap”, “hold-fist”, “wave-left”, “wave-right” and “spread-finger” of different respondents for different input actions such as “input a character”, “change character”, “delete a character”, “line break”, “space character”. The experimental results show the superiority of hand gesture recognition and accuracy of character input compared to state-of-the-art systems.

Keywords: human-computer interaction; virtual keyboard; hand gesture; Myo Armband

1. Introduction

Human-computer interaction (HCI) is an ever-evolving advancement in the development of technology as a new method of communication between people and computers in the modern world [1,2]. Several new assistive methods, such as virtual reality [3], sign language recognition [4,5], speech recognition [6], visual analysis [7], brain activity [8], touch-free writing [9], have emerged in recent years to achieve this goal. Hand gesture recognition implies the importance of performing various visual tasks and working in an unobtrusive environment. However, the ability to control hand movement action provides a convenient and natural interaction mechanism for HCI. In particular, hand gesture-based character writing is an important aspect of the HCI character input system, which allows users to provide rich interaction commands using different movements of hand gestures. Recently, virtual keyboard-based character writing has been widely adopted to realize the non-touch input system [10,11]. As technology advances, these types of innovations are contributing enormously to everyday tasks and making life easier. This technology includes touch and non-touch devices that help users secure their personal and institutional information, operate in risky environments—even instruct robots—and ensure a healthy environment. The invention and use of virtual keyboards creates a new dimension for the character input system. The user can use it in any environment through the camera and image processing methods. In most cases, hand gestures are considered for controlling the virtual keyboard. A text input system using hand gestures is presented in Reference [12]. The author

suggested performing 260 different words without repeating for about 30 min. They achieved an average accuracy of 70%. In Reference [13], the authors developed a template pattern-based handwriting character recognition system. There are 46 Japanese hiragana characters and 26 alphabets are used to identify. The overall average error rate is 12.3%. Also, joysticks are often used as input devices for remote activities [14]. However, the controls can be a bit of a concern as the directions of the joystick are limited and can be broken if extra force is used on it. In addition, sensor-based HGR technology is used to provide user authentication, sign language recognition, character input, computer vision and virtual reality and so on. However, there has been a high demand for character input systems for the protection of confidential information and even for actions in a healthy environment.

Therefore, in this paper, we propose HGR techniques based on wearable devices, such as the Myo Armband, to provide a character input using a virtual keyboard. Keyboards and mice are often used as HCI devices. However, it is unsafe and also unhealthy due to dirt, as it is used for a long time. Many researchers have suggested the HCI using computer vision, voice recognition, bio-signals such as electroencephalogram (EEG), electromyography (EMG) [15], electrooculography (EOG) [16]. However, noise at the voice interface and the speed of processing of vision-based HCIs are major concerns. The EMG-based approach is of significant use in advancing and understanding HCI by sensing users’ hand movements. Moreover, EOG-based eye-tracking devices are used as input devices for communication, especially for people with amyotrophic lateral sclerosis. In this case, an expensive and stationary device is required to record the bio signal which is concerned with the performance of the recognition accuracy. However, this system is capable of distinguishing finger spread configurations, hold fist, wave left and right, and wrist movements. Using these types of signals, the activity of a user’s muscles and how they generate energy have been determined. The active areas of the nozzle make a huge difference when performing different gestures. Thus, the speed can be characterized by EMG. In this study, we used a Myo Armband device that includes accelerometer, gyroscope and EMG sensor. We acquired and analyzed different gestures and movements to perform character input using these sensors. This system provides an efficient technique for providing reliable performance for character input without touching any device or screen, which facilitates HCI and allows the user to operate the system in a healthy and secure way.

The paper outlines the following—in Section 2, we explain the proposed method for the character writing system, where signal preprocessing, feature extraction, and classification processes are described. Section 3 describes the system configuration and provides an overview of the virtual keyboard. Section 4 explains the experimental results and describes the findings with figures and tables. We discuss a brief review of the results in Section 5. Finally, we summarize the research in Section 6.

2. The Proposed System

The proposed character writing system has four main processes, namely, hand movement activity, preprocessing, feature extraction and recognition of actions. The system provides a virtual keyboard where the user can perform gesture actions and produce character input. Figure 1 shows the basic structure of the gesture-based character input system. Hence the wearable devices like Myo Armband which has three sensors (accelerometer, gyroscope, and EMG) are used in the proposed system for collecting the input signal.

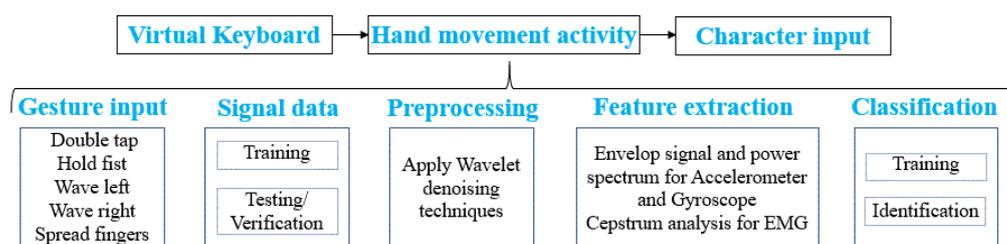


Figure 1. Basic structure of a character input system based on gesture movement.

2.1. Preprocessing

Due to the handheld features of the Myo Armband, reconstructing the input signal from the noise requires preprocessing to extract effective features. We used the wavelet denoising technique, which performs correlation analysis; therefore the output is expected to be the highest in the wavelet of mothers corresponding to most descriptions of the input signal. The wavelet techniques operate on a finite-length signal with the following processes: to get the noise data, to apply the waveform transform, and to obtain the denoised signal. By following Equations (1)–(3), we obtained the denoised signal to level 4 using *wdenoise*, where the observations of x with noise is $y(t)$, the input signal $x(t)$, noisy signal $n(t)$ and the standard deviation is σ . Figure 2 shows the steps of the preprocessing of the input signal.

$$y(t) = x(t) + \sigma n(t) \quad (1)$$

$$y(t) \xrightarrow{\text{wavelet_transform}} w(t) \quad (2)$$

$$y(t) \xrightarrow{\text{inverse_wavelet_transform}} \hat{x}(t). \quad (3)$$

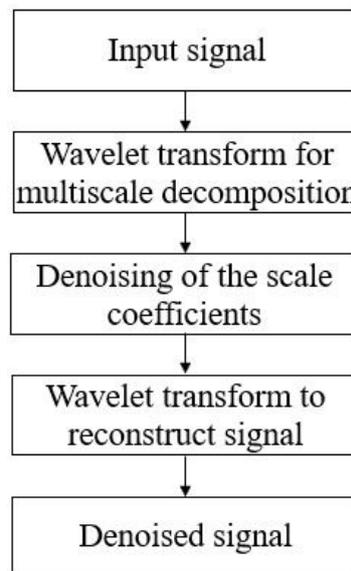


Figure 2. Steps for signal preprocessing.

In order to decompose the EMG signal and achieve the optimal reconstructed signal, the family of Daubechies wavelets is considered for this work. However, we selected db4 to extract the high and low-frequency detail from the input signal without losing data. The decomposition levels were selected on the basis of the effective frequency of the signal and the effectiveness of the features obtained from the components of the individual wavelet. In this case, thresholds are important factors for extracting meaningful information using the technique of wavelet denoising [17,18]. Therefore, we used the SURE threshold, an adaptive threshold technique to determine the threshold limits of each level. When the threshold coefficients are extracted from each level, the noise effect of the input signal is removed. At each level, reconstructed signals are obtained using the inverse wavelet transform. Figure 3 depicts the process of generating reconstructed signals, where cAs and cDs are the coefficients of decomposition approximation and detail.

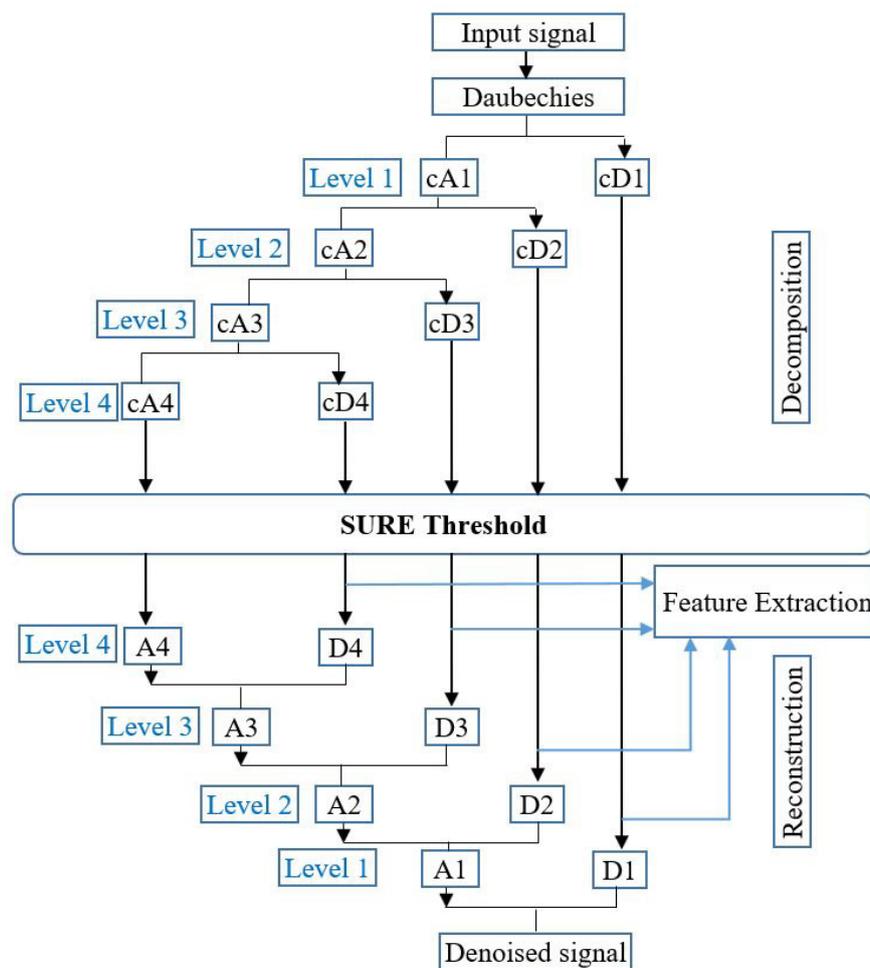


Figure 3. Process of wavelet analysis.

2.2. Feature Extraction

Each hand movement has a standard pattern for character writing. Depending on the pattern, the features are constantly extracted and fed to the action classifier. This pattern can be used as a feature, or to extract hand movements or gestures that are interpretable features for performing character input. Frequency domain features [19] like center frequency (CF), root mean square (RMS) frequency, root variance frequency (RMF), and slope sign change (SSC) can be used. Table 1 represents the descriptions of frequency-domain features. These features were extracted from the accelerometer and gyroscope signals using the Envelope Power Spectrum (EPS), and cepstrum was analyzed to extract the features from EMG sensor data.

Table 1. Frequency domain features of accelerometer and gyroscope signal.

Features	Descriptions
Center frequency	$CF = \frac{1}{N} \sum_{i=1}^N f_i$
RMS Frequency	$RMS = \sqrt{\left(\frac{1}{N} \sum_{i=1}^N f_i^2\right)}$
Root variance frequency	$RVF = \sqrt{\left(\frac{1}{N} \sum_{i=1}^N f_i^2 - CF^2\right)}$
Slope sign change	$S = \sum_{k=2}^{N-1} (s(k) - (s(k-1))) \times (s(k) - s(k+1)) $

2.2.1. Feature Extraction of Accelerometer and Gyroscope Sensors

The EPS is a process whereby the frequency amplitude curve is derived from the Fourier amplitude spectrum [20]. It accurately describes the point of interest in a window in time. In this work, the EPS

is used to extract the feature for the accelerometer and gyroscope signal. To do this, the DC bias is removed from the signal. Therefore, it is processed by a finite impulse response (FIR) in which the filter is the bandpass of order 10. The EPS is combined with the Hilbert transformed signal to generate the processed and analytical signal, which is defined in Equations (4) and (5).

$$\hat{x}(t) = \frac{1}{n} \int_{-\infty}^{\infty} \frac{x(\tau)}{t - \tau} d\tau \quad (4)$$

$$s(t) = x(t) + j.\hat{x}(t), \text{ with } j = \sqrt{-1}, \quad (5)$$

where t and τ is the time and translation parameter. The envelope signal is the magnitude of the analytic signal $s(t)$, which can be determined by Equation (6) as follows.

$$x_{env}(t) = |s(t)| = \sqrt{x(t)^2 + \hat{x}(t)^2}. \quad (6)$$

2.2.2. Feature Extraction of EMG Sensors

The Myo is used to acquire the EMG signal which includes eight built-in EMG sensors, $E_i = emg(k) | k = 1, 2, \dots, 8$. The EMG signals of each channel are obtained depending on the actions of the hand movement. In this work, we used the first six sensors to extract the feature of EMG. Cepstrum analysis was applied based on the spectral representation of the signal. However, the cepstrum method used for power spectrum analysis where the signal is seen as convolution to the source signal with a filter, therefore, we separated the source spectrum from the filter transfer function. Therefore, to achieve energy per unit of time by using power spectral analysis, $PDFT_i$. In the frequency domain, this convolution is multiplied by the corresponding Fourier transforms and used to measure the frequency spectrum, DFT_i using Equation (7). This frequency spectrum is then used to obtain the power spectrum and its complex conjugate using Equation (8).

$$DFT_i = FFT(E_i) \quad (7)$$

$$PDFT_i = DFT_i + \overline{DFT_i}. \quad (8)$$

To obtain the cepstrum coefficient C_i , we applied a Fourier transform to the logarithm of the power spectrum, therefore, we obtain the signal cepstrum by following Equation (9).

$$C_i = FFT(\log |PDFT_i|). \quad (9)$$

2.3. Classification Using SVM

In this work, we used support vector machines (SVMs) that implement one-versus-rest (OVR) multiclass techniques. An SVM creates a set of hyper-planes or hyper-planes in higher or infinite-dimensional space, which can be used for classification, regression or other tasks [21,22]. Since there is a non-linear relationship between the data sets, therefore, we used the different SVM kernel functions (KF) such as linear KF, quadratic KF, Cubic KF, fine Gaussian KF, and radial basis function (RBF). However, the kernel scale can be determined as shown in Table 2.

For training the support-vectors, low-order N-dimensional featured vectors are used, along with the statistical features of the frequency domain and the cepstrum coefficient. The feature vector has 24 features from the accelerometer and gyroscope signals and six (6) from the EMG signals. The feature vector $f_N (N = 1, 2, \dots, 30)$ obtained from Equation (10).

$$f_N = (cc_i, a_{xm}, a_{ym}, a_{zm}, g_{xn}, g_{yn}, g_{zn}), \quad (10)$$

where a_x, a_y, a_z and g_x, g_y, g_z presents the feature of accelerometer and gyroscope sensor, respectively, cc_i is cepstrum coefficient, $i = 1, 2, \dots, 6$, $m = 1, 2, 3, 4$ and $n = 1, 2, 3, 4$.

Table 2. Kernel Scale of support vector machine (SVM) models.

SVM Kernel Function	Kernel Scale
Linear KF	Auto
Quadratic KF	Auto
Cubic KF	Auto
Fine Gaussian KF	0.79
Mediaum Gaussian KF	3.2
Coarse Gaussian KF	1.3

3. System Configuration and Virtual Keyboard

In this study, we used an iPad Air 2 (iOS 10.1.1) and a Myo Armband, and the environment was developed with Swift 3.0, Objective C, and XCode 8.3.2, respectively. Users had to register their fingertips in a rectangular frame of the screen, for example, shown in Figure 4. Therefore, the system displays a virtual keyboard for producing character. We applied the template-matching method for detecting the user’s fingertip. To consider the brightness, the input converts the RGB color image to the HSV color space. The coordinate of the finger is defined as R given as the following Equations (11)–(13).

$$T'(x', y') = T(x', y') - \frac{1}{w.h} \cdot \sum_{x'', y''} T(x'', y'') \tag{11}$$

$$T'(x + x', y + y') = I'(x + x', y + y') - \frac{1}{w.h} \cdot \sum_{x'', y''} I(x + x', y + y') \tag{12}$$

$$R(x, y) = \sum_{x', y'} (T'(x', y') I(x + x', y + y')), \tag{13}$$

where I and T defines captured and finger data of the image, W and h is the width and height of the image. $x' = 0 \dots w - 1$, $y' = 0 \dots h - 1$.

After registering, this system displays a virtual keyboard on the display screen. The example of the virtual keyboard is shown in Figure 5. Therefore, the user can input characters by performing different gesture functions on different character blocks of the virtual keyboard. Table 3 presents the description of different hand gesture functions. A user input the selected character by double-tapping on the index and thumb fingers. The uppercase or lowercase letters are produced by identifying the hold fist. The user can delete a character by performing the wave left gesture and a line break is entered by performing the wave right gesture. Also, a space character is produced by detecting the spread of fingers. Figure 6 shows the structure of the environmental setup. The user is 30 cm away from the iPad and when entering a character, he or she releases the finger from the virtual keyboard.

Table 3. Description of hand gesture functions.

Gestures	Description	Functions
Double-tap	Twice tapping the index and thumb fingers	Input a character
Hold-fist	Fist all the fingers of the right hand together	Change character
Wave-left	Move all the fingers together to the left	Delete a character
Wave-right	Move all the fingers together to the right	Line break
Spread-fingers	Move all the fingers of the hand away from each other	Space character

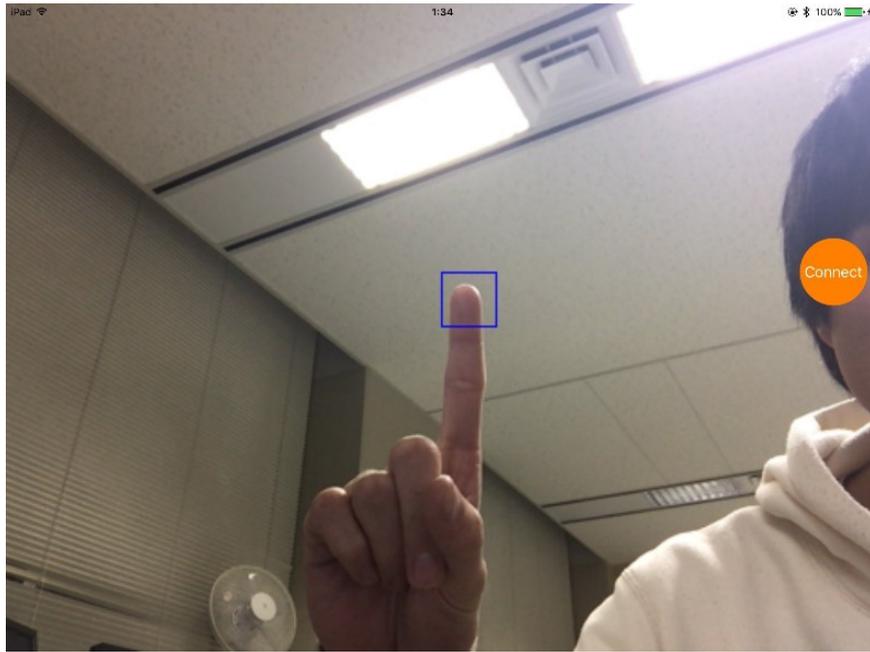


Figure 4. Region of interest location of the registration process.

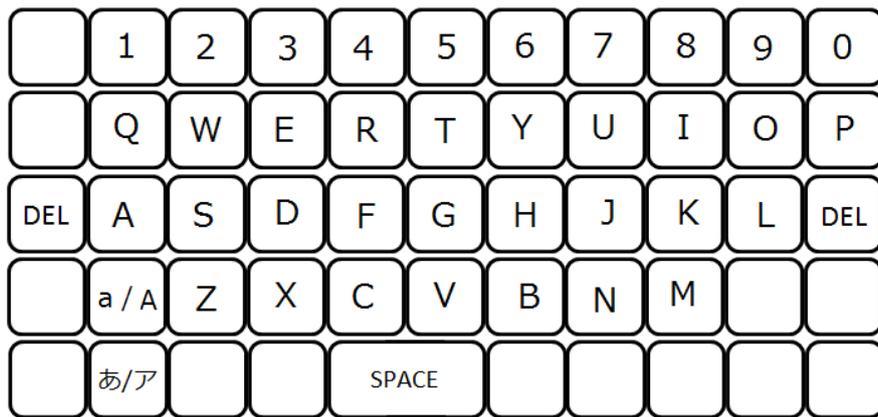


Figure 5. Example of a virtual keyboard of this system.

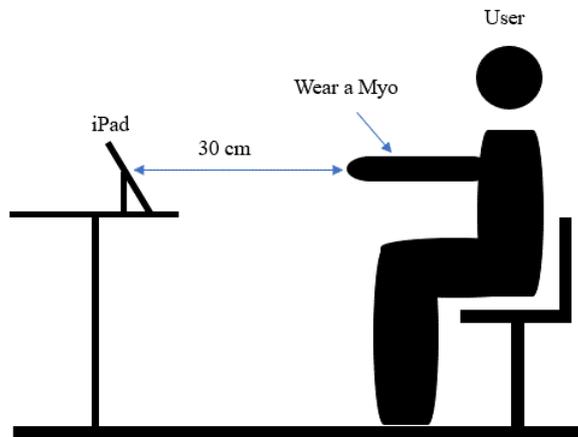


Figure 6. Experimental setup.

4. Experimental Result Analysis

4.1. Data Collection

The Myo Armband device has a three-axis acceleration and a gyroscope that samples 50 Hz data at 8 bits and EMG sensor data at 8 bits giving 200 Hz. It has 8 PODs that use elastic material that fits for almost everyone’s hands, which can stretch from 19 cm to 34 cm. We have neglected the last two pods which have very little information related to this work. However, for these experiments, 3 axis accelerometers and gyroscopes and 6 EMG pod sensor data were collected. Twenty participants were asked to perform hand gesture actions such as double-tap, hold-fist, wave-left and wave-right and spread-fingers. The example of hand gesture functions is shown in Figure 7. Each function is performed approximately 7 times in 10 s and a total of 700 sample data were collected. The acquired data are transmitted over the computer via Bluetooth and stored in a database. Figure 8 shows examples of database signals.

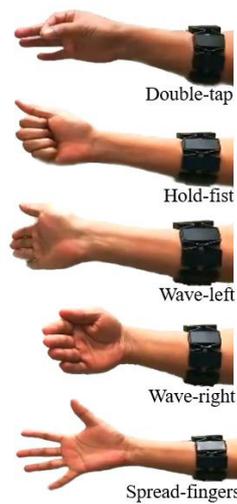


Figure 7. Example of hand movements.

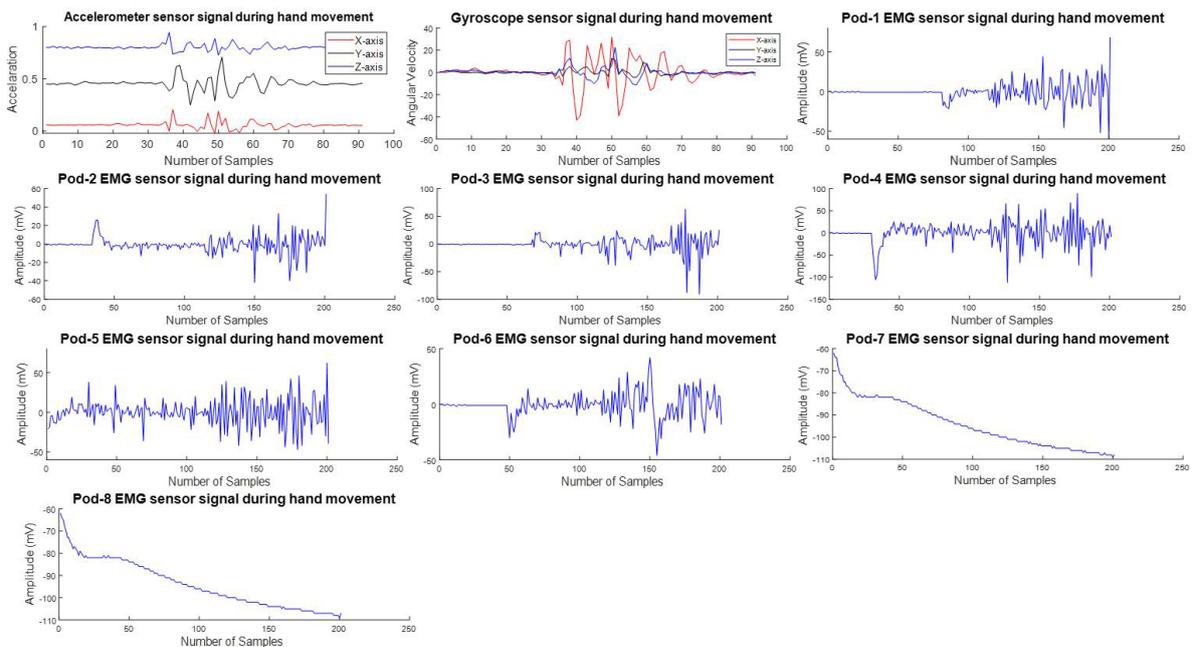


Figure 8. Example of accelerometer, gyroscope and electromyography (EMG) POD signal of double-tap gestures.

4.2. Signal Preprocessing and Feature Extraction

The raw EMG signal can be corrupted by sensor noise and unwanted peak. To eliminate the impact caused by these factors, we used a cut-off frequency of 0.1–8 Hz. Subsequently, a 10-sequence FIR bandpass filter was applied to process the raw EMG signal to enable efficient character input of this system. Figure 9 shows the denoised signals of accelerometer, gyroscope, and EMG sensor signals. Then, the denoised signals are used to extract the effective features of the hand movement. However, the envelope analysis is applied to the accelerometer and gyroscope signal, and the cepstrum coefficient analysis is applied to extract features from the EMG signal of a hand movement. The envelope signal and envelope spectrum were extracted from the envelope analysis process shown in Figure 10 and Figure 11, respectively. Subsequently, four types of frequency domain features were extracted from the envelope spectrum. In that case, these features were taken from the three-axis accelerometer and the gyroscope signal. Besides, six cepstrum coefficients were collected from six signals of the EMG sensor. Figure 12 illustrates the cepstrum coefficient of the EMG signal. The total size of the feature vector is 30 that includes the frequency-domain features of the accelerometer and gyroscope signal, and the cepstrum coefficient for each of the six EMG signals. Next, these extracted features have been applied to the training and identification process using SVM.

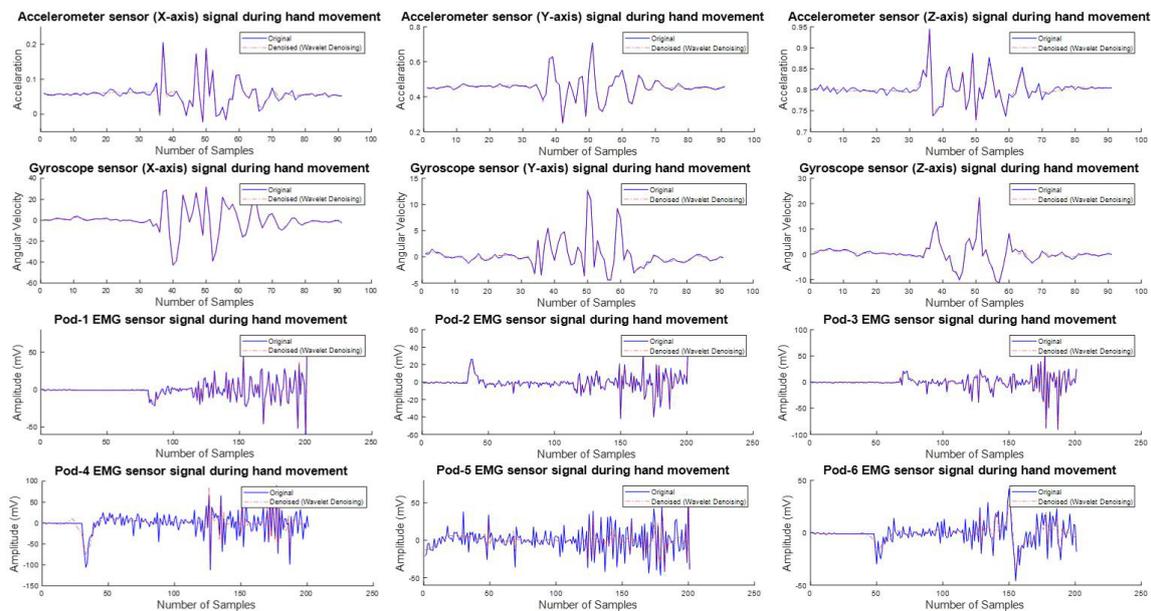


Figure 9. Example of the denoised signal of double-tap gestures.

4.3. Experimental Results and Performance Analysis

In this experiment, the results were calculated based on data analysis of 20 participants (18 Male, 2 Female). On average, the age was 26.4 ± 4.44 (average \pm standard deviation, range 18–35) years old. All participants were right-handed and used their right hand during the experiment. For the SVM training and testing, we divided our dataset into 2 groups, the training dataset, and the test dataset. We divided the first 10 participants into the training group and the rest to the test group. The extracted train feature vector is used in multiclass SVM training for future detection. Moreover, the test feature is used to validate the identification process and model. For this purpose, K-fold cross-validation ($K = 2$) was used to evaluate the proposed model. Table 4 represents the classification accuracy of the different SVM kernel functions. We compared this experiment with gesture action recognition and character input. The movement activity of hand to enter a character using the double-tap of the index and thumb fingers. Hold-fists are detected by changing the character. The delete function occurs by performing the wave-left activity of the hand and the line break is performed by the wave-right activity.

To enter the space between the character or the word, the spread-finger of the hand is recognized. The recognition accuracy of different gesture actions is shown in Table 5.

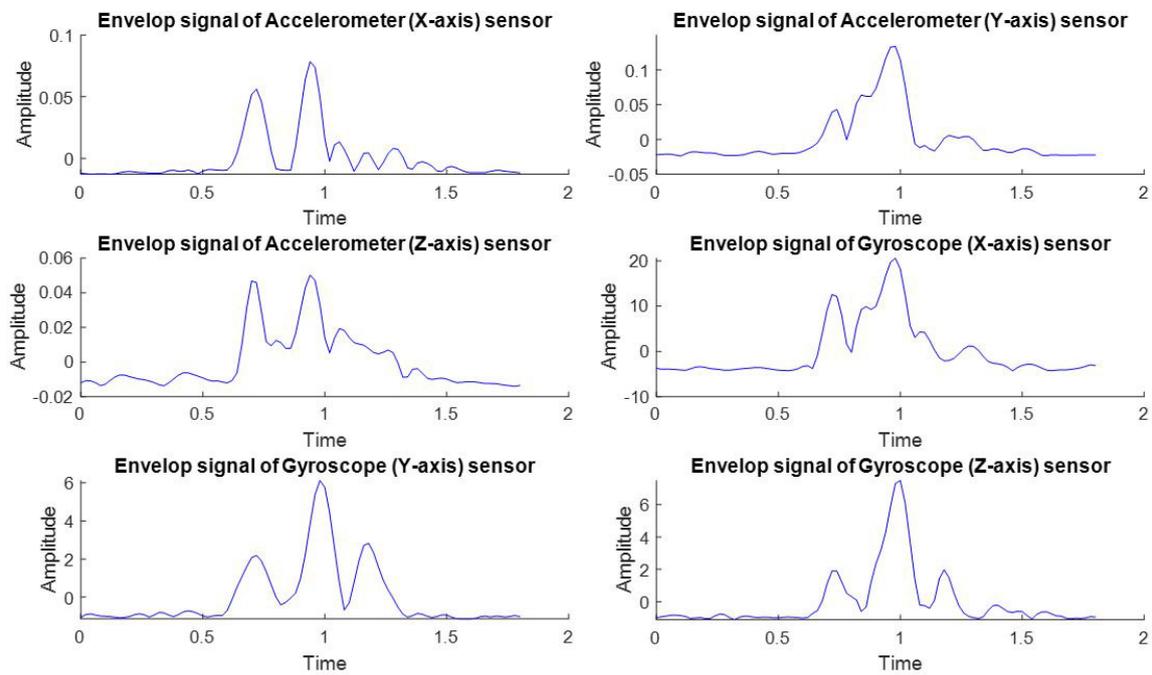


Figure 10. Examples of envelop signal of accelerometer and gyroscope sensor.

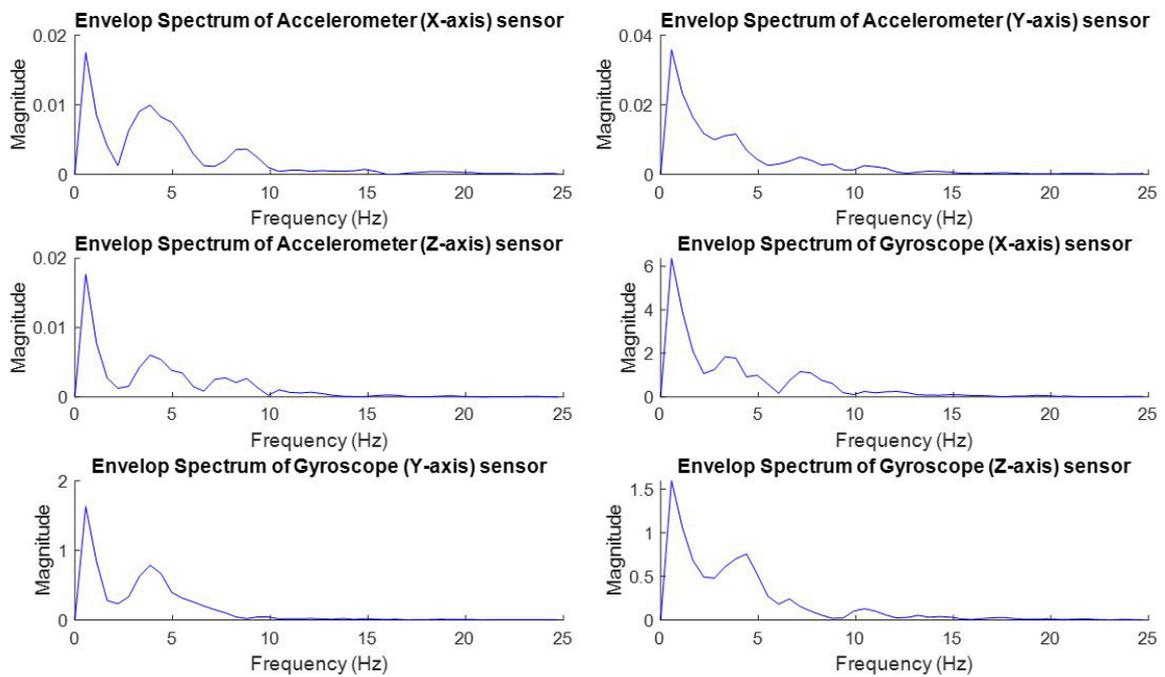


Figure 11. Example of envelope spectrum of accelerometer and gyroscope sensor.

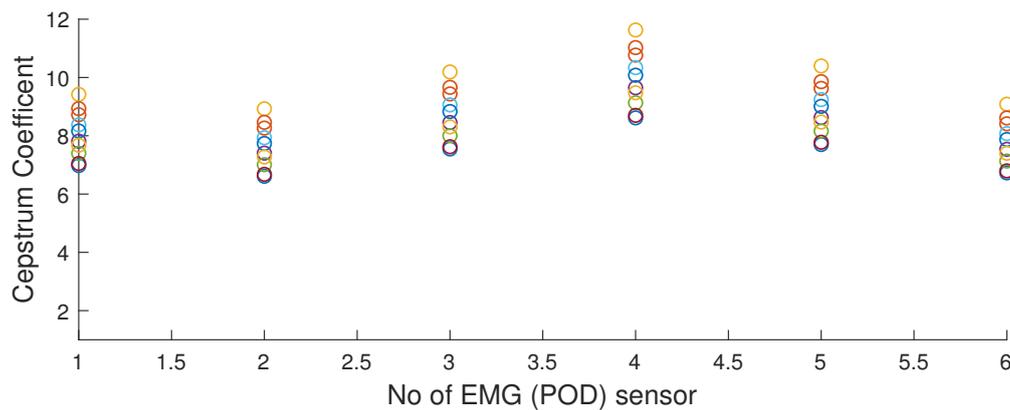


Figure 12. Cepstrum analysis of EMG six sensors.

Table 4. Classification of different SVM kernel functions.

SVM Kernels	Average Classification Accuracy (%)
Linear	95.92
Quadratic	97.62
Cubic	96.56
Fine Gaussian	97.10
Medium Gaussian	97.16
Coarse Gaussian	95.10
RBF	97.50

Table 5. Classification accuracy of different gesture functions.

Gestures	Average Accuracy (%)
Double-tap	99
Hold-fist	99.3
Wave-left	97.68
Wave-right	98.02
Spread-fingers	98.56

Furthermore, twenty random users were recruited to enter the characters from A to Z a total of five times to determine character input performance. Figure 13 presents an example of user performances. In this case, users need to perform a registered gesture to activate the virtual keyboard. Therefore, the system activates the virtual keyboard by displaying a blue dot in the area of the rectangle. However, the system only allows the user's dominant hand to input characters on the keyboard. Figure 14 depicts the average recognition accuracy of the character input of all users. The average accuracy is 96.57% with the highest accuracy being 100% and the lowest being 93.85%. Figure 15 illustrates the character input performances in time. Performance for the fifth time is much better than the first to fourth and the average input speed of a character is 1.32 s for all users.

Moreover, we compared the accuracy of our system classification with other Myo sensor-based systems. The proposed model achieves better performance than the state-of-the-others system. Table 6 shows the comparison of classification accuracy. Table 7 presents the comparison of the character input performance with state-of-the-art systems.

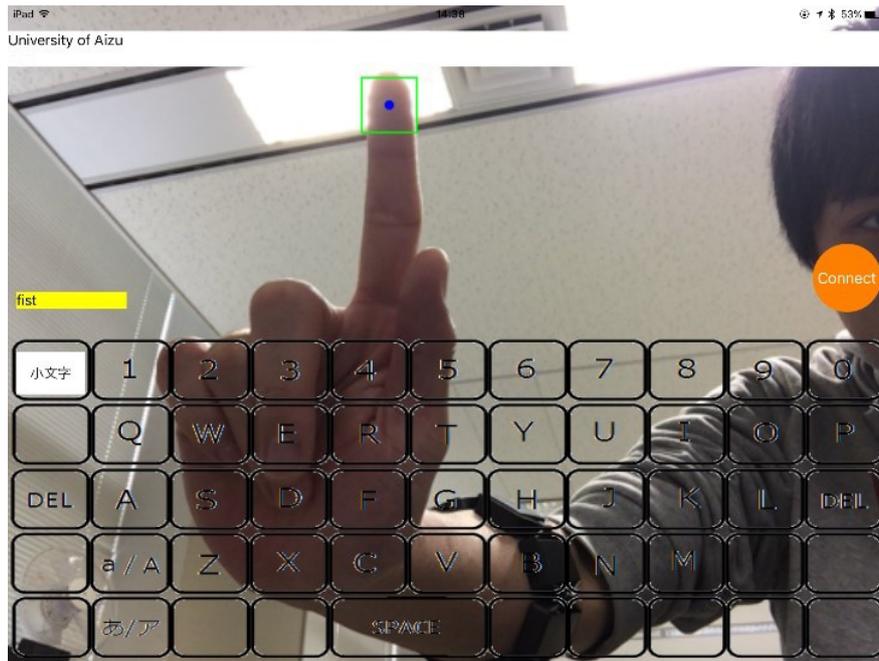


Figure 13. Simulation of a user’s performance.

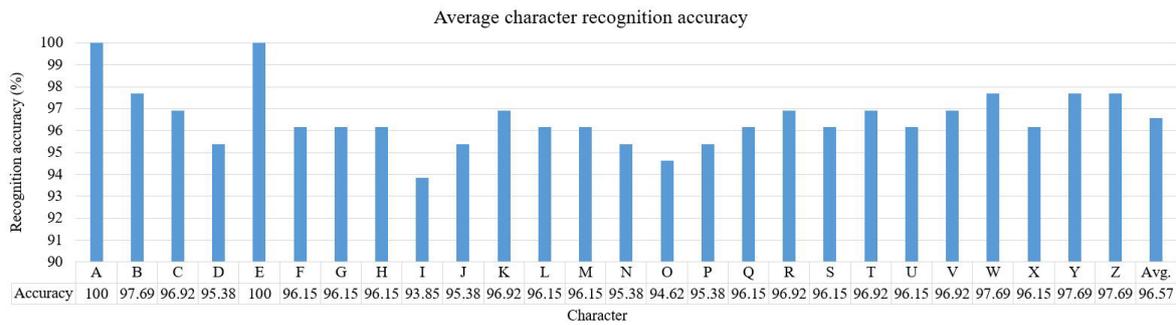


Figure 14. Average character recognition accuracy of all users.

Table 6. Comparison of classification accuracy.

Method	Classification Accuracy (%)
Ref. [12]	70
Ref. [13]	86.7
Proposed	96.57

Table 7. Performance comparison of character inputs with state-of-the-art systems.

Reference	Perspective	Average Time	Average Error Rate
Ref. [12]	MYO-Typer	2.56 words/min	10.75%
Ref. [13]	Character writing in Air	72 characters/12.168 s	12.3%
Proposed	Virtual keyboard-based character writing	45 characters/min	3.43%

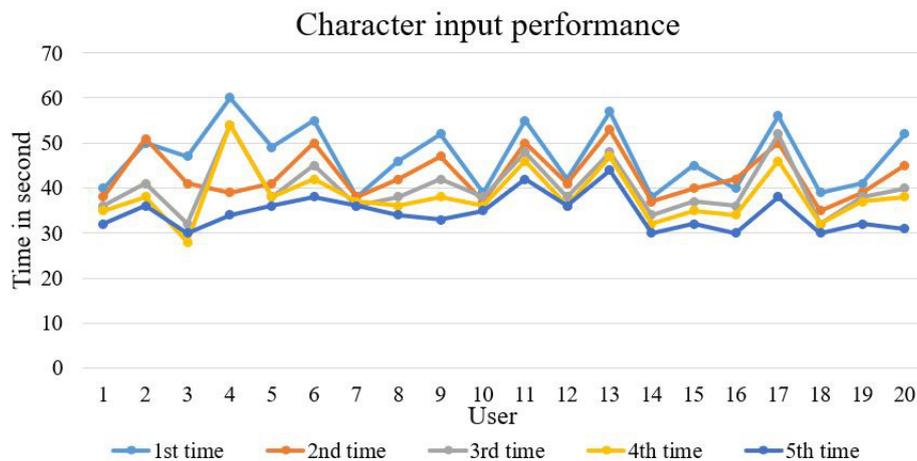


Figure 15. Character input performances of all users.

5. Discussion

We introduced a character input system that helps people input characters with hand activity. The proposed virtual keyboard and a hand gesture set are designed for character input. From the experimental results, we observed that the average accuracy of the character input is 96.57% which is depicted in Figure 14. However, the highest accuracy among the character input was 100% and the minimum accuracy was 93.85%. When the user selects a key near to the character blocks, the character input may be selected incorrectly. This may cause an incorrect character selection which affects the average recognition accuracy. Using a deep image processing technique can be a possible solution. Also, we have recognized the effectiveness of various gestures. The average accuracy is shown in Table 5. The highest accuracy is 99.3% of the hold-fist gestures and 98.02% of the lowest wave-right gestures. As compared to the character input accuracy, our system achieves better than the state-of-the-art method [12,13]. Based on the character input performance shown in Figure 15, the average character input of the proposed system is 45 characters per minute. Moreover, the average error rate shown in Table 7 is 3.43%. Our system achieved better performance compared to the error rates of 10.75% and 12.3%, respectively, as described in References [12,13]. This result indicates that the user can easily perform character input using the proposed system with the minimum error rates.

6. Conclusions

This work describes the character writing process on a virtual keyboard using hand movement activity. For collecting and processing activity signals, we considered a sensor-based device such as the Myo Armband that has an accelerometer, gyroscope and EMG sensor. In the proposed method, the input signal is pre-processed to remove the noise and these signals are fed to the feature extraction. We extracted the feature from the accelerometer and gyroscope using the envelope spectrum and extracted the feature from the EMG using cepstrum analysis and train SVMs. In the classification, the features are taken from the test data and used in the SVM classification. We calculated the average classification accuracy to validate the proposed method where the quadratic and RBF kernel functions are displayed at 97.62% and 97.5%, respectively. The experimental results show that the average character recognition accuracy is 96.57%, which is an increase in performance compared to state-of-the-art methods.

In advanced studies, it will be interesting to look at other gesture activities, including deleting multiple character at one time, copying, pasting, and so on, which may support greater functionality and direction for advanced character input systems.

Author Contributions: Conceptualization, M.A.R. and J.S.; methodology, M.A.R.; software, M.A.R.; validation, M.A.R. and J.S.; formal analysis, M.A.R. and J.S.; investigation, M.A.R.; resources, J.S.; data curation, M.A.R.;

writing—original draft preparation, M.A.R.; writing—review and editing, M.A.R. and J.S.; visualization, J.S.; supervision, J.S.; project administration, J.S.; funding acquisition, J.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

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

References

1. Mencarini, E.; Rapp, A.; Tirabeni, L.; Zancanaro, M. Designing Wearable Systems for Sports: A Review of Trends and Opportunities in Human–Computer Interaction. *IEEE Trans. Hum. Mach. Syst.* **2019**, *49*, 314–325. [[CrossRef](#)]
2. Esposito, A.; Esposito, A.M.; Vogel, C. Needs and challenges in human computer interaction for processing social emotional information. *Pattern Recognit. Lett.* **2015**, *66*, 41–51. [[CrossRef](#)]
3. Sherman, W.R.; Craig, A.B. *Understanding Virtual Reality: Interface, Application, and Design*; Morgan Kaufmann: Burlington, MA, USA, 2018.
4. Rahim, M.A.; Islam, M.R.; Shin, J. Non-Touch Sign Word Recognition Based on Dynamic Hand Gesture Using Hybrid Segmentation and CNN Feature Fusion. *Appl. Sci.* **2019**, *9*, 3790. [[CrossRef](#)]
5. Yang, H.D. Sign language recognition with the Kinect sensor based on conditional random fields. *Sensors* **2015**, *15*, 135–147. [[CrossRef](#)] [[PubMed](#)]
6. Ramakrishnan, S.; El Emary, I.M. Speech emotion recognition approaches in human computer interaction. *Telecommun. Syst.* **2013**, *52*, 1467–1478. [[CrossRef](#)]
7. Rautaray, S.S.; Agrawal, A. Vision based hand gesture recognition for human computer interaction: A survey. *Artif. Intell. Rev.* **2015**, *43*, 1–54. [[CrossRef](#)]
8. Corsi, M.C.; Chavez, M.; Schwartz, D.; Hugueville, L.; Khambhati, A.N.; Bassett, D.S.; De Vico Fallani, F. Integrating eeg and meg signals to improve motor imagery classification in brain–computer interface. *Int. J. Neural Syst.* **2019**, *29*, 1850014. [[CrossRef](#)] [[PubMed](#)]
9. Rahim, M.A.; Shin, J.; Islam, M.R. Gestural flick input-based non-touch interface for character input. *Vis. Comput.* **2019**, 1–19. [[CrossRef](#)]
10. Kim, J.O.; Kim, M.; Yoo, K.H. Real-time hand gesture-based interaction with objects in 3D virtual environments. *Int. J. Multimed. Ubiquitous Eng.* **2013**, *8*, 339–348. [[CrossRef](#)]
11. Rusydi, M.I.; Azhar, W.; Oluwarotimi, S.W.; Rusydi, F. Towards hand gesture-based control of virtual keyboards for effective communication. *IOP Conf. Ser. Mater. Sci. Eng.* **2019**, *602*, 012030. [[CrossRef](#)]
12. Wang, F.; Cui, S.; Yuan, S.; Fan, J.; Sun, W.; Tian, F. MyoTyper: A MYO-based Texting System for Forearm Amputees. In Proceedings of the Sixth International Symposium of Chinese CHI, Montreal, QC, Canada, 21–22 April 2018; pp. 144–147.
13. Tsuchida, K.; Miyao, H.; Maruyama, M. Handwritten character recognition in the air by using leap motion controller. In *International Conference on Human-Computer Interaction*; Springer: Berlin, Germany, 2015; pp. 534–538.
14. Scalera, L.; Seriani, S.; Gallina, P.; Di Luca, M.; Gasparetto, A. An experimental setup to test dual-joystick directional responses to vibrotactile stimuli. *IEEE Trans. Haptics* **2018**, *11*, 378–387. [[CrossRef](#)] [[PubMed](#)]
15. Zhang, Y.; Chen, Y.; Yu, H.; Yang, X.; Lu, W.; Liu, H. Wearing-independent hand gesture recognition method based on EMG armband. *Pers. Ubiquitous Comput.* **2018**, *22*, 511–524. [[CrossRef](#)]
16. Ding, X.; Lv, Z. Design and development of an EOG-based simplified Chinese eye-writing system. *Biomed. Signal Process. Control* **2020**, *57*, 101767. [[CrossRef](#)]
17. Schimmack, M.; Mercorelli, P. An on-line orthogonal wavelet denoising algorithm for high-resolution surface scans. *J. Franklin Inst.* **2018**, *355*, 9245–9270. [[CrossRef](#)]
18. Mercorelli, P. Biorthogonal wavelet trees in the classification of embedded signal classes for intelligent sensors using machine learning applications. *J. Franklin Inst.* **2007**, *344*, 813–829. [[CrossRef](#)]
19. Shin, J.; Islam, M.R.; Rahim, M.A.; Mun, H.J. Arm movement activity based user authentication in P2P systems. *Peer-to-Peer Netw. Appl.* **2019**, 1–12. [[CrossRef](#)]
20. Nguyen, H.; Kim, J.; Kim, J.M. Optimal sub-band analysis based on the envelope power Spectrum for effective fault detection in bearing under variable, low speeds. *Sensors* **2018**, *18*, 1389. [[CrossRef](#)] [[PubMed](#)]

21. Zhang, W.; Yu, L.; Yoshida, T.; Wang, Q. Feature weighted confidence to incorporate prior knowledge into support vector machines for classification. *Knowl. Inf. Syst.* **2019**, *58*, 371–397. [[CrossRef](#)]
22. Schölkopf, B.; Smola, A.J.; Bach, F. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*; MIT Press: Cambridge, MA, USA, 2002.



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).