

Article

Assistive Data Glove for Isolated Static Postures Recognition in American Sign Language Using Neural Network

Muhammad Saad Amin ¹, Syed Tahir Hussain Rizvi ^{2,*}, Alessandro Mazzei ¹ and Luca Anselma ¹¹ Dipartimento di Informatica, Università degli Studi di Torino, 10149 Torino, Italy² Department of Electrical Engineering and Computer Science, University of Stavanger, 4021 Stavanger, Norway

* Correspondence: tahir098@msn.com

Abstract: Sign language recognition is one of the most challenging tasks of today's era. Most of the researchers working in this domain have focused on different types of implementations for sign recognition. These implementations require the development of smart prototypes for capturing and classifying sign gestures. Keeping in mind the aspects of prototype design, sensor-based, vision-based, and hybrid approach-based prototypes have been designed. The authors in this paper have designed sensor-based assistive gloves to capture signs for the alphabet and digits. These signs are a small but important fraction of the ASL dictionary since they play an essential role in *fingerspelling*, which is a universal signed linguistic strategy for expressing personal names, technical terms, gaps in the lexicon, and emphasis. A scaled conjugate gradient-based back propagation algorithm is used to train a fully-connected neural network on a self-collected dataset of isolated static postures of digits, alphabetic, and alphanumeric characters. The authors also analyzed the impact of activation functions on the performance of neural networks. Successful implementation of the recognition network produced promising results for this small dataset of static gestures of digits, alphabetic, and alphanumeric characters.

Keywords: assistive glove; American Sign Language (ASL); gesture recognition; neural network (NN); sign recognition



Citation: Amin, M.S.; Rizvi, S.T.H.; Mazzei, A.; Anselma, L. Assistive Data Glove for Isolated Static Postures Recognition in American Sign Language Using Neural Network. *Electronics* **2023**, *12*, 1904. <https://doi.org/10.3390/electronics12081904>

Academic Editors: Enzo Pasquale Scilingo and Rui Pedro Lopes

Received: 14 November 2022

Revised: 16 February 2023

Accepted: 27 March 2023

Published: 18 April 2023



Copyright: © 2023 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 (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

In today's world of smart technology, sign language (SL) recognition is a major task. This is also the need for time as it can be used to overcome the communication gap for the Deaf (The cap-case "Deaf" word refers to a community of deaf people who share a language and a culture. In contrast, the lower-case "deaf" refers to the audiological condition of not hearing). Globally, almost every country has Deaf communities (according to the world's population, 15% to 20% of people are part of the deaf population [1]) and people from these communities are not always able to communicate by using the vocal national language in written form. So, in order to help Deaf communities to overcome the language barrier, many researchers try to develop software and hardware translation systems. For this purpose, different methodologies such as sensor based, vision based, or hybrid approaches have been adopted in the literature to design assistive models for capturing sign gestures [2]. These methodologies require the acquisition of posture data made by Deaf people.

Sensor-based prototypes cope with different types of sensors only [3–5]. Choosing a good combination of different sensors is a subjective matter [6]. Based on the dataset and classification requirements, a variety of different sensors can be used collectively. However, this creates a problem. If the number of sensors is increasing, then system complexity and cost are also increasing, and complex systems often result in low or bad accuracy [7]. Similarly, for vision-based approaches, only image-based or video-based data can be analyzed [8]. Usually, there is no proper involvement of sensors in the vision-based model. However, this model also has some drawbacks regarding data extraction

from the foreground, background, and noisy channels [9]. Lastly, a hybrid approach is the combination of both sensor-based and vision-based models [10]. This approach can normally be used for experimental setups, though the cost of these prototypes is very much high and the prototype models are very much complex. For the fast computation of data, normally GPUs or GPGPUs are required [11].

In this paper, we have developed a smart assistive glove (*data glove*) to capture two specific sets of signs which are alphabetical signs and the numbers 0–10. Even though numbers and alphabetical signs are a small fraction (thirty-seven signs) of the ASL dictionary (The project <https://www.spreadthesign.com/> (accessed on 1 February 2023) contains more than 20,000 signs for over 40 sign languages), these signs play an essential role in *fingerspelling*, which is a universal signed linguistic strategy for expressing personal names, technical terms, gaps in the lexicon and emphasis [12]. Both alphabetical signs and numbers are signs which can be captured by a data glove since, to the best of our knowledge, in ASL they are naturally signed only with hands, that is without using other articulators such as the head, eyebrows, or shoulders.

This data glove contains five flex sensors embedded on each finger of the hand and a gyroscope sensor attached to the top of the palm [13]. According to the posture orientation for standard numeric (ASL), as shown in Figure 1, the dataset is collected for thirty-seven different sign postures. These sign postures include data for the digit numbers 0 to 10 and from the letter A to letter Z. Self-collected thirty-seven separate postures data are used to train the fully connected bilayered and trilayered neural networks. A scaled conjugate gradient back propagation-based algorithm is used to perfectly classify these sign data. Listing all the deployments of the designed model, the whole procedure consists of the following points.

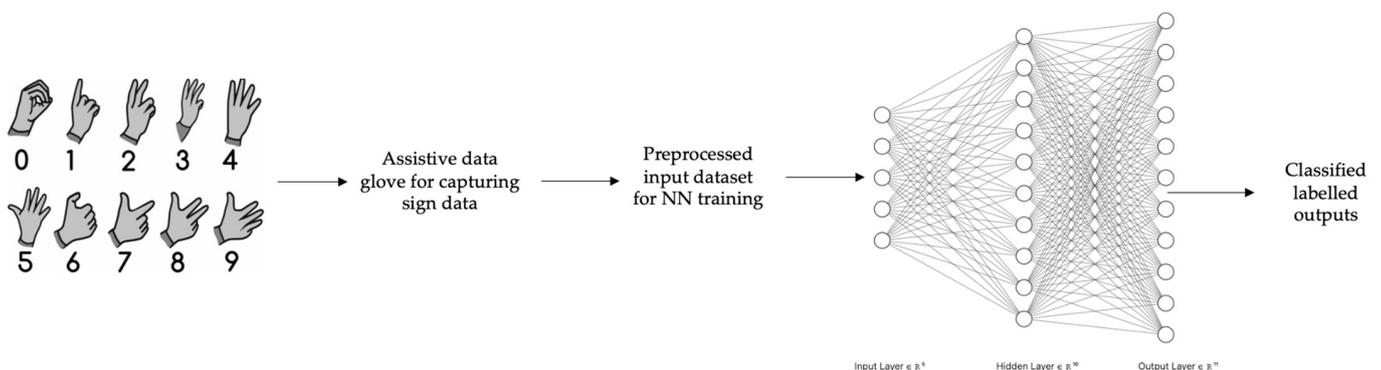


Figure 1. An overview of complete methodology for NN-based ASL recognition.

1. Developing an assistive glove based on flex and gyroscope sensors;
2. Collecting datasets for numeric, alphabetic, and alphanumeric (i.e., numbers and alphabet) ASL;
3. Training NN models;
4. Analyzing the impact of activation functions on the performance of neural models;
5. Testing the trained models.

This proposed framework is novel as it is utilizing just two kinds of sensors to catch the total ASL numbers and letter sets information, which simplifies our model. Beforehand, various researchers working in the SL space had utilized a wide range of sensors that made the framework complex. Due to a vast amount of information based on sensor values, significant performance parameters such as general framework precision, effectiveness, and acquisition time are impacted [14–19]. In the framework we propose, just two sorts of sensors are utilized, i.e., a flex sensor to acquire the finger bowing data and an accelerometer/gyroscope sensor to get the hand orientation. Furthermore, we had to gather information physically since, to the best of our knowledge, no dataset containing the total data on ASL stances utilizing just two sorts of sensors is available. This allowed us

to perform the acquisition of the new dataset in an efficient way as no complex information had to be collected. In addition to this, we have carried out our developed dataset on various variations of neural networks and got noteworthy and cutting-edge performance results as discussed in the later sections. These outcomes reflect the quality of our gathered ASL dataset. Previously, most analysts exploited just one kind of neural model for obtaining maximum precision and accuracy results [20].

We also used TabTransformers and gMLP-based most recent and state-of-the-art models, though it does not perform well with our data in some preliminary experiments. This is due to the two following reasons: (one) our dataset does not have any categorical features, thus it contains all numeric features representing only sensor values; and (two) we obtained an overfitting of data, which is obviously not desirable. To make the classification and recognition process simple, we preferred to use a fully connected version of the neural network (i.e., multilayer perceptron—MLP).

We are aware that data gloves are not always well accepted by the Deaf community for at least two reasons. The first technical limitation of the data gloves is that they cannot capture articulators differently by hands [21,22]. Furthermore, in our project, we focus on numbers and alphabetical signs that are signed using only hands. A second sociological limitation of data gloves is that the burden of communication by wearing the glove is taken only by the deaf person to produce a one-way asymmetrical communication Deaf-to-not Deaf, thus not solving the general problem of accessing the speech. We believe that this second limitation is generally true, though, in some specific situations, data gloves can be used advantageously. For instance, gloves can be used as educational tools for SL learners. Moreover, in particular tasks, such as buying tickets in person, one can imagine that a Deaf person can use the glove for communicating the name of a city to a not-signing ticket seller by using fingerspelling [23].

The remaining paper is structured as follows: a literature review is discussed in Section 2. Section 3 focuses on methodology. Materials and methods are discussed in Section 4. The results of the implementation are briefly discussed in Section 5, and Section 6 presents conclusion statements.

2. Literature Review

Accurate identification and classification of sign gestures perfectly and accurately is always a challenging task for all researchers in this domain. Many different techniques and methodologies have been adopted to perform this task. Different strategies have also been adopted for capturing and classifying postures data. Keeping in mind the major aspects of sign language, literature review-based studies are categorized into three main domains. Sensor-based recognition models, vision-based recognition models, and hybrid recognition models.

Sensor-based recognition models purely focus on one or a combination of different types of sensors. For data acquisition, flex sensors, gyroscope sensors, accelerometer sensors, contact sensors, optical sensors, or inertial motion sensors have been used [24]. Authors have used the mentioned sensors solo or in combination with different sensors to capture sign data [25]. Some of the authors have also worked on EEG signals for capturing brain data in the form of analog signals and then converting analog data into digital form for machine training [26–29]. In this challenging aspect, some authors have also used commercial data gloves that are purely made for capturing gesture data. However, in this scenario, the purpose of using an already made commercial data glove is to increase the accuracy and efficiency of an already-developed model [30]. Some of the authors in this domain have also focused on regional languages e.g., Pakistani sign language, Italian, Indian, Arabic, Russian, Chinese, Taiwanese, and Persian SLs, etc. [31–35]. This is considered a more challenging task as there is no predefined dataset available for regional languages and all the time authors must collect their own dataset for very few postures [36,37]. The good thing about sensor-based prototypes is that they are each worn and carried in

public. The resultant data is normally displayed on an LCD or transmitted to the mobile or computer screen via a Bluetooth module [38,39].

In concluding the literature discussion, our model is better than the literature models due to the accompanying reasons. Initially, the majority of the writers have zeroed in on just a single sort of SL information, for example, numbers or letter sets. Some of them had zeroed in on both, however, this did not cover the total ASL domain due to stance and sensor intricacies. However, we have zeroed in on all ASL numbers and letters in order and a blend of numeric and alphabetic information also. Secondly, due to the expanded number of sensors, by and large, framework effectiveness and precision have not created many astonishing outcomes. However, in our model, we have utilized an extremely fine blend of two kinds of sensors that gave us the best outcomes with phenomenal precision and effectiveness. Third, a large number of authors take care of just the AI or neural network model that gives them great outcomes. Be that as it may, we have tried our manually-collected dataset on different neural models and it performed very well in all neural formats, which mirrors the creativity and flawlessness of our model and information. A point-by-point accuracy examination is likewise recorded in tabular form in the results and discussion section.

3. Methodology

In sign language recognition, there is a list of concatenated tasks starting from capturing posture data with the help of an assistive glove to the identification of resultant values. For the development of assistive gloves, five flex sensors and gyroscope sensors are used. It is a property of the flex sensor to produce a resistance value based on the bending performed to make gestures. Sensors attached to each finger and the palm of the hand help in getting values regarding one-sign posture. A user wearing an assistive glove will make sign gestures for ASL and the resultant sensor values will be analyzed and captured with the help of a microcontroller. A prototype design is a combination of microcontrollers and sensors. The purpose of the microcontroller in the development of assistive gloves is to capture sensor-based values and transmit these values to the processing unit i.e., the computer or server. These collected values are preprocessed and then stored in a database or file with the help of a parallax microcontroller data acquisition add-on tool for Microsoft Excel (PLX-DAQ). The core functionality of PLX-DAQ is the transmission of sensor values i.e., coming through a microcontroller via serial communication directly into the Excel file. This is the point where dataset generation is performed by collecting all sensor values into a local or online server-based file. This processed data is forwarded to a neural network for training purposes. Once a model is completely trained, it is tested for new incoming data to analyze its performance. The complete methodology is discussed in Figure 1, which displays a neural network-based classification process for digits. Alphabetic and alphanumeric neural models also work in the same way.

Neural network-based implementation of sign language requires data in numeric format. The preprocessed data is utilized as input to train a fully connected neural network. Based on patterns of sensor values, deep gesture classification is performed for thirty-seven sign postures. A scaled conjugate gradient back propagation algorithm is used which has proved helpful in getting maximum accuracy.

4. Materials and Methods

Materials are the connected components that are used collectively for capturing sign postures. In our developed assistive glove, we have used flex sensors, gyroscope sensors, resistances, and an Arduino microcontroller as materials and we have used a neural network-based scaled conjugate gradient back propagation algorithm as a method to classify postures made by wearing an assistive glove. A very brief description of materials and methods is discussed in the sections below.

4.1. Hardware Components

4.1.1. Flex Sensor

A flex sensor is also known as a bending sensor. The internal structure of the flex sensor is based on a phenolic resin substrate with conducted ink deposits which produces increased resistance when it is bent to some angle. A flex sensor works on the principle of the voltage divider rule where V_{in} is the input voltage, V_{out} is the final output voltage, while R_1 and R_2 are combinations of fixed resistances, and R_{flex} is the resistance of the flex sensor, as shown in Equation (1):

$$V_{out} = V_{in} [R_1 / (R_2 + R_{flex})] \quad (1)$$

The bending of the flex sensor is directly proportional to the resistance value. The higher the bending is, the higher the resistance inside the material. The physical shape of the sensor consists of two pins. While interconnecting with the microcontroller, as shown in Figure 2, one pin is connected with the analog pin of the microcontroller, and the other pin is connected to the ground. To avoid voltage overflow, a minimum value resistance is also connected to the pin of the flex sensor. In our assistive glove, we have used five flex sensors and five resistances connected with these flex pins.

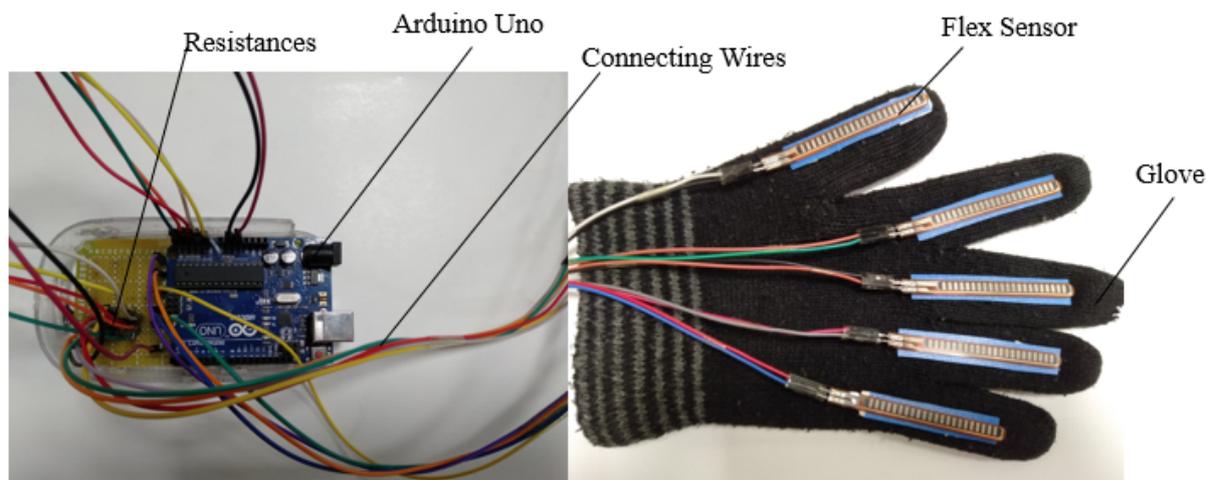


Figure 2. Flex sensor, gyroscope, and Arduino-based data glove designed for capturing ASL-based gestures.

4.1.2. MPU 6050

A gyroscope sensor is a three-axis based shrewd sensor gadget that assists in catching with protesting direction. Concerning the SLR model, not all standard American sign motions can be caught with just flex sensors. This is due to the idea of sign motions. All number-based sign motions have no kind of stance covering. Taking into account the ASL letter sets' poses, this is a kind of complicated characterization issue having 26 classes. In an alphabet-based recognition problem, posture overlapping happens. Certain signals cannot be caught without catching motion direction. A gyroscope sensor is utilized in this experiment to effortlessly catch sign directions.

Hand orientations made toward any direction are caught as 3-axis-based numeric values. Three-directional information is captured as the angle is caught. Hand orientation-based or directional change in representing any letter set is caught with the assistance of three parametric values, for example, the x-axis, y-axis, and z-axis. A complete prototype design is shown in Figure 2.

4.1.3. Arduino Microcontroller

For processing the input data from the sensors, an AT mega 328P-based Arduino microcontroller is used. This microcontroller has both analog and digital pins attached to it.

This is a 10-bit microcontroller having values ranging from 0 to 1023 and can easily operate on 16 MHz frequency. Arduino has 32 KB of memory and 2 KB of RAM for quick data processing. It can easily be operated with the help of a 5v DC battery or by connecting with the USB port on the computer. While interconnecting with flex sensors, five sensor pins are connected with the five analog ports of Arduino, and the common ground of Arduino is attached to all the second pins of flex sensors. A simple interconnection of the flex and the Arduino microcontroller is shown in Figure 3.

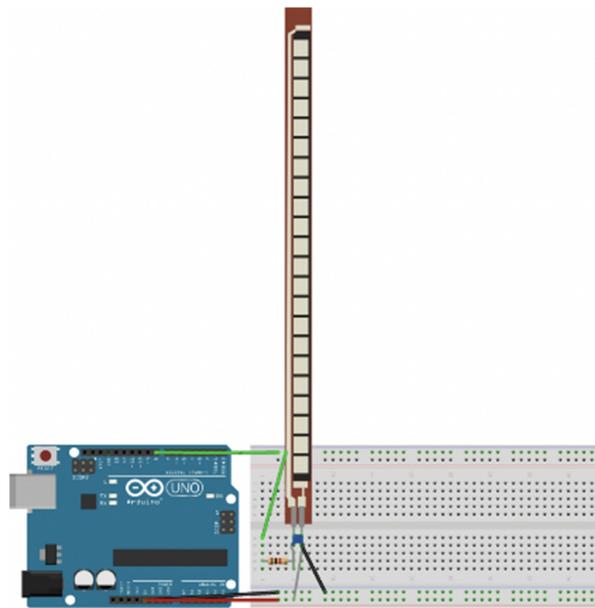


Figure 3. A simple interconnection of the flex sensor with the Arduino microcontroller.

4.2. Dataset Generation

For the implementation of SL classification, we have used a self-collected dataset based on the flex and gyroscope sensor values. For this experiment, we have created and gathered three separate datasets: numeric ASL having 11 (numbers 0 to 10), alphabetic ASL having 26 classes (letters A to Z), and alphanumeric ASL stances having 37 classes (0–10 and A–Z). Every SL pose has 200 examples gathered from 9 distinct male and female volunteers 24 to 26 years of age. All datasets are gathered under ordinary conditions of the lab. The dataset size for every variation can be determined by multiplying the number of sign posture classes with the number of SL samples gathered for each stance. This dataset is further split into training, validation, and testing sets for neural implementation.

4.3. Neural Network Architecture

The classification of sign gestures is usually considered a complex task. In our experiment, we have used a fully connected bilayered and trilayered neural network having 5 inputs and 11 outputs for the digit datasets, as shown in Figure 1; similarly, 8 inputs and 26 and 37 outputs for alphabet and alphanumeric datasets, respectively. After the input layer, the second layer is the hidden layer and the third one is the output layer. The preprocessed training data is fed into the network through the input layer and the resulting classified data is analyzed through the output layer of the network. All the statistical information of the bilayered and trilayered neural models is listed in Table 1.

Table 1. Statistical information of different variants of neural models on digit, alphabetic, and alphanumeric datasets.

Neural Network Specifications		Dataset Type		
		Digit	Alphabet	Alphanumeric
BiLayered	Dataset Split	80% training, 10% validation, and 10% testing		
	Training Algorithm	Scaled Conjugate Gradient based Back Propagation		
	Training accuracy	97.7%	95.3%	96.5%
	Testing Accuracy	94.3%	90.7%	91.5%
	Prediction speed	170,000 obs/s	260,000 obs/s	270,000 obs/s
	Training time	1.9319 s	21.79 s	50.77 s
	Connected Layers	2		
	Each layer size	10		
	Regularization strength	Lambda		
	Performance	Cross Entropy Error		
	Activation Functions	ReLU, Tanh, Sigmoid		
	Iteration Limit	1000		
	ReLU Accuracy	98.7%	97.5%	95.1%
	Tanh Accuracy	95.5%	94.8%	93.3%
	Sigmoid Accuracy	91.9%	92.6%	90.2%
TriLayered	Connected Layers	3		
	Each layer size	10		
	Regularization strength	Lambda		
	Performance	Cross Entropy Error		
	Activation Functions	ReLU, Tanh, Sigmoid		
	Iteration Limit	1000		
	ReLU Accuracy	96.8%	93.2%	97.6%
	Tanh Accuracy	94.7%	92.5%	95.9%
	Sigmoid Accuracy	90.4%	87.9%	78.5%

4.4. Scaled Conjugate Gradient Back Propagation Algorithm

We consider the scaled conjugate gradient (SCG) back propagation algorithm for implementing back propagation. With respect to other algorithms, it is computationally fast and does not require a line search after each iteration. Equation (2), given below, is the mathematical notation of the SCG algorithm where $E(w)$ is a global error function that depends on the biases and the weights associated with the neural network. $E(w)$ is calculated with one forward pass and $E'(w)$ is calculated with one backward pass of the neural network iteration. On each iteration, the optimal distance is measured which leads to a better line search for gradient computation as in Equation (3). In Equation (3), p is the number of patterns presented to the network as weighted vectors during training, and a_k denotes the step size of the function that aims at regulating the indefiniteness of the Hessian metrics.

$$E(w + y) = E(w) + E'(w)^T y + \frac{1}{2} 2y^T E''(w) y \quad (2)$$

$$y_{k+1} = y_k + a_k \times p_k \quad (3)$$

The complete operational pipeline of the proposed model starts with the prototype design. The purpose of making a new data glove is twofold; (one) it is possible to capture all static sign postures with the help of only two sensors. This can make the computational model less complex and fast in computations, and (two) analysis of the neural model performance in case of less complex data samples i.e., whether it perfectly classifies or goes towards underfitting or overfitting. While experimenting with capturing signs, in between transitions of signs occurred when the signer switched from one posture to another posture. To cope with this problem, we adopted a dual conditional approach i.e., we first checked the orientation of each finger for each ASL posture and then analyzed the hand orientation for each individual posture. Then, we set the minimum and maximum range for each sensor to get the label of each posture made by the signer. In case of the posture perfectly matching

the ranges of sensor value, we get the numeric or alphabetic label by the microcontroller, e.g., 1,2,3 or A, B, C. In case of no matching, we get ‘-1’ as noise which was filtered out for dataset formation.

5. Results and Discussion

Sign language recognition being the most emerging and challenging domain requires very efficient and accurate findings. Results obtained after the successful implementation of the discussed models are illustrated in detail in this section. Statistical information of the neural model used for classification and recognition is completely listed in Table 1. The information of the model includes the preset, the number of fully connected layers, the first layer size, the activation function used, the limit of maximum iterations, the prediction speed, the accuracy, and the training time. As in the implementation, different variants of neural networks are used. Therefore, statistical information related to each neural model is included in the table. Apart from different neural models, three different types of datasets are also used. These different datasets include digits, alphabets, and alphanumeric datasets. A very comprehensive description of each dataset is reported below.

a. Number datasets

The number dataset contains sensor information for eleven distinct stances. These stances incorporate information from numbers 0 to 10, hence this is an 11-class problem. Training of the neural network results into a display of performance in the form of training, validation, and testing plots occurred. These plots provide information concerning epochs and cross entropy of the model under progress. The blue line indicates training, the green line reflects validation, the red line displays testing, and the dotted line highlights the best performance of the model. The best validation performance for digits is 9.1511×10^{-7} at the 59th epoch, as shown in Figure 4a. For digit classification, only flex sensors are utilized. Therefore, the value ranges for five flex sensors are listed on the y-axis and the total number of sign gestures for 11 numbers of ASL sign postures are displayed on the x-axis of Figure 4b. Each color represents each flex sensor attached to the prototype.

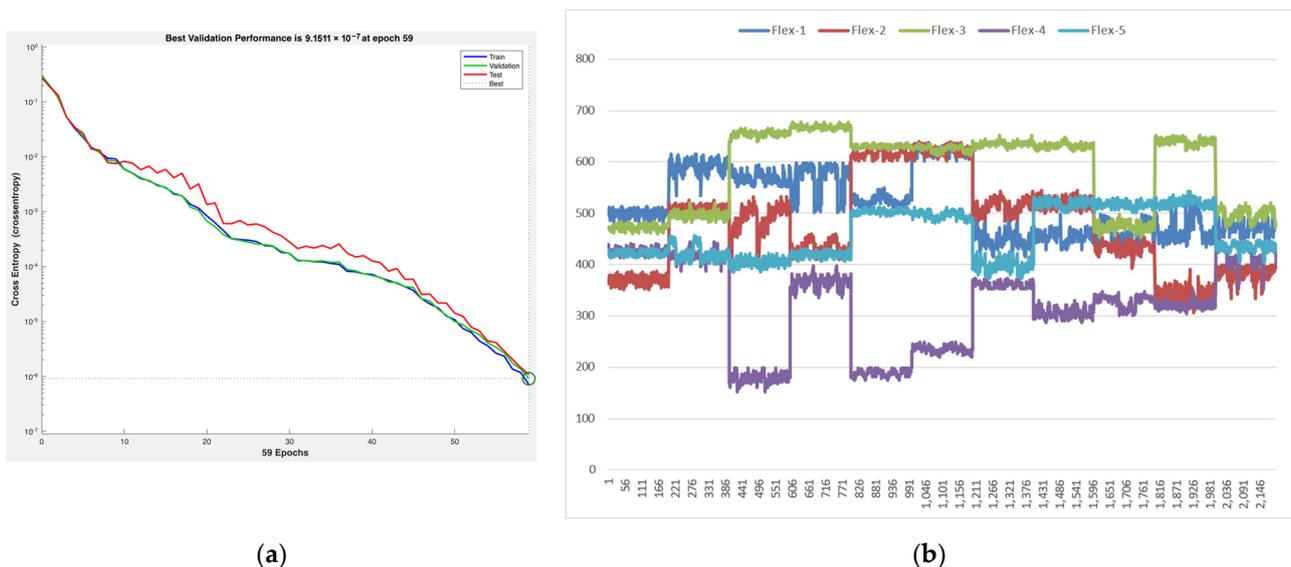


Figure 4. Training, validation, and testing performance plot (a) along with flex sensors values plot (b) for number dataset.

b. Alphabets dataset

The alphabet dataset contains sensor information for twenty-six distinct stances. These stances incorporate information from letters A to Z, hence this is alluded to as a 26-class problem. The training, validation, and testing plot of the alphabetic neural network is

shown in Figure 5a below with the best validation performance of 1.2097×10^{-6} at the 62nd epoch. For alphabet classification, a combination of flex sensors, accelerometer, and gyroscope sensors are utilized. Therefore, the value ranges for the five flex sensors, the three-axis accelerometer, and the gyroscope sensors are listed in the y-axis and the total number of sign gestures for the 26 letters of ASL sign posture is displayed on the x-axis of Figure 5b. Each color represents each sensor value attached to the prototype.

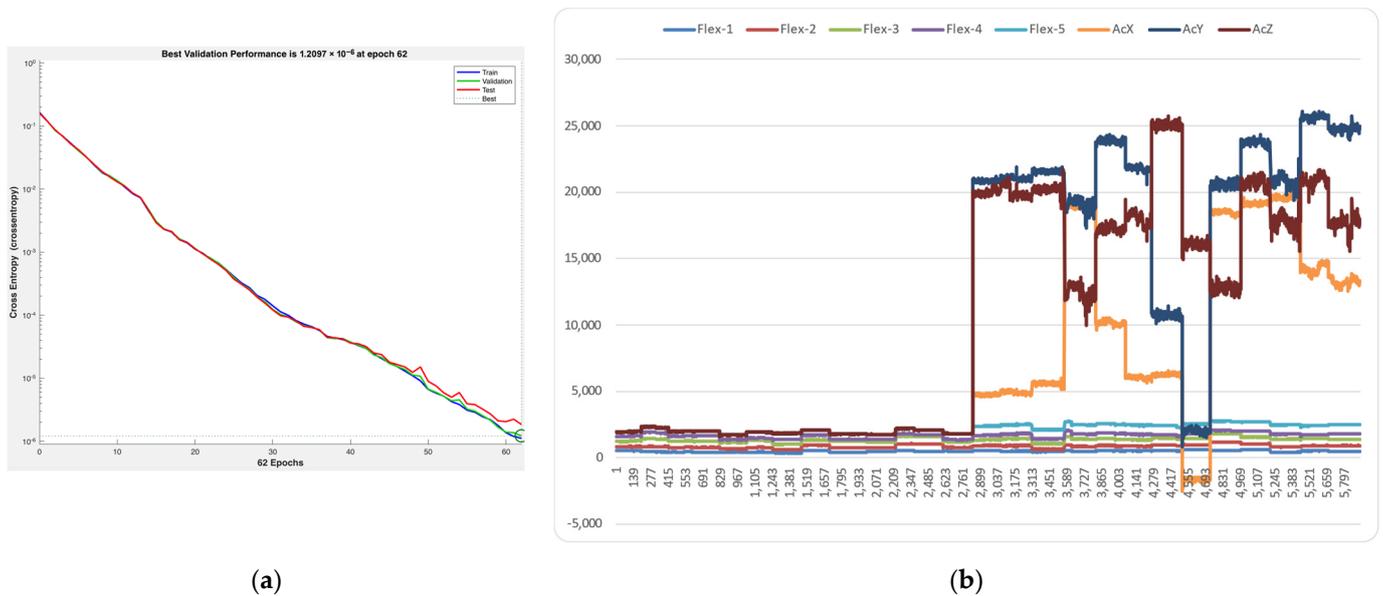


Figure 5. Training, validation, and testing performance plot (a) along with flex, accelerometer, and gyroscope sensors values plot (b) for the alphabets dataset.

c. *Alphanumeric dataset*

The alphanumeric dataset contains sensor information for thirty-seven distinct stances. These stances incorporate information from letters A to Z and data from numbers 0 to 10, hence this is alluded to as a 37-class problem. The training, validation, and testing plot of the alphanumeric neural network is shown in Figure 6a below with the best validation score of 1.6671×10^{-6} at the 102nd epoch. For alphanumeric sign classification, the same combination of flex sensors, accelerometer, and gyroscope sensors is utilized. Therefore, the value ranges for the five flex sensors, the three-axis accelerometer, and the gyroscope sensors are listed on the y-axis, and the total number of sign gestures for the 37 alphanumeric ASL sign postures are displayed on the x-axis of Figure 6b. Each color represents each sensor value attached to the prototype.

Activation functions play a very important role in updating the weights of the neural nodes during training. Choosing the correct and most appropriate activation function for your model helps in achieving good accuracy and training results. The authors in this paper also adopted the strategy of analyzing the impact of activation functions on the performance of the neural networks by using three different activation functions i.e., ReLU, Tanh, and Sigmoid. Replicating the same experiment by changing the activation function results in different accuracies, as listed in Figure 7 below. This experimental strategy is repeated six times by taking three types of activation functions on a bilayered neural network shown in Figure 7a and then implementing the same three types of activation functions for the trilayered neural network shown in Figure 7b. The analysis states that for the bilayered neural networks, ReLU has the highest accuracy for all formats of the dataset, i.e., number, alphabetic, and alphanumeric. Tanh stands second in this implementation and sigmoid lags due to the mathematical behavior of the function. The same is the case in the trilayered neural network model. ReLU performs very well by providing the best results

for number, alphabetic, and alphanumeric datasets. Tanh stands second and sigmoid is in the last stage in this comparison. All these model values are also listed in Table 1.

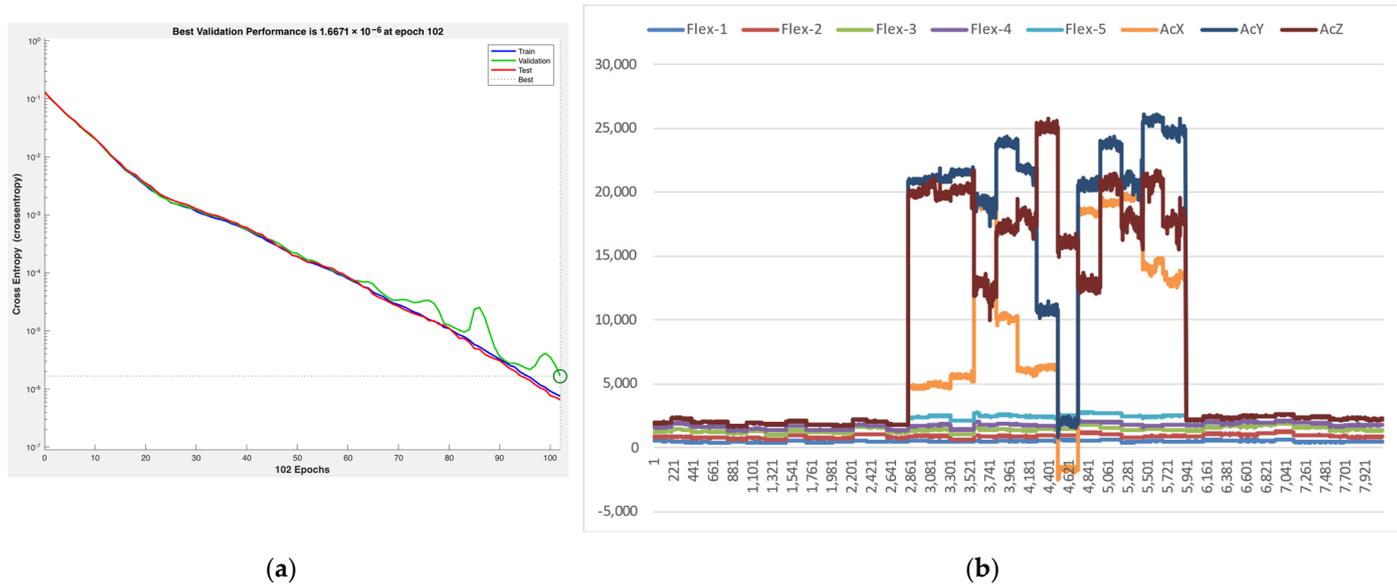


Figure 6. Training, validation, and testing performance plot (a) along with flex sensors values plot (b) for the alphanumeric dataset.



Figure 7. Activation function impact on the accuracy of the neural model.

Accuracy comparison is a good way of checking the developed model’s progress. Considering the literature review-based implementation of gesture classification, we have compared the results of the literature with our results. Table 2, given below, highlights the algorithmic performance of the literature model corresponding to the accuracy and the reference number. Comparing our results (in bold) with the literature review, it is clearly seen that our model performed very well in all aspects of evaluation, i.e., accuracy, speed, and training time.

Table 2. Literature review-based accuracy analysis and comparison.

Sr. No	Literature-Based Recognition Models	Accuracy
1	Support Vector Machine (SVM) [1]	91.93%
2	Template-matching approach [5]	83.58%
3	Template-matching approach [6]	99.5%
4	DTW and Nearest Mapping [7]	96.5%
5	LDA, KNN and SVM [10]	98%
6	Template-matching approach [12]	92%
7	Wrist-based gesture recognition system [13]	92.66% and 88.8%
8	Local Fusion algorithm on motion sensor [15]	91%, 92%, and 93%
9	K-Nearest Neighbor (KNN) [17]	99.53% for static gestures and 98.64% for dynamic gestures
10	Multilayer Perceptron [23]	96.1%
11	Template-matching algorithm [26]	98%
12	Convolutional Neural Network [31]	92.88%
13	Recurrent Neural Network [36]	95%
14	Feed-forward Artificial Neural Network [32]	91.11%
15	Color segmentation and Neural Network [33]	90%
16	Multistream 3D CNN [34]	91%
17	Long Short. Term Memory Networks [37]	92.8%
18	Artificial Neural Network [38]	93.91%
19	3-branch Convolutional Neural Network [35]	90%
20	Bilayered NN (digit dataset)	98.7%
21	Bilayered NN (alphabet dataset)	97.5%
22	Bilayered NN (alphanumeric dataset)	95.1%
23	Trilayered NN (digit dataset)	96.8%
24	Trilayered NN (alphabet dataset)	93.2%
25	Trilayered NN (alphanumeric dataset)	97.6%

For experimental and educational purposes these types of assistive technologies play a very vital and effective role in society. For experimentation, the focus of researchers is mainly on computational speed, model performance, prototype cost, and recognition response. However, the prototypes associated with real-time recognition or translation of sign postures must deal with all types of social factors as well, i.e., enabling two-way communication by not putting the burden of communication on the Deaf only. Considering the applications of sign-to-speech (S2S) assistive technologies, they only deal with 50% of problems in the case of Deaf people.

Similarly, dealing with regional languages, e.g., Italian, Spanish, etc., requires much experimental and analysis work to do since sign gestures for every region vary from each other. Even considering just one regional language, it is not possible to capture and translate all language postures with the data glove only. Data gloves can only capture hand movements, not arm, head, articulation, and other body movements. If we consider increasing the number of sensors to capture all movement types, then it would be very unrealistic to go in public with a body full of sensors. These are some challenges and future directions associated with our implementation that can lead researchers to think and work accordingly.

6. Conclusions

In this paper, neural network-based model for sign language recognition was proposed where the assistive glove was designed and implemented for capturing real-time data and compiling it into a dataset. Among different domains of gesture classification, we have focused on the purely sensor-based implementation of standard ASL postures. An assistive glove was used to collect a dataset having 200 samples each for 11 numbers, 26 letters, and 37 alphanumeric sign postures. A fully connected bilayered and trilayered neural network was used to classify eleven, twenty-six, and thirty-seven isolated static sign gestures. A scaled conjugate gradient back propagation algorithm was used to train neural models for

the self-collected datasets. The impact of the activation function on the performance of the model was also analyzed in this paper. Successful implementation of the model has helped the authors in achieving promising training and testing accuracy for numeric, alphabetic, and alphanumeric datasets, respectively.

However, our self-generated dataset has a small portion of static gestures used by the American Sign Language Community. In the future, all representative samples of ASL would be collected using this glove and other models would be trained to perform the recognition.

Author Contributions: Conceptualization, M.S.A.; Methodology, M.S.A. and S.T.H.R.; Software, M.S.A.; Writing—original draft, M.S.A.; Writing—review & editing, S.T.H.R., A.M. and L.A.; Supervision, S.T.H.R., A.M. and L.A. 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. Amin, M.S.; Rizvi, S.T.H.; Hossain, M. A Comparative Review on Applications of Different Sensors for Sign Language Recognition. *J. Imaging* **2022**, *8*, 98. [[CrossRef](#)] [[PubMed](#)]
2. Amin, M.S.; Rizvi, S.T.H. Sign Gesture Classification and Recognition Using Machine Learning. *Cybern. Syst.* **2022**, 1–15. [[CrossRef](#)]
3. Amin, M.T.; Latif, M.Y.; Jathol, A.A.; Ahmed, N.; Tarar, M.I.N. Alphabetical Gesture Recognition of American Sign Language using E-Voice Smart Glove. In Proceedings of the 2020 IEEE 23rd International Multitopic Conference (INMIC), Bahawalpur, Pakistan, 5–7 November 2020; pp. 1–6. [[CrossRef](#)]
4. Kim, S.; Kim, J.; Ahn, S.; Kim, Y. Finger language recognition based on ensemble artificial neural network learning using armband EMG sensors. *Technol. Health Care* **2018**, *26*, 249–258. [[CrossRef](#)] [[PubMed](#)]
5. Verdadero, M.S.; Cruz, J.C.D. An Assistive Hand Glove for Hearing and Speech Impaired Persons. In Proceedings of the 2019 IEEE 11th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM), Laoag, Philippines, 29 November–1 December 2019; pp. 1–6. [[CrossRef](#)]
6. Sarker, S.; Hoque, M.M. An Intelligent System for Conversion of Bangla Sign Language into Speech. In Proceedings of the 2018 International Conference on Innovations in Science, Engineering and Technology (ICISSET), Chittagong, Bangladesh, 27–28 October 2018; pp. 513–518. [[CrossRef](#)]
7. Ahmed, M.A.; Zaidan, B.B.; Zaidan, A.A.; Salih, M.M.; bin Lakulu, M.M. A Review on Systems-Based Sensory Gloves for Sign Language Recognition State of the Art between 2007 and 2017. *Sensors* **2018**, *18*, 2208. [[CrossRef](#)]
8. Al-Qurishi, M.; Khalid, T.; Souissi, R. Deep Learning for Sign Language Recognition: Current Techniques, Benchmarks, and Open Issues. *IEEE Access* **2021**, *9*, 126917–126951. [[CrossRef](#)]
9. Rao, G.A.; Syamala, K.; Kishore, P.V.V.; Sastry, A.S.C.S. Deep convolutional neural networks for sign language recognition. In Proceedings of the 2018 Conference on Signal Processing And Communication Engineering Systems (SPACES), Vijayawada, India, 4–5 January 2018; pp. 194–197. [[CrossRef](#)]
10. Dutta, N.; Saha, J.; Sarker, F.; Zaman, H.U. A Novel Design of a Multi-DOF Mobile Robotic Helping Hand for Paralyzed Patients. In Proceedings of the 2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI), Bangalore, India, 19–22 September 2018; pp. 2219–2225. [[CrossRef](#)]
11. Shah, F.; Shah, M.S.; Akram, W.; Manzoor, A.; Mahmoud, R.O.; Abdelminaam, D.S. Sign Language Recognition Using Multiple Kernel Learning: A Case Study of Pakistan Sign Language. *IEEE Access* **2021**, *9*, 67548–67558. [[CrossRef](#)]
12. Subedi, B.; Dorji, K.U.; Wangdi, P.; Dorji, T.; Muramatsu, K. Sign Language Translator of Dzongkha Alphabets Using Arduino. In Proceedings of the 2021 Innovations in Power and Advanced Computing Technologies (i-PACT), Kuala Lumpur, Malaysia, 27–29 November 2021; pp. 1–6. [[CrossRef](#)]
13. Kos, A.; Umek, A.; Tomazic, S.; Zhang, Y. Identification and Selection of Sensors Suitable for Integration into Sport Equipment: Smart Golf Club. In Proceedings of the 2016 International Conference on Identification, Information and Knowledge in the Internet of Things (IIKI), Beijing, China, 20–21 October 2016; pp. 128–133. [[CrossRef](#)]
14. Chowdhury, T.H.; Mamun, M.; Islam, A.; Tabassum, S.N.; Karim, T. Verbalink-A Gesture Based Speech Generating Device. In Proceedings of the 2020 11th International Conference on Electrical and Computer Engineering (ICECE), Dhaka, Bangladesh, 17–19 December 2020; pp. 375–378. [[CrossRef](#)]
15. Jani, A.B.; Kotak, N.A.; Roy, A.K. Sensor Based Hand Gesture Recognition System for English Alphabets Used in Sign Language of Deaf-Mute People. In Proceedings of the 2018 IEEE SENSORS, New Delhi, India, 28–31 October 2018; pp. 1–4. [[CrossRef](#)]
16. Faisal, A.A.; Abir, F.F.; Ahmed, M.U. Sensor Dataglove for Real-time Static and Dynamic Hand Gesture Recognition. In Proceedings of the 2021 Joint 10th International Conference on Informatics, Electronics & Vision (ICIEV) and 2021 5th International Conference on Imaging, Vision & Pattern Recognition (icIVPR), Kitakyushu, Japan, 16–20 August 2021; pp. 1–7. [[CrossRef](#)]

17. Silva, B.; Calixto, W.; Furriel, G. Devices analysis and artificial neural network parameters for sign language recognition. In Proceedings of the 2017 CHILEAN Conference on Electrical, Electronics Engineering, Information and Communication Technologies (CHILECON), Pucon, Chile, 18–20 October 2017; pp. 1–5. [\[CrossRef\]](#)
18. Suri, A.; Singh, S.K.; Sharma, R.; Sharma, P.; Garg, N.; Upadhyaya, R. Development of Sign Language using Flex Sensors. In Proceedings of the 2020 International Conference on Smart Electronics and Communication (ICOSEC), Trichy, India, 10–12 September 2020; pp. 102–106. [\[CrossRef\]](#)
19. Mardiyanto, R.; Utomo, M.F.R.; Purwanto, D.; Suryoatmojo, H. Development of hand gesture recognition sensor based on accelerometer and gyroscope for controlling arm of underwater remotely operated robot. In Proceedings of the 2017 International Seminar on Intelligent Technology and Its Applications (ISITIA), Surabaya, Indonesia, 28–29 August 2017; pp. 329–333. [\[CrossRef\]](#)
20. Nyaga, C.N.; Wario, R.D. Sign Language Gesture Recognition through Computer Vision. In Proceedings of the 2018 IST-Africa Week Conference (IST-Africa), Gaborone, Botswana, 9–18 May 2018; Cunningham, P., Cunningham, M., Eds.; Africa: IIMC International Information Management Corporation: Dublin, Ireland, 2018; p. 8.
21. Mummadi, C.K.; Leo, F.P.P.; Verma, K.D.; Kasireddy, S.; Scholl, P.M.; Kempfle, J.; Van Laerhoven, K. Real-Time and Embedded Detection of Hand Gestures with an IMU-Based Glove. *Informatiks* **2018**, *5*, 28. [\[CrossRef\]](#)
22. Deriche, M.; Aliyu, S.O.; Mohandes, M. An Intelligent Arabic Sign Language Recognition System Using a Pair of LMCs With GMM Based Classification. *IEEE Sens. J.* **2019**, *19*, 8067–8078. [\[CrossRef\]](#)
23. Kudrinko, K.; Flavin, E.; Zhu, X.; Li, Q. Wearable Sensor-Based Sign Language Recognition: A Comprehensive Review. *IEEE Rev. Biomed. Eng.* **2020**, *14*, 82–97. [\[CrossRef\]](#)
24. Huang, S.; Ye, Z. Boundary-Adaptive Encoder With Attention Method for Chinese Sign Language Recognition. *IEEE Access* **2021**, *9*, 70948–70960. [\[CrossRef\]](#)
25. Jiang, S.; Lv, B.; Guo, W.; Zhang, C.; Wang, H.; Sheng, X.; Shull, P.B. Feasibility of Wrist-Worn, Real-Time Hand, and Surface Gesture Recognition via sEMG and IMU Sensing. *IEEE Trans. Ind. Inform.* **2017**, *14*, 3376–3385. [\[CrossRef\]](#)
26. Kamal, S.M.; Chen, Y.; Li, S.; Shi, X.; Zheng, J. Technical Approaches to Chinese Sign Language Processing: A Review. *IEEE Access* **2019**, *7*, 96926–96935. [\[CrossRef\]](#)
27. Lee, B.G.; Lee, S.M. Smart Wearable Hand Device for Sign Language Interpretation System With Sensors Fusion. *IEEE Sens. J.* **2017**, *18*, 1224–1232. [\[CrossRef\]](#)
28. Li, L.; Jiang, S.; Shull, P.B.; Gu, G. SkinGest: Artificial skin for gesture recognition via filmy stretchable strain sensors. *Adv. Robot.* **2018**, *32*, 1112–1121. [\[CrossRef\]](#)
29. Muneer, A.-H.; Ghulam, M.; Wadood, A.; Mansour, A.; Mohammed, B.; Tareq, A.; Amine, M.M. Deep Learning-Based Approach for Sign Language Gesture Recognition With Efficient Hand Gesture Representation. *IEEE Access* **2020**, *8*, 192527–192542.
30. Aly, S.; Aly, W. DeepArSLR: A Novel Signer-Independent Deep Learning Framework for Isolated Arabic Sign Language Gestures Recognition. *IEEE Access* **2020**, *8*, 83199–83212. [\[CrossRef\]](#)
31. Kishore, P.; Kumar, D.A.; Sastry, A.C.; Kumar, E.K. Motionlets Matching With Adaptive Kernels for 3-D Indian Sign Language Recognition. *IEEE Sens. J.* **2018**, *18*, 3327–3337. [\[CrossRef\]](#)
32. Maraqa, M.; Abu-Zaiter, R. Recognition of Arabic Sign Language (ArSL) using recurrent neural networks. In Proceedings of the 2008 First International Conference on the Applications of Digital Information and Web Technologies (ICADIWT), Ostrava, Czech Republic, 4–6 August 2008; pp. 478–481. [\[CrossRef\]](#)
33. Adithya, V.; Vinod, P.R.; Gopalakrishnan, U. Artificial neural network based method for Indian sign language recognition. In Proceedings of the 2013 IEEE Conference on Information & Communication Technologies, Thuckalay, India, 11–12 April 2013; pp. 1080–1085. [\[CrossRef\]](#)
34. Akmeiliawati, R.; Ooi, M.P.-L.; Kuang, Y.C. Real-Time Malaysian Sign Language Translation using Colour Segmentation and Neural Network. In Proceedings of the 2007 IEEE Instrumentation & Measurement Technology Conference IMTC 2007, Warsaw, Poland, 1–3 May 2007; pp. 1–6. [\[CrossRef\]](#)
35. Calado, A.; Errico, V.; Saggio, G. Toward the Minimum Number of Wearables to Recognize Signer-Independent Italian Sign Language WITH Machine-Learning Algorithms. *IEEE Trans. Instrum. Meas.* **2021**, *70*, 2513809. [\[CrossRef\]](#)
36. Papastratis, I.; Dimitropoulos, K.; Konstantinidis, D.; Daras, P. Continuous Sign Language Recognition Through Cross-Modal Alignment of Video and Text Embeddings in a Joint-Latent Space. *IEEE Access* **2020**, *8*, 91170–91180. [\[CrossRef\]](#)
37. De Castro, G.Z.; Guerra, R.R.; Guimarães, F.G. Automatic translation of sign language with multi-stream 3D CNN and generation of artificial depth maps. *Expert Syst. Appl.* **2023**, *215*, 119394. [\[CrossRef\]](#)
38. DelPreto, J.; Hughes, J.; D’Aria, M.; de Fazio, M.; Rus, D. A Wearable Smart Glove and Its Application of Pose and Gesture Detection to Sign Language Classification. *IEEE Robot. Autom. Lett.* **2022**, *7*, 10589–10596. [\[CrossRef\]](#)
39. Montemurro, K.; Brentari, D. Emphatic fingerspelling as code-mixing in American Sign Language. *Proc. Linguistic Soc. Am.* **2018**, *3*, 61. [\[CrossRef\]](#)

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.