electronics

MDPI

Article

2D Camera-Based Air-Writing Recognition Using Hand Pose Estimation and Hybrid Deep Learning Model

Taiki Watanabe [1], Md. Maniruzzaman [1], Md. Al Mehedi Hasan [2], Hyoun-Sup Lee [3], Si-Woong Jang [4] and Jungpil Shin [1,*]

[1] School of Computer Science and Engineering, The University of Aizu, Aizuwakamatsu 965-8580, Fukushima, Japan
[2] Department of Computer Science & Engineering, Rajshahi University of Engineering and Technology, Rajshahi 6204, Bangladesh
[3] Department of Applied Software Engineering, Dongeui University, Busanjin-Gu, Busan 47340, Republic of Korea
[4] Department of Computer Engineering, Dongeui University, Busanjin-Gu, Busan 47340, Republic of Korea
* Correspondence: jpshin@u-aizu.ac.jp

Abstract: Air-writing is a modern human–computer interaction technology that allows participants to write words or letters with finger or hand movements in free space in a simple and intuitive manner. Air-writing recognition is a particular case of gesture recognition in which gestures can be matched to write characters and digits in the air. Air-written characters show extensive variations depending on the various writing styles of participants and their speed of articulation, which presents quite a difficult task for effective character recognition. In order to solve these difficulties, this current work proposes an air-writing system using a web camera. The proposed system consists of two parts: alphabetic recognition and digit recognition. In order to assess our proposed system, two character datasets were used: an alphabetic dataset and a numeric dataset. We collected samples from 17 participants and asked each participant to write alphabetic characters (A to Z) and numeric digits (0 to 9) about 5–10 times. At the same time, we recorded the position of the fingertips using MediaPipe. As a result, we collected 3166 samples for the alphabetic dataset and 1212 samples for the digit dataset. First, we preprocessed the dataset and then created two datasets: image data and padding sequential data. The image data were fed into the convolution neural networks (CNN) model, whereas the sequential data were fed into bidirectional long short-term memory (BiLSTM). After that, we combined these two models and trained again with 5-fold cross-validation in order to increase the character recognition accuracy. In this work, this combined model is referred to as a hybrid deep learning model. Finally, the experimental results showed that our proposed system achieved an alphabet recognition accuracy of 99.3% and a digit recognition accuracy of 99.5%. We also validated our proposed system using another publicly available 6DMG dataset. Our proposed system provided better recognition accuracy compared to the existing system.

Keywords: air-writing; hand pose estimation; deep learning; character recognition

Citation: Watanabe, T.; Maniruzzaman, M.; Hasan, M.A.M.; Lee, H.-S.; Jang, S.-W.; Shin, J. 2D Camera-Based Air-Writing Recognition Using Hand Pose Estimation and Hybrid Deep Learning Model. *Electronics* **2023**, *12*, 995. https://doi.org/10.3390/electronics12040995

Academic Editors: Juan M. Corchado, In Lee, Fuji Ren, Rashid Mehmood, Byung-Gyu Kim and Carlos A. Iglesias

Received: 13 January 2023
Revised: 8 February 2023
Accepted: 14 February 2023
Published: 16 February 2023

1. Introduction

Over the last few decades, we have become habituated to interacting with digital environments using touchscreens and other electronic devices for various purposes. In order to interact with the digital world, the next wave of technology is expected to eliminate the necessity of intermediary physical devices such as smartphones, which have the extra load in order to carry them with us and take them out of our pockets [1]. Virtual reality (VR) and augmented reality (AR), in which the output is frequently projected precisely into the user's eyes, appear to be leading the next era of such technology [2]. Speech recognition is one of the well-defined methods that has received great attention because it is considered

as a simple natural way of interacting. Unfortunately, speech recognition does not meet all of the requirements for interacting with technologies [1].

Recently, gesture recognition has also gained great attention and can also be used as a communication method [2]. Moreover, various traditional methods such as keyboards, touchpads, or other pushing and touching devices have been widely used for interacting. Many bacteria are said to exist on such kinds of traditional methods that are touched by an unspecified number of people. These are thought to be one of the infection routes for various diseases. In recent years, due to the influence of the coronavirus, these traditional methods have become a big problem in terms of hygiene. These traditional methods do not fit into VR, AR, and gesture-based technologies. As a result, various technologies such as acceleration sensors, cameras, electromagnetic, photosensors, and auditory signals are widely used in place of these traditional methods to create new forms of interaction [3]. In this situation, air-writing can be used in order to address the necessity of the touchless writing method. Air-writing is one of the most popular forms of writing. Writing words or letters with finger or hand movements in free space is called air-writing. The recognition of air-writing is a particular case of gesture recognition in which gestures can be matched to write digits and characters in the air [4]. In the case of gesture recognition, a wide range of interaction choices are made possible by air-writing movements without the user having to learn and recall them.

Air-writing character recognition is a challenging task [4]. Air-writing characters are different from gestures due to fine-grain movement and can be written in various ways by different people. Moreover, air-writing also differs from traditional paper-based writing [5] in that pen lift movements are far less obvious and there is no tactile or visual input, which makes it difficult for the user to maintain their spatial orientation. In this work, we hypothesized that air-writing characters could be correctly recognized by analyzing a character dataset obtained from a web camera. Previously, a lot of existing works have proposed systems to recognize air-writing characters. Most of the researchers used convolution neural networks (CNN), long short-term memory (LSTM), and so on in order to recognize the air-writing characters. In this work, we extracted two sets of features and developed a hybrid architecture in order to combine these two sets of features for air-writing character recognition. We built this hybrid architecture with the combination of CNN and bidirectional long short-term memory (BiLSTM) for air-writing character recognition. In this work, we defined this hybrid approach as a hybrid deep learning model. In order to assess this proposed system, we collected 3166 alphabetic samples and 1212 numeric or digit samples (each alphabetic character and numeric digit were air-written and recorded 5–10 times for each participant) from 17 participants. The experimental results illustrated that our proposed system could recognize air-writing characters with higher recognition accuracy than the state-of-the-art methods. More specifically, the proposed system achieved an alphabetic recognition accuracy of 99.3% and a digit recognition accuracy of 99.5%, which was comparatively higher than the existing system.

The rest of this paper is organized as follows: Section 2 represents related work. Section 3 presents the materials and methods, including more clear descriptions of the proposed air-writing system, the character datasets, and the character recognition system. The experimental settings and performance evaluation metrics are discussed in Section 4. Section 5 describes the experimental results and their discussion is presented in Section 6. Finally, the conclusions and future direction of this paper are summarized in Section 7.

2. Related Works

In this section, we discussed the existing works on character recognition. Various existing studies proposed air-writing systems using various devices (Kinect, Leap Motion, Wearable devices, etc.) and we summarized their findings, which are shown in Table 1. For example, Murata et al. [6] used a Kinect sensor for hand gestures and character recognition. They extracted features from the collected data and recognized characters and digits with DP matching. They obtained a recognition accuracy of 95.0% for numeric or digit and 98.9% for alphabetic characters. Amma et al. [7] proposed an air-writing recognition system using

the wearable sensor. They also collected 6500-character samples from 10 subjects (males: 5 and females: 5), whereas every subject wrote the alphabet characters 25 times. On the other hand, one subject was asked to write 652 English words in order to make another word dataset, which was used to validate their proposed system. They used a Hidden Markov Model (HMM) for character and word recognition. The HMM-based classier achieved recognition accuracy of 94.8% for characters and 97.5% for words. Hayakawa et al. [8] also proposed a Japanese Kanji character recognition system using a PS Move device and obtained a recognition accuracy of 98.2%.

Bastas et al. [9] proposed a system for the recognition of air-written characters. They engaged 10 subjects (8 males and 2 females) ranging in age from 23 to 50 years. Each subject wrote air-written digits (0 to 9) about 10 times using a Leap Motion device. As a result, the researchers obtained 1200 samples for analysis. They randomly divided the dataset into training, test, and validation sets. They selected 200 samples for the test set, 100 samples for the validation set, and the rest of the samples for the training set. They built an LSTM network and its BiLSTM, 1D CNNs, CNN-LSTM, Temporal Convolutional Network (TCN), and deep CNN architecture. They illustrated that the highest recognition accuracy rate of 99.5% was obtained by LSTM. Amma et al. [10] introduced a two-stage system for the detection and recognition of handwriting gestures. The first stage was the spotting stage and another was the recognition stage. In the spotting stage, support vector machine (SVM) was used to determine the segmentation of data, which contained handwriting. Whereas, the HMM was used for recognition. In order to evaluate their proposed system, they collected gesture samples from three subjects. They illustrated that their proposed system achieved a recall of 99.0%, a precision of 25.0%, a person-independent error rate of 11.0%, and a person-dependent error rate of 3.0%.

Arsalan and Santra [11] proposed an air-writing system based on a millimeter-wave radar network. They used markers to write letters in the air. They proposed LSTM, BiLSTM, and ConvLSTM models with connectionist temporal classification (CTC) loss functions for character recognition. Recognition targets were alphabet (A–J) and digits (1–5). The dataset was recorded with 100 samples for the training set and 25 samples for the testing set from random trials for each character. The highest character recognition accuracy (98.3%) was obtained by ConvLSTM-CTC. Yanay and Shmueli [2] developed a novel air-writing system using a smart band. Fifty-five people participated in the creation of the dataset, among them, 28 participants were female and the rest of the participants were male. The dominant hand of the participants was the right hand for 46 participants and the left for 9 of them. The recognition method changed depending on whether there were user data to be entered in the dataset. They used dynamic time warping (DTW) with k-nearest neighbors (k-NN) for user-dependent, whereas CNN was used for user-independent recognition. They obtained a user-dependent recognition accuracy of 89.2% and a user-independent recognition accuracy of 83.2%.

Sonpda and Muraoka [12] proposed an input method in a wearable computer environment using a wearable video camera. The proposed an air-writing system that captures the movement of the user's hand with a wearable video camera, and character input was performed by analyzing monochrome grayscale images. The dataset consisted of 5 subjects and 360 alphanumeric characters. They used a DP matching approach to recognize the letter and obtained a 75.3% recognition accuracy. Setiawan and Pulungan [13] proposed an air-writing system using Leap Motion. Deep belief networks (DBN) with resilient back-propagation (Rprop) fine-tuning were used to identify alphabets and digits. They used an alphabet dataset, which had 30,000 samples. About 22,000 samples were used for the training set and the rest of the samples were used for the test set. Another digit dataset was MNIST. The authors illustrated that DBN achieved a character recognition accuracy of 99.7% and a digit recognition accuracy of 96.3%. Chen et al. [14] used an ultrasonic transmitter. Motion tracking utilizes direction of arrival (DOA) information. An ultrasound receiver array tracks the motion of the wearable ultrasound transmitter by observing changes in the DOA of the signal. They proposed a new 2D DOA-based algorithm that could track changes in

transmitter orientation based on measured phase differences between receiver array elements. They used a method called order-restricted matching (ORM) and achieved an accuracy of 96.3%.

Saez-Mingorance et al. [15] introduced a novel air-writing system using one array of ultrasonic transceivers. The recognition targets were numbers (1–4) and letters (A–D). In order to recognize numbers and letters, they implemented LSTM, CNN, ConvAutoencoder, and convolutional LSTM classification methods. The dataset consisted of 27,670 samples. About 5539 samples were used for the test set and 22,131 were used for the training set. The highest recognition accuracy of 99.5% was obtained by ConvLSTM. Alam et al. [16] proposed a CNN-LSTM system for trajectory-based air-writing recognition networks. They used four datasets (RTD, RTC, smart band, and Abas) in order to check the efficiency of their proposed system. CNN-LSTM system achieved recognition accuracy of 99.6% for RTD, 98.7% for RTC, 95.6% for the smart band, and 99.9% for Abas, respectively. Alam et al. [5] introduced an air lighting recognition system using 3D trajectories collected by a fingertip-tracking depth camera. They used two datasets. One dataset was the RTD-based digit dataset, which had 21,000 samples. Another dataset was the 6D motion gesture (6DMG)-based alpha number dataset. They employed CNN and LSTM as recognition methods. They showed that classification accuracy for the RTD dataset was 99.2% for LSTM and 99.16% for CNN. The classification accuracy of the 6DMG dataset was 99.3% for LSTM and 99.3% for CNN. Chen et al. [4] performed character or word recognition based on 6-DOF hand motion data. They extracted 5720 motion characters and 5400 motion word samples from 22 participants (all right-handed, 17 males and 5 females). They implemented HMM for the recognition of characters and words and achieved a 0.8% recognition error for words and a 1.9% recognition error for letters/characters.

Table 1. Summary of the existing studies for air-writing recognition.

Authors	Device	Methods	Recognition Target	Accuracy (%)
Murata et al. [6]	Kinect	DP matching	Alphabet	98.9
			Digit	95.0
Amma et al. [7]	Wearable	HMM	Alphabet	94.8
			Digit	95.0
Hayakawa et al. [8]	PS Move	CNN	Japanese	98.2
Bastas et al. [9]	Leap Motion	LSTM	Digit	99.5
Amma et al. [10]	Wearable	HMM	Word	Recall: 99.0 Precision: 25.0
Arsalan and Santra [11]	60-GHzwave Radars	ConvLSTM-CT	Alphabet Digit	98.3
		DCNN		98.3
Yanay and Shmueli [2]	Smart-bands	DTW + KNN	Alphabet	98.2
Sonpda and Muraoka [12]	Video Camera Wearable Comput	DP Matching	Alphabet Digit	75.3
Setiawan and Pulungan [13]	Leap Motion	DBN	Alphabet	99.7
			Digit	96.3
Chen et al. [14]	Ultrasonic Transmitter	ORM	Alphabet	96.3

Table 1. *Cont.*

Authors	Device	Methods	Recognition Target	Accuracy (%)
Saez-Mingorance et al. [15]	Ultrasonic Transceivers	ConvLSTM	Digit Alphbet	99.5
Alam et al. [16]	Intel RealSence Camer	TARNet	Alphabet	98.7
			Digit	99.6
	Smart Band		Alphabet	95.6
	Leap Motion		Alphabet	99.9
Alam et al. [5]	Intel RealSence Camera	LSTM	Digit	99.2
		CNN	Digit	99.1
	Wii Remote Controller	LSTM	Alphabet	99.3
		CNN	Alphabet	99.3
Chen et al. [4]	Wii Remote Controller	HMM	Word Alphabet	99.2 98.1

3. Materials and Methods

3.1. Proposed Air-Writing System

In this work, we propose an air-writing system and its methodology is presented in Figure 1. First, the system performed character data collection. In data collection, it collected images from the web camera. Using these collected images and media pipe, we estimated the coordinates of the hand joints. The information to be estimated was 21 joints, and the joint coordinates had three-dimensional data (x, y, z). Next, the coordinates were collected by tracking the tip of the index finger. The next step was to perform data pre-processing. In a pre-processing step, we normalized the character dataset in order to make it independent of character size. The system used normalized data to create image data and sequential data. After that, the system performed feature extraction and model training. We propose a hybrid deep learning approach for character recognition by the combination of CNN and BiLSTM models. We first trained the CNN model for the image dataset. On the other hand, the sequential dataset was fed into BiLSTM and combined with these two models. In the test phase, the trained model was used for character recognition. We describe each step of this study in detail in the following subsections.

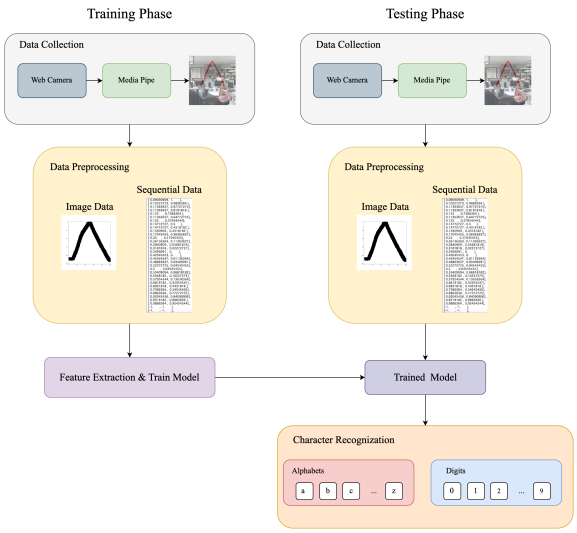
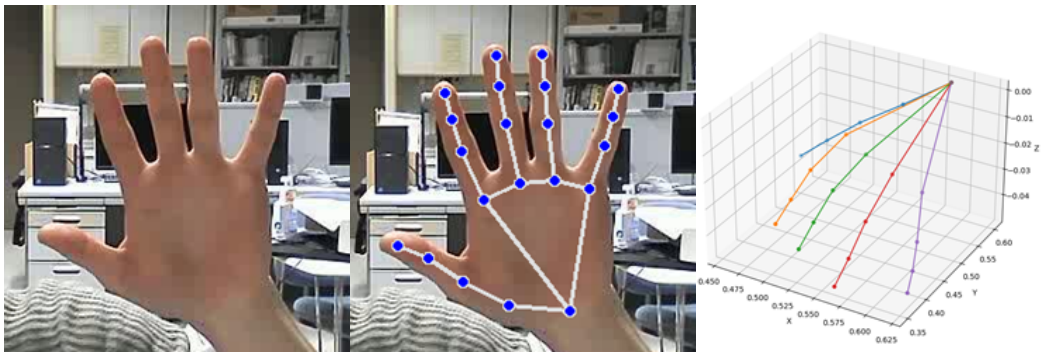


Figure 1. Proposed methodology of air-writing system.

3.2. Dataset

3.2.1. Hand Pose Estimation

Our proposed system used MediaPipe [17] for hand skeleton coordinate estimation. MediaPipe was an API for estimating joint coordinates developed by Google. A simple flow-up to coordinate estimation is shown in Figure 2. First, the palm was detected from the input image and it performed an accurate key point localization of the hand joint coordinates within the detected hand region. The estimated joint coordinates consisted of 21 joints. Each joint had 3-dimensional data (x, y, z). x and y were normalized to [0, 1] by the image width and height, respectively. z represented the depth with the depth of the wrist as the origin, and the smaller the value, the closer the joint was to the camera.



(**a**) (**b**) (**c**)

Figure 2. Example of MediaPipe: (**a**) original hand image; (**b**) estimated hand image; and (**c**) estimated hand joints.

3.2.2. Data Collection Procedure

In this work, 17 participants (Males: 16 and Females: 1) aged 19–23 years were engaged in order to create character data. All participants were right-handed and used their right hand. The following specific rules were followed during data collection: (i) the participants were asked to write with the tip of their index finger; (ii) the participants practiced before writing any letters; (iii) each letter was written 5–10 times; (iv) fingertip data were recorded using MediaPipe device. Participants were freely set the distance from the web camera and the size of the written character.

3.2.3. Graffiti Characters used for Character Recognition

In this work, we used Graffiti characters in order to propose a character recognition system. The study required a single-stroke shorthand handwriting recognition system used in personal digital assistants (PDA) and based on Palm OS. Graffiti was created by Palm Inc. as a recognition system for GEOS devices (HP OmniGo 120 and Magic Cap-line). The software was usually developed using uppercase characters that could be blindly drawn on a touch-sensitive display using a stylus. The most difficult Graffiti letters were “A”, “F”, “K”, and “T”, which could be drawn without any need to match a cross-stroke [6,18,19]. This character could be written with a single stroke, and we considered that it was compatible with air-writing (see Figure 3).

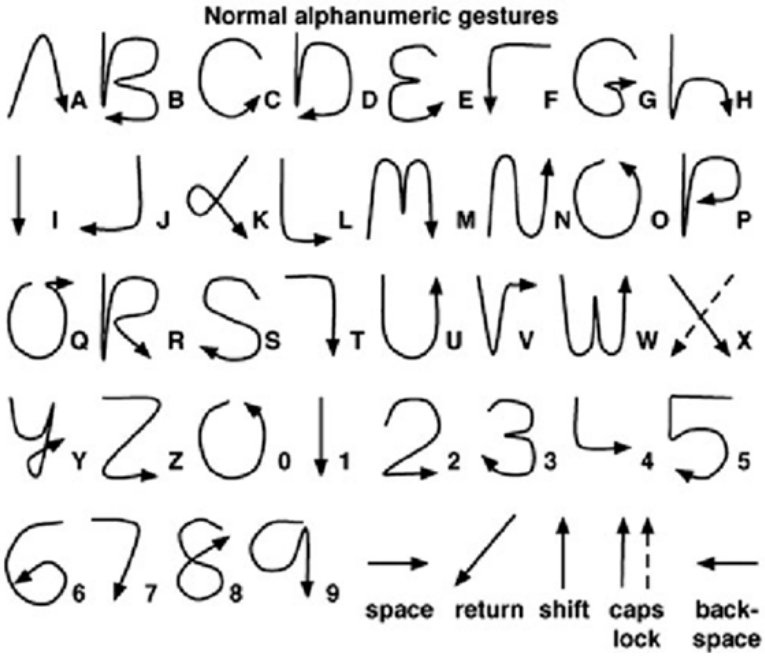


Figure 3. Graffiti characters.

3.2.4. Dataset Formation

In this study, we engaged 17 participants in order to create a character dataset. We asked each participant to write alphabetic characters (A to Z) and numeric digits (0 to 9) about 5–10 times. At the same time, we recorded the position of fingertips using MediaPipe. As a result, we obtained a total sample of 3166 for the alphabet and 1212 for the numeric digit dataset. Each class had approximately 120 samples. The dataset descriptions of alphabet characters and numeric digits are presented in Tables 2 and 3.

Table 2. Dataset descriptions of alphabetic character.

Character	Number of Samples	Character	Number of Samples
A	127	N	121
B	125	O	126
C	126	p	116
D	126	Q	116
E	126	R	121
F	122	S	122
G	120	T	122
H	121	U	121
I	121	V	121
J	120	W	121
K	121	X	123
L	120	Y	121
M	120	Z	122

Table 3. Dataset descriptions of digit dataset.

Digit	Number of Samples
0	116
1	122
2	122
3	121
4	122
5	121
6	123
7	123
8	121
9	121
Total	1212

3.3. Dataset Pre-Processing

The flowchart of data pre-processing is more clearly explained in Figures 4 and 5. The first step was data normalization using min-max normalization [20]. Using this dataset, we created two datasets. The first contained image data. We calculated the moving average [8] from the normalized data, create image data, and processed dilation. Another dataset contained sequential data. The next step was padding in order to fix the length of the coordinate data. Every step is clearly explained as follows:

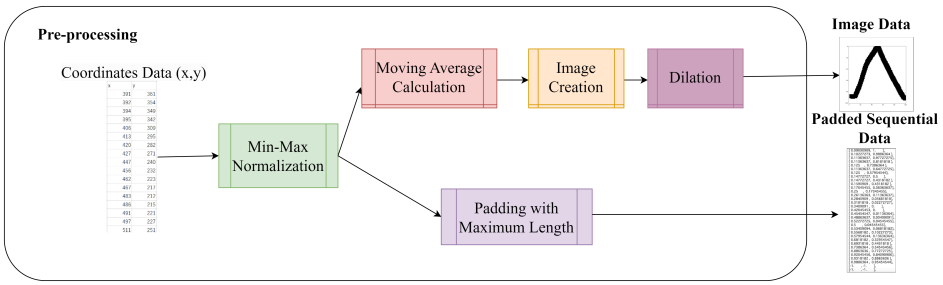


Figure 4. Flowchart of data pre-processing.

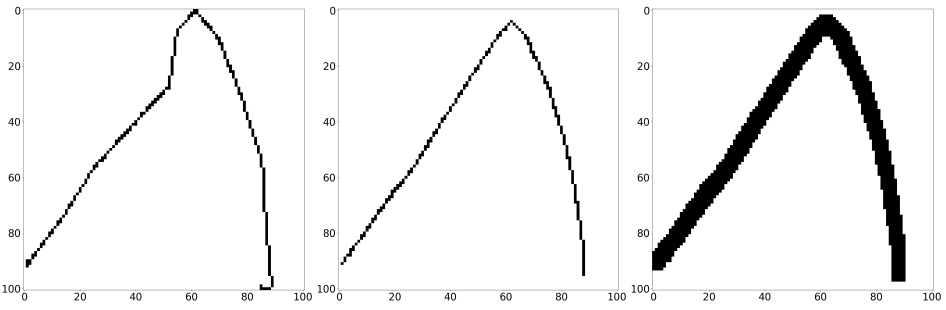


Figure 5. Pre-processing image dataset: (**a**) image before moving average; (**b**) created image; and (**c**) dilation image.

3.3.1. Min-Max Normalization

Min-max normalization [20] is a method of scaling the minimum value to 0 and the maximum value to 1. Specifically, it is an operation that divides the deviation from the

minimum value of the data by the range of the data. This deviation is the maximum value minus the minimum value. This operation converts the minimum value to 0 and the maximum value to 1, making it possible to unify the size of characters written by the user. Additionally, for the maximum and minimum values here, the value with the longer distance in the X or Y direction should be used. This allows to maintain of the vertical-to-horizontal ratio of the characters.

3.3.2. Moving Average Calculation

We calculated the moving average [8] from the min-max-normalized data. A moving average is a technique for smoothing time series data. In addition to audio and image processing, it is also used in fields such as finance and weather [21,22]. It can be found by calculating the simple arithmetic mean of the most recent data. It is a method of adding several terms before and after the term of interest and taking the average. The calculation method differs depending on whether the moving average interval is odd or even. For example, a three-order moving average can be calculated by adding the terms before and after the central term and dividing by 3. On the other hand, the fourth-order moving average added 2 terms before and after the central term, but the farthest x_{i-2} and x_{i+2} are both multiplied by 0.5. Our system used a five-order moving average. We believed that this would reduce the variation in characters between individuals.

3.3.3. Image Creation

We created an image using data that has undergone two possessing steps: min-max normalization and moving average calculation. We created an image by multiplying the specified image size by the processed data and converted it to a two-dimensional array of coordinates and plotted it.

3.3.4. Dilation

Dilation processing was performed to express the characteristics of characters from the created image. "Dilation" refers to replacing the focused white pixel with a black pixel only when there are more than a certain number of black pixels in the 8 neighborhoods when looking at the 8 neighborhoods around a certain white pixel around the black pixel.

3.3.5. Padding

The coordinate data for each character have a different length. Since it is necessary to align the length, padding is performed.

$$x_i' = \frac{x_i - min(x_i)}{max(x_i) - min(x_i)} \tag{1}$$

$$x_i' = \begin{cases} \frac{1}{n}\left\{ x_i + \sum_{j=1}^{k} x_{i-j} + x_{i+j} \right\} & \text{if} \quad n = 2k+1 \\ \frac{1}{n}\left\{ x_i + 0.5(x_{i-k} + x_{i+k}) + \sum_{j=1}^{k-1} x_{i-j} + x_{i+j} \right\} & \text{if} \quad n = 2k \end{cases} \tag{2}$$

3.4. Hybrid Deep Neural Network Architecture

In this work, we propose a hybrid deep-learning model for character recognition, which is based on the combination of CNN and BiLSTM-based models. The first took an input of a 56 × 56 × 1 image and then, we extracted image features by convolutional layers. The first convolutional layer used 32 convolutional filters of size 3 × 3. The second and third convolution layers were 128 and 256 convolution filters of size 3 × 3, respectively. All convolutional layers used rectified linear units (ReLU) as activation functions. The pooling layer used max pooling layer and the pooling size is 2 × 2. Following convolutional layers and max pooling layers, a dropout layer of 0.5, flatten layer, a fully connected layer

with 128 units, a Relu activation function, and a dropout layer of 0.25 were appended. The second took the input of 296 × 2 coordinate data for alphabet recognition. In the case of digits recognition, it took the input of 154 × 2 coordinate data and extracted features of time series data from two BiLSTM layers. The number of units was 128 and the recurrent_droput is 0.5. Finally, the outputs from the CNN and BiLSTM layers were combined, a dropout layer of 0.5 is applied, and a fully connected layer with a SoftMax activation function was added in order to improve the recognition accuracy. The proposed used architecture for character recognition is shown in Figure 6.

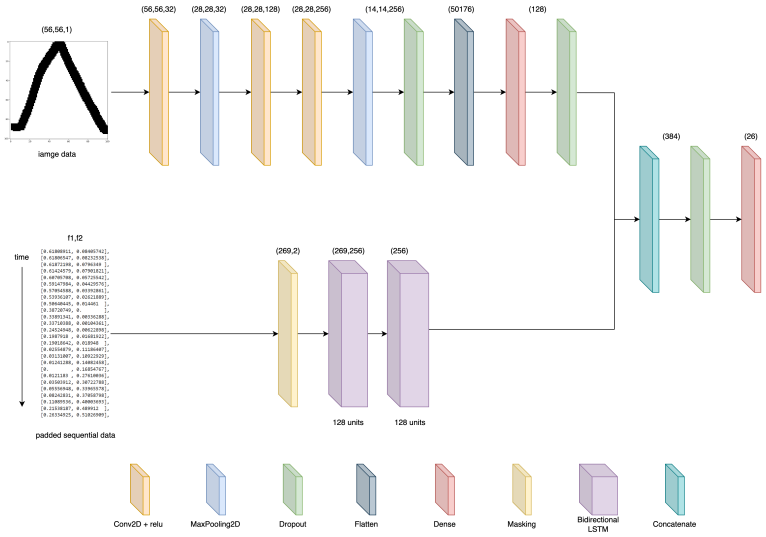


Figure 6. Proposed hybrid the architecture of deep learning model.

4. Experimental Setting and Evaluation Metrics

In this work, we use two datasets in order to evaluate our proposed air-writing system. One is an alphabetic dataset and the other is a digit dataset for character recognition. We trained our proposed air-writing system for character recognition. Classification accuracy was used as an evaluation performance metric, which is calculated using the following formula:

$$Accuracy = \frac{t_p + t_N}{N} \tag{3}$$

Here, $N = t_P + t_N + f_N + f_P$; t_P is the true positive, t_N is the true negative, f_P is the false positive, and f_N is the false negative, respectively.

5. Experimental Results

For character recognition, we used two datasets. One was an alphabet dataset and another was a digit dataset. In this work, we extracted features from image data using CNN, whereas BiLSTM was used to extract features from time-series data. After that, CNN-BiLSTM was implemented for character recognition. Five-fold cross-validation was used to train the model. In five-fold cross-validation, the entire dataset was divided into five equal parts. The first part was used as a test set and the remaining $(5 - 1) = 4$ parts were used to train the model. Then, the trained model was evaluated on the test set and computed recognition accuracy. These processes were repeated five times and finally, we computed the average recognition accuracy. The recognition accuracy of our proposed system is shown in Table 4. From Table 4, it can be seen that our proposed system produced an alphabet recognition accuracy of 99.3% and a number or digit recognition accuracy of 99.5%.

Table 4. Recognition accuracy of our proposed system for character recognition.

Recognition Target	Accuracy (%)
Alphabet	99.3
Digit	99.5

5.1. Comparison Performance between Our Proposed System and Similar Existing Methods

In this section, we compared our proposed character recognition system with similar existing studies using webcams, which is presented in Table 5. For example, Lee and Kim [20] proposed a novel TPS-ResNet-BiLSTM-Attn system for air-writing digit and text recognition and obtained a digit recognition accuracy of 96.0% and a word/text recognition accuracy of 79.7%. Choudhury et al. [23] proposed an ensemble model for Assames handwritten character recognition and obtained a character recognition accuracy of 98.1% and digit recognition accuracy of 98.6%. Yoon et al. [18] proposed a HMMs-based system for alphabetical hand gesture recognition and achieved alphabetic recognition accuracy of 93.8%. Whereas, our proposed system produced an alphabetic recognition accuracy of 99.3% and a digit recognition accuracy of 99.5%, which was comparatively higher than existing studies. We believe that there are two reasons for obtaining this good result. The first was to use Graffiti. Since it can be written with a single stroke, we were able to reduce the variation in characters between individuals. The second was a combination of CNN and BiLSTM. As hypothesized, we succeeded in successfully extracting character features from grayscale images and character features from time-series data.

Table 5. Comparison of character recognition accuracy of our proposed system against similar existing systems using web camera.

Authors	Method	Recognition Target	Accuracy (%)
Lee and Kim [20]	TPS-ResNet-BiLSTM-Attn	Digit	96.0
		Word	79.7
Choudhury et al. [23]	CNN-LSTM	Assamese	98.1
		Digit	98.6
Yoon et al. [18]	HMM	Alphabet	93.8
Proposed method	CNN-BiLSTM	Alphabet	99.3
		Digit	99.5

5.2. Validation of Our Proposed System

In order to validate our proposed system, we used another publicly available 6DMG dataset. The character of the 6DMG dataset was written in one stroke [24]. The dataset contained samples of uppercase characters (A–Z), lowercase characters (a–z), and numeric digits (0–9), which were collected from 22 subjects (males: 17 and females: 5). All subjects were right-handed. Each subject was asked to write each character about ten times and obtained 1470 samples for lowercase characters, 6501 samples for uppercase characters, and 600 samples for numeric digits. In this work, we used only the samples for uppercase characters and numeric digits, which are shown in Table 6.

Table 6. Descriptions of 6DMG character dataset.

Characters	Number of Samples	Total Samples
A–Z	About 250	6501
0–9	60	600

In this work, we implemented our proposed system on the 6DMG dataset and compared its performance with existing studies. The recognition accuracy comparison of our proposed method against existing methods using the 6DMG dataset is presented in Table 7. comparison between our proposed method and existing methods using the 6DMG dataset. As shown in Table 7. our proposed method produced an alphabetic accuracy of 99.48% and a digit accuracy of 99.17% for the 6DMG dataset, which are comparatively higher than existing methods.

Table 7. Accuracy (in %) comparison of our proposed method and existing methods using 6DMG dataset.

Authors	Methods	Alphabetic	Digit
Xu and Xue [25]	LSTM	98.34	97.33
Chen et al. [4]	HMM	98.10	–
Alam et al. [16]	CNN LSTM	99.26 99.32	–
Proposed method	CNN-BiLSTM	99.48	99.17

6. Discussion

In this paper, we proposed an air-writing system using a web camera. Our system can be roughly divided into three steps. The first step was hand joint estimation. We used MediaPipe for joint estimation. It estimated 21 joints from images captured from a webcam. The estimated joint coordinates were composed of (x, y, and z), and a total of 63 pieces of information can be obtained. Our system treated the index finger like a pen. Therefore, the index finger was tracked to obtain the character data. The second step was the pre-processing of character data. In this step, two types of data, image data and time-series data, were created from the collected character data. Image data were created by min-max normalization, calculation of a moving average, and dilation processing. Through these operations, it was possible to create features that emphasize the characteristics of characters by suppressing variations in characters between users without depending on the size of each character written by the user. On the other hand, time series data are created by performing min-max normalization and padding. The third step was to perform character recognition. In this work, we proposed a hybrid model that combined CNN and BiLSTM-based models in order to improve recognition accuracy. The experimental results showed that our proposed system achieved high recognition accuracy. We also validated our proposed system using the 6DMG dataset and compared its performance with existing methods [4,16,25]. Our proposed system achieved high recognition accuracy compared to previous methods using the same dataset. Therefore, our proposed system can perform air-writing without using an expensive special device, which has been a problem in previous studies. The main strength of this study was as follows: (i) Our proposed system was available with a webcam. One could easily collect data using a webcam. Our proposed system is less expensive and has no need for training for collecting data. Whereas, previous studies used special devices, which were expensive and (ii) our proposed air-writing system achieved very high recognition accuracy compared with existing studies.

7. Conclusions and Future Work Direction

In this paper, we proposed a hybrid deep learning model for character recognition, which was based on CNN-BiLSTM. We used Graffiti characters, which was written with one stroke. We used alphabet and digit characters separately in order to recognize characters. In order to character recognition, first, we pre-processed data and made two types of datasets: image data and padding sequential data. The image data was fed into a CNN-based model, whereas the padding sequential data were fed into a BiLSTM model. At the same time, we combined these two models (CNN and Bi-LSTM) in order to improve recognition accuracy.

In this work, this combined model was referred to as a hybrid deep learning model. The experimental results illustrated that our proposed system produced a recognition accuracy of 99.3% for the alphabetic character dataset and 99.5% for the digit character dataset, which was comparatively higher than existing studies. This research can be expected to be a countermeasure against infectious diseases such as coronavirus and useful for enriching our human-computer interaction (HCI) in the future. As a future prospect, we will try to implement thinking about word recognition. Moreover, we will also use special characters and a mix of lowercase and uppercase letters in order to check our proposed system for air-writing character recognition.

Author Contributions: Conceptualization, T.W., M.A.M.H. and J.S.; methodology, T.W., M.M., M.A.M.H., H.-S.L., S.-W.J. and J.S.; investigation, T.W., M.M., M.A.M.H. and J.S.; data curation, T.W., M.M., M.A.M.H., H.-S.L., S.-W.J. and J.S.; writing—original draft preparation, T.W., M.M., M.A.M.H. and J.S.; writing—review and editing, T.W., M.M., J.S. and M.A.M.H.; visualization, T.W., M.M. and M.A.M.H.; supervision, M.A.M.H. and J.S.; funding acquisition, H.-S.L., S.-W.J. and J.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the MSIT (Ministry of Science and ICT), Korea, under the Grand Information Technology Research Center support program (IITP-2023-2020-0-01791) supervised by the IITP (Institute for Information & communications Technology Planning & Evaluation). This work was supported by the Competitive Research Fund of The University of Aizu, Japan.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

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

References

1. Amma, C.; Schultz, T. Airwriting: Bringing text entry to wearable computers. *XRDS Crossroads, ACM Mag. Stud.* **2013**, *20*, 50–55. [CrossRef]
2. Yanay, T.; Shmueli, E. Air-writing recognition using smart-bands. *Pervasive Mob. Comput.* **2020**, *66*, 101183. [CrossRef]
3. Garg, P.; Aggarwal, N.; Sofat, S. Vision based hand gesture recognition. *Int. J. Comput. Inf. Eng.* **2009**, *3*, 186–191. [CrossRef]
4. Chen, M.; AlRegib, G.; Juang, B.H. Air-writing recognition—Part I: Modeling and recognition of characters, words, and connecting motions. *IEEE Trans. Hum.-Mach. Syst.* **2015**, *46*, 403–413. [CrossRef]
5. Alam, M.S.; Kwon, K.C.; Alam, M.A.; Abbass, M.Y.; Imtiaz, S.M.; Kim, N. Trajectory-based air-writing recognition using deep neural network and depth sensor. *Sensors* **2020**, *20*, 376. [CrossRef] [PubMed]
6. Murata, T.; Shin, J. Hand gesture and character recognition based on kinect sensor. *Int. J. Distrib. Sens. Netw.* **2014**, *10*, 278460. [CrossRef]
7. Amma, C.; Gehrig, D.; Schultz, T. Airwriting recognition using wearable motion sensors. In Proceedings of the 1st Augmented Human International Conference, Megève, France, 2–3 April 2010; pp. 1–8. [CrossRef]
8. Hayakawa, S.; Goncharenko, I.; Gu, Y. Air Writing in Japanese: A CNN-based character recognition system using hand tracking. In Proceedings of the 2022 IEEE 4th Global Conference on Life Sciences and Technologies (LifeTech), Osaka, Japan, 7–9 March 2022; pp. 437–438. [CrossRef]
9. Bastas, G.; Kritsis, K.; Katsouros, V. Air-writing recognition using deep convolutional and recurrent neural network architectures. In Proceedings of the 2020 17th International Conference on Frontiers in Handwriting Recognition (ICFHR), Dortmund, Germany, 8–10 September 2020; pp. 7–12. [CrossRef]
10. Amma, C.; Georgi, M.; Schultz, T. Airwriting: Hands-free mobile text input by spotting and continuous recognition of 3D-space handwriting with inertial sensors. In Proceedings of the 2012 16th International Symposium on Wearable Computers, Newcastle, UK, 18–22 June 2012; pp. 52–59. [CrossRef]
11. Arsalan, M.; Santra, A. Character recognition in air-writing based on network of radars for human-machine interface. *IEEE Sens. J.* **2019**, *19*, 8855–8864. [CrossRef]
12. Sonoda, T.; Muraoka, Y. A letter input system based on handwriting gestures. *Electron. Commun. Jpn. (Part III Fundam. Electron. Sci.)* **2006**, *89*, 53–64. [CrossRef]
13. Setiawan, A.; Pulungan, R. Deep Belief Networks for Recognizing Handwriting Captured by Leap Motion Controller. *Int. J. Electr. Comput. Eng.* **2018**, *8*, 4693–4704. [CrossRef]
14. Chen, H.; Ballal, T.; Muqaibel, A.H.; Zhang, X.; Al-Naffouri, T.Y. Air writing via receiver array-based ultrasonic source localization. *IEEE Trans. Instrum. Meas.* **2020**, *69*, 8088–8101. [CrossRef]

15. Saez-Mingorance, B.; Mendez-Gomez, J.; Mauro, G.; Castillo-Morales, E.; Pegalajar-Cuellar, M.; Morales-Santos, D.P. Air-Writing Character Recognition with Ultrasonic Transceivers. *Sensors* **2021**, *21*, 6700. [CrossRef] [PubMed]
16. Alam, M.; Kwon, K.C.; Md Imtiaz, S.; Hossain, M.B.; Kang, B.G.; Kim, N. TARNet: An Efficient and Lightweight Trajectory-Based Air-Writing Recognition Model Using a CNN and LSTM Network. *Hum. Behav. Emerg. Technol.* **2022**, *2022*, 6063779. [CrossRef]
17. Zhang, F.; Bazarevsky, V.; Vakunov, A.; Tkachenka, A.; Sung, G.; Chang, C.L.; Grundmann, M. Mediapipe hands: On-device real-time hand tracking. *arXiv* **2020**, arXiv:2006.10214. [CrossRef]
18. Yoon, H.S.; Soh, J.; Min, B.W.; Yang, H.S. Recognition of alphabetical hand gestures using hidden Markov model. *IEICE Trans. Fundam. Electron. Commun. Comput. Sci.* **1999**, *82*, 1358–1366. [CrossRef]
19. Költringer, T.; Grechenig, T. Comparing the immediate usability of Graffiti 2 and virtual keyboard. In Proceedings of the CHI'04 Extended Abstracts on Human Factors in Computing Systems, Vienna, Austria, 24–29 April 2004; pp. 1175–1178. [CrossRef]
20. Lee, S.K.; Kim, J.H. Air-Text: Air-Writing and Recognition System. In Proceedings of the 29th ACM International Conference on Multimedia, Virtual, 20–24 October 2021; pp. 1267–1274. [CrossRef]
21. Salman, A.G.; Kanigoro, B. Visibility forecasting using autoregressive integrated moving average (ARIMA) models. *Procedia Comput. Sci.* **2021**, *179*, 252–259. [CrossRef]
22. Kothapalli, S.; Totad, S. A real-time weather forecasting and analysis. In Proceedings of the 2017 IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI), Chennai, India, 21–22 September 2017; pp. 1567–1570. [CrossRef]
23. Choudhury, A.; Sarma, K.K. A CNN-LSTM based ensemble framework for in-air handwritten Assamese character recognition. *Multimed. Tools Appl.* **2021**, *80*, 35649–35684. [CrossRef]
24. Chen, M.; AlRegib, G.; Juang, B.H. 6dmg: A new 6d motion gesture database. In Proceedings of the 3rd Multimedia Systems Conference, Chapel Hill, NC, USA, 22–24 February 2012; pp. 83–88.
25. Xu, S.; Xue, Y. A long term memory recognition framework on multi-complexity motion gestures. In Proceedings of the 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), Kyoto, Japan, 9–15 November 2017; Volume 1, pp. 201–205.