International Journal of Speech Technology
https://doi.org/10.1007/s10772-025-10184-2

RESEARCH

Bangla sign language translator for deaf and speech impaired people using deep LSTM

Aonmoy Das[1] · Ananna Dev Aishi[1] · Masbah Uddin Toha[1] · Md Fazlul Kader[1]

Received: 12 October 2024 / Accepted: 13 March 2025

Abstract
People with speaking and hearing disabilities use sign language, a non-verbal language, to convey their thoughts to others. As a minority community in Bangladesh, deaf people are experiencing difficulty in getting education and jobs as well as doing everyday chores of life because most people do not understand the sign language, i.e., Bangla sign language (BdSL) they use. This work aims to build a medium that can translate sign words into visible text on a screen, thus helping speech and hearing impaired people bridge the gap and communicate with others. We have built a vision-based system that will be able to track dynamic gestures using Mediapipe holistic and interpret them. The long-short-term memory architecture was used to train and build the deep learning model. The dataset we used in this study was custom-made using a total of 15000 video sequences of 100 sign words from BdSL. The developed model achieved 88.33% test accuracy and a 98.56% area under the curve score. The proposed system can detect the sequence of sign patterns and is set to show them in Bangla font. This sign language translator may be used to learn Bangla sign words as well as act as a medium of communication between the deaf and mute and the rest of society. The outcome of this study suggests that the proposed sign language translator can be a new possibility for human-computer interaction to interact with people using sign language.

Keywords Bangla sign language (BdSL) · Accuracy · Area under the curve (AUC) · Deep learning · Dynamic gesture recognition · Long short term memory (LSTM)

1 Introduction

1.1 Theoretical background

People who are not capable of hearing or speaking cannot interact with other people through verbal language. People with speech and hearing impairments use sign language as their communication medium. Sign language is a composition of hand movements incorporating visual motions and gestures. Various parts of the body, such as the fingers, hands, arms, head, body, and facial expressions are used by speech impaired people to express thoughts and sayings in sign language (Cheok et al., 2019). The number of parameters involved in building an accurate sign word is five, which are the shape of the hand, the orientation of the palm, motion, position, and expression (Rastgoo et al., 2021). As a unique and complete language, it has its own set of grammar and forming structures.

There are more than 200 sign languages in the world, as per the World Federation of the Deaf, and every sign language is different from the other. This variety in sign language makes it difficult for people who do not use the specific sign library to understand. In addition, learning different sign languages is a difficult task that involves complexity and is almost impossible for common people. A sign language translation system can help to improve the situation by breaking the social barrier for the speech impaired. There are many advantages of the sign language recognition system in real-world applications, such as service of interpretation, human-computer interaction (Supančič et al., 2018), recognition system of real-time multi-person (Cao et

✉ Md Fazlul Kader
f.kader@cu.ac.bd

Aonmoy Das
19702037@std.cu.ac.bd

Ananna Dev Aishi
aishi.eee@std.cu.ac.bd

Masbah Uddin Toha
masbahuddintoha@gmail.com

[1] University of Chittagong, Chittagong, Bangladesh

Published online: 10 April 2025

al., 2017), games, environment of virtual reality, controlling of robots, etc. Furthermore, according to the World Health Organization, over 430 million people in the world have hearing problems, and the number will be 2.5 billion by 2050. So, there is growing interest in removing communication barriers that exist between the hearing-impaired community and other people who have normal hearing due to the large population of hearing-impaired people.

Sign language is a composite of dynamic hand gestures and poses that a user makes. Tracking those gestures in a sequential manner is the main challenge in building a sign language recognition system. There are also some environmental factors present in the sign language recognition system. While using a vision-based method for translation, many unwanted parameters like skin color, light intensity, and other objects can pass detection. So, our goal was to eliminate these unwanted factors and only work with the parameters we needed to recognize hand movement. Hand pose estimation based on Azure kinect (Zhu et al., 2021) can be a good solution for this. The effect of gestures on narrative speech in terms of fluency and prosodic features is studied in (Cravotta et al., 2021). Moreover, the MediaPipe holistic, which is developed by Google, is being used for human action tracking with high accuracy. In the context of real time action capture, a sign comprises several frames. By analyzing those actions in multiple frames, a sign can be recognized and interpreted. Different machine learning and deep learning techniques are being used as classifiers in health informatics, which includes human activity recognition for predicting brain diseases (Khan et al., 2021). For traditional sensor-based human activity recognition, the AdaBoost algorithm has shown impressive performance when implemented with k-nearest neighbors (KNN) (Pires et al., 2021). However, in vision-based methods, deep learning techniques like convolutional neural networks (CNNs), recurrent neural networks (RNNs), etc. are the go–to solution as they do not require separate feature extraction. The long short-term memory (LSTM) structure, which is a variant of an RNN, can analyze frame-by-frame changes in the video sequences (Van Houdt et al., 2020).

In recent years, artificial intelligence and robotics have been utilized to improve the flexibility of people with impairments and difficulties (Barche et al., 2024; Hashan et al., 2024). The major purpose in such cases is to enhance the standard of living by allowing the users to conduct their daily essential tasks more conveniently. For example, for the ease of daily communication, various technologies such as voice calling, video calling, and video conferencing have been developed. Only people who use spoken language can take advantage of these technological tools. Keeping this in mind, we set our goal to enable Bangladeshi hearing-impaired people to take advantage of these newest technologies.

Sign language is a structured form of hand movement, facial expression, and body language. Like any other spoken language, they are also well developed and complex. Sometimes a sign can represent multiple words, but usually one word per sign. Also, there are fingerspelling signs. These signs are used to spell words. Sign language is essentially a visual transmission of sign patterns. Each pattern conveys a different meaning. These patterns change when the locality changes, just like in verbal languages. People who are not familiar with these signs cannot communicate with the deaf people of such localities. So, an interpreter is always needed. There are many popular sign languages in the world.

American sign language (ASL) is a complete and organized visual language with linguistic features similar to those of a spoken language. But it is quite different from the English language. Many deaf and hard-of-hearing North Americans use ASL, including many hearing people (American-sign-language, n.d.). ASL has its own patterns of expressing thoughts. It is expressed through hand and facial movements. There was a mixing of local sign languages and French sign language in the making of ASL. Although modern ASL and modern French sign language are totally different languages, each other's users do not understand.

British sign language (BSL) is the most used form of sign language in Britain. In 2003, BSL was finally recognized by the UK government as a minority language in Britain. This sign language is used by over 145,000 people in the UK (What is british sign language, n.d.). Another form of BSL is available and is known as sign-supported English. It has the same sign as the BSL but uses the order of natively spoken English. This helps students with hearing impairments learn English grammar as well as the signs of these impairments. BSL is not a universal language like English. It is quite different from other sign languages. Sometimes there is significant variation in BSL from city to city within Britain, which is similar to a regional accent.

The sign language used in Bangladesh is known as "Bangla Ishara Vasha" among the local community. Deaf and mute people in the country use Bangladeshi sign language as their primary medium of communication. There is a distinct set of rules and gestures for the language. A Bangla Sign Language dictionary was published in 1994 by the collaboration of the Ministry of Social Welfare and Bangladesh National Federation of the Deaf (Committee, 1994). Every region has a different set of structures for their own sign language, e.g., Bangla sign language (BdSL) is different from ASL. So, it creates a communication gap between specific sign language users and people who do not understand them. For example, if a speech impaired person tries

to discuss a matter in an online meeting, only people who know the signs will understand him. In Bangladesh, speech impaired people are having trouble getting education and proper jobs. A sign language translation system can assist in solving the problems. There is a lot of research going on for translation systems of popular sign languages, e.g., ASL and BSL, but BdSL is lagging behind in this field.

To address the Bangladeshi sign language recognition problem, we have presented an efficient deep learning model for tracking and recognizing sign words combining MediaPipe holistic (MediaPipe Holistic, n.d.) and LSTM units. A total of 100 sign words have been collected from the "Bangla Sign Language Dictionary" to build a new dataset. In addition, we have developed a system to implement the Bangla sign language translation system in real-time.

1.2 MediaPipe

We used the MediaPipe holistic (MediaPipe Holistic, n.d.) for the detection of poses and hands of the signer. A MediaPipe holistic tracking example is shown in Fig. 1. The extraction of critical points was done simultaneously. In the coding environment, we used the MediaPipe holistic library with an OpenCV interface while collecting the video sequences. The preprocessing step is automated using this library. In the data collection and detection step, the MediaPipe plays an important role in dispelling the need to use CNN. This enables the model to be trained faster than most of the existing sign language recognition models. Because many unwanted parameters, such as lighting conditions, human skin, background, etc. are removed from the model training parameters. Thus MediaPipe improves the model's efficiency. Again, while implementing the model on the user end, this library is needed. The initial process of recognition will be done by the MediaPipe holistic. It is lightweight and optimized for mobile devices. So, the implementation of the library did not create any problems, and the optimum performance will be ensured.

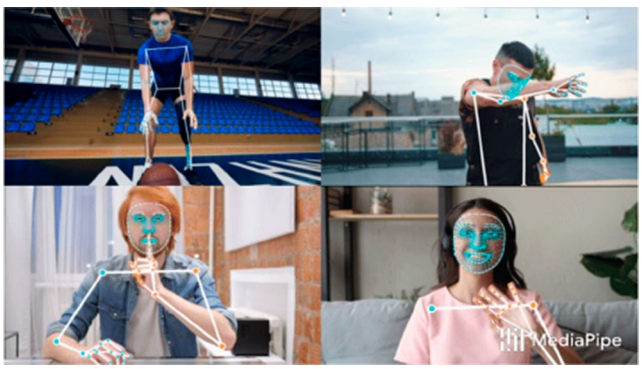

Fig. 1 MediaPipe holistic tracking

1.3 Paper organization

The remainder of the paper is carried out as follows: previous works on sign language translation using various techniques have been described in Sect. 2. The proposed framework for the Bangla sign language translation has been explained in detail in Sect. 3. Experimental results and comparisons with existing works are reported in Sect. 4. Finally, Sect. 5 concludes the paper.

2 Related works

Many studies started two decades ago on the automatic sign language recognition process (Aly et al., 2019). These works are designed specifically for sign languages such as American, Chinese, Arabic, Polish, British, and others. Many techniques have been developed and tested based on different modalities, sensors, machine learning techniques, recognition of sign structures, image analysis, etc. However, glove-based or sensor-based and vision-based approaches are two major categories for automatic sign language recognition.

2.1 Sensor based methods

In sensor based methods, sign gestures are recorded through physical sensors or devices while communicating using sign language. A recognition system then makes use of these recorded activities and decodes them using a predefined algorithm. The glove-based models need to make use of several mechanical or optical sensors and electronic circuits present within a glove. This setup enables the model to use electrical signals for hand gesture detection. The main benefit of this technique is that it does not require complex data processing. Also, the user does not have to worry about the lighting conditions or surroundings. On the other hand, vision-based approaches need captured video data of the target. By tracking its motion and classifying it, the system recognizes the specific sign. It is more natural and user friendly because users don't have to wear gloves to be able to translate the sign words. Environmental factors such as lighting, camera location, background noise, etc. can affect the accuracy of the detection. Kudrinko et al. (Kudrinko et al., 2021) showed various aspects of sensor-based sign language recognition. Different types of sensors, such as surface electromyography, pressure, strain, and inertial sensors are used in these types of approaches. Hybrid systems, which consist of both computer vision and sensor-based systems, were also described. This system can be a solution to the problem caused by environmental factors in vision-based models and can achieve a high recognition rate. An intelligent electronic

glove system was presented by Montalvo et al. (Rosero-Montalvo et al., 2018). The system is able to detect the sign information of numbers from 1 to 10 in SL. The glove has built-in flex sensors in each finger, which are used to collect data. Data balancing was done with the Kennard-Stone and KNN as classifiers. W. Aly et al. (Aly et al., 2019) proposed a novel method using depth images captured by the Kinect depth sensor. The system was built to recognize the ASL alphabet. The feature was extracted from the segmentation region using the principal component analysis network. The model was developed using the support vector machine (SVM) as a classifier.

2.2 Vision-based methods

Computer-vision based solution for ASL recognition was proposed by Bantupalli et al. (Bantupalli & Xie, 2018). Video sequences were taken for extracting temporal and spatial features. Inception was used for recognizing spatial features, and RNN was used to train on temporal features. The dataset consists of 600 training samples of 300 frames. The model faced detection losses due to facial features and skin tones. Raj et al. (RAJ & Jasuja, 2018) used histograms of oriented gradients (HOG) and artificial neural networks (ANN) for BSL recognition. A camera captures the hand movements and passes them to the system. The system then extracts the HOG features, and ANN classifies the sign. A novel Arabic sign language recognition technique was presented by S. Aly et al. (Aly & Aly, 2020). DeepLabv3 + was trained using input video of hand motion. A set of pixel-labeled hand images was used to extract the hand region from the video using. The feature was extracted by using a single layer convolutional self-organizing map. The model was then trained using deep Bi-directional LSTM. Three distinct users waved 23 isolated words to train the system. M. Rahman (Rahaman et al., 2020) developed a model that is able to automatically recognize hand-sign-spelled Bangla sign language. A two-step classifier was used in the first phase for the Bangla language modeling algorithm. All "hidden characters" were discovered based on recognizing characters' from 52 hand signs of BdSL. S. Islam et al. (Islalm et al., 2019) presented a large dataset of BdSL. The static dataset consists of 10 numerals and 35 alphabets. CNN was used to build and test the dataset. Sanzidul et al. (Sanzidul Islam et al., 2018) proposed another dataset of BdSL named 'Ishara Lipi'. 50 sets of 36 Bangla sign characters were collected with the help of various speech impaired people to develop the dataset. The CNN method was used to extract features from 1800 images of characters. A vision-based system for automatic sign language recognition can be developed using the dataset. An approach for detecting BdSL in real time was presented in (Urmee et al., 2019). 2000 augmented images of 37 different BdSL characters sign classes were used to build a dataset named 'BdSLInfinite'. Xception architecture was used in CNN to develop the model. The average detection time for the model on the test set was 48.53 ms. In (Shanta et al., 2018), it is shown that for BdSL detection, scale-invariant feature transform as feature extraction technique in input and CNN works well. But the model has some limitations, such as, it could not detect two handed gestures, and illumination will affect it while using in real time recognition. Principal component analysis and KNN were used to detect two-handed BdSL in (Haque et al., 2019). With different backgrounds and lighting, the model is said to be efficient. The problem with the approach is that the model cannot be used for dynamic hand signs and real-time applications. Sadik et al. (Sadik et al., 2019) proposed a method for recognizing BdSL with skin segmentation and binary masking. YCbCr color space was used to preprocess the images of the sign samples. The model was trained using a multiclass SVM. The model is also not suitable for using a real-time application, and the background can lead to detection errors. Hoque et al. (Hoque et al., 2018) represented a technique of detecting BdSL in real-time using faster region based CNN. The system has limitations in detecting sign characters with similar patterns, and the data training requires a huge amount of time. M. Islam et al. (Islam et al., 2022) employed four popular transfer learning methods, such as VGG16, VGG19, InceptionV3, and AlexNet, on a static sign image dataset to recognize 11 common Bengali sign words. These methods outperform the conventional CNN methods, and among these methods, VGG16 claimed to show comparatively better performance (99.92% training accuracy and 92.42% testing accuracy). But the problem with the method is that it cannot work with Bengali sign words consisting of dynamic sign gestures in a real-time scenario. So, as we can see, some researchers developed some work in the sector of BdSL recognition. But most of the recognition processes are not able to cope with real-time applications. Also, the word-level sign of the BdSL has some complex sets of hand gestures. In this regard, BdSL is lagging behind.

2.3 Long short-term memory (LSTM) in sign language recognition

Gesture recognition related computer vision problems contend with the difficulty of representing short-term and long-term sequences as well as temporal dependencies among inputs. CNNs are excellent for one-to-one mapping but cannot process a sequence of vectors. Sequence learning method such as RNN are utilized in this scenario. RNNs put more of an emphasis on altering state neurons than traditional neural networks to discover contextual relationships

inside and between sequential data (Lyu & Huang, 2018). LSTM architecture is a RNN variant that has emerged as a powerful and scalable method for solving problems involving sequential data (Greff et al., 2016). LSTMs are particularly well-suited for tasks that involve modeling long-term dependencies in sequential data. LSTMs have relative intensity gaps, which give them an advantage over competing sequence learning models like hidden Markov models and other RNNs (Kang et al., 2021).

In our proposed work, LSTM is well-suited because sign language is a sequential data modality, with gestures occurring one after the other in time. An LSTM is able to remember the context of previous gestures and use that information to accurately predict the next gesture, which is difficult for traditional models to do. In addition, LSTMs are able to process data in real-time, which is important for a sign language recognition system because it uses a camera, as the system would need to be able to process the input data and make predictions quickly in order to keep up with the user's movements. LSTMs are relatively robust to noise and can handle variations in the input data, which is important in a real-time sign language recognition system that uses a camera. Cameras can be prone to noise and variations in lighting, which could affect the quality of the input data. An LSTM tends to handle these variations better than traditional methods.

2.4 Contributions

The contributions of this work in the field of sign language translation are summarized below:

- Based on the MediaPipe holistic and LSTM networks, a novel framework for word level BdSL dynamic sign gesture detection is proposed.
- A BdSL dataset is created. The data set can be used for further research purposes.
- A system for dynamic gesture recognition models has been developed that can translate word-level sign structures. A deaf or mute person can express their thoughts using the system.
- Most importantly, the system will show its prediction of Bengali sign words in Bangla fonts. It will help the common Bengali speaking people to easily understand the translated signs.

3 Methodology

The proposed framework comprises seven different stages to build the model for the detection of Bangla sign words. To begin the process, we first need to look into the 'Bangla Sign Language Dictionary' for the desired sign words. We chose 100 commonly used Bangla sign words from the dictionary in order to create a dataset. Those sign words were then practiced several times before starting the process. This is necessary to ensure the signs are waved correctly while being captured on the camera. The deep learning model development process is summarized in the Fig. 2. The 7 phases of the recognition model are described below.

1. The most important part of this work is to create a reliable dataset of sign patterns. To ensure this, the iPhone 15 Pro Max's 12 megapixel front camera is used to collect sequences of the 100 patterns. 5 videos were collected for a particular sign, thirty video sequences were

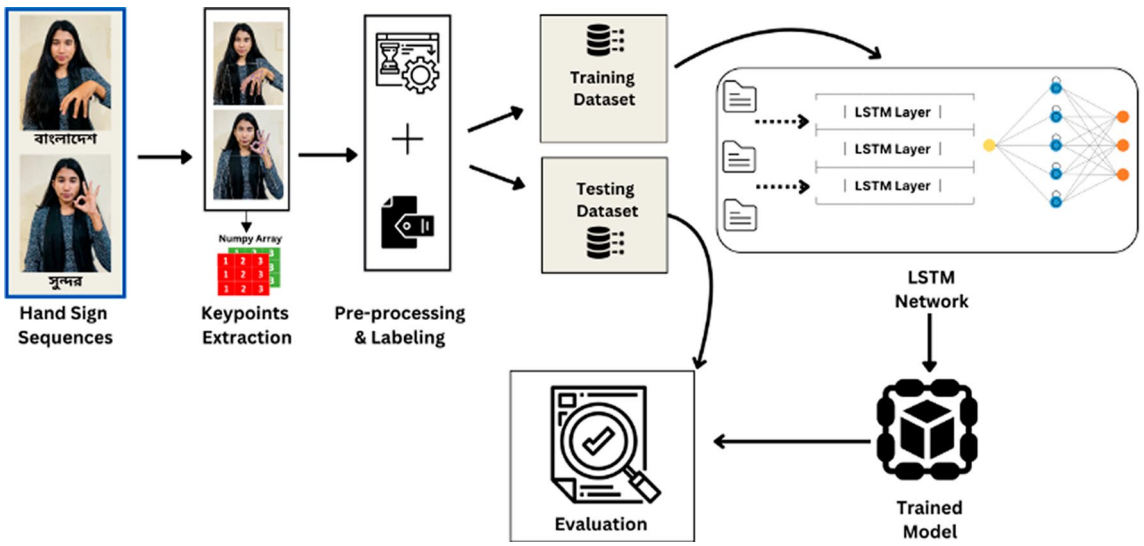

Fig. 2 Proposed methodology

collected for a particular video. Every video sequence was 30 frames long. These sequences were saved in a labeled folder, named after the sign and the corresponding video number.

2. The sequences were fed into the OpenCV interface to be able to read and work with them. Hand landmarks and poses were detected in those sequences using MediaPipe holistic. The key points were extracted. Those key-points of such sequences are saved as numpy arrays. This step removes the unwanted features that are not related to the sign sequences. The numpy array of the sequences was then saved in a labeled folder, named after the sign and the corresponding video number.
3. Different things and words have different signs. These sign sequences are to be distinguished by their names, which are already named in the initial folders. But we needed to label the numpy array of the images, which contains key-point information. All the frames of a single sign pattern were aligned into a folder and labeling was done through the code environment. All the sign patterns were labeled in a similar way. Automatic preprocessing was done while collecting the key-points using MediaPipe holistic.
4. After the preprocessing and labeling, the key-points of the video sequences were stored in a single folder. The stored folder contains the required dataset for the model. For training and validation purposes, the dataset was randomly split into 3 separate datasets. 80 percent of the data was set to serve as the training dataset, 10 percent as the validation dataset and the remaining was used as the testing dataset. This process was done using the 'train test split' function in the coding environment.
5. LSTM architecture was used for building the sequential deep learning network. The Tensorflow library and Keras application programming interface were used in the coding environment for importing the necessary components of the sequential network.
6. The training dataset is fed into a sequential LSTM structure. Depending on the data size, the epoch number was set to 500. After the training process, we got a trained model (weight file), which was later used to make predictions.
7. Before using the trained model in real-time testing, evaluation of the trained model must be done. The test dataset was used in a function that evaluated the accuracy, confusion matrix, and area under the curve (AUC). Precision, recall, F1 score and support were calculated for each class.

3.1 Sign gesture recognition

In this stage, the train dataset, which contains cleaned and labeled sign gestures, is fed to our deep LSTM model (Yu et al., 2019) for sign recognition. We have built a sequential deep learning network that employs the LSTM units. The sequential model was built in Keras with four LSTM layers and three fully connected (dense) layers.

3.2 BdSL dataset

A dataset for the BdSL recognition model was developed in this work. For the deep learning process, a large amount of data with relevant labels is needed. It is crucial for any classification task that the solution is accurate enough to produce performance characteristics. The thing that makes a deep learning model accurate is a good number of tagged datasets. For the dynamic gesture classification task, we must use a sequence dataset. The dataset must be carefully labeled and processed so that unwanted features do not interfere. So, as we can see, there are several steps to making the dataset mentioned at the start of the section. We needed to ensure those steps were strictly followed. The dataset (BdSL-LSTM dataset containing 100 Bangla Sign Words, 2025) used in the development of the recognition model contains 100 sign words. Table 1 represents the recorded BdSL words with their English meanings. For each of the sign words, we captured 5 videos, each video containing 30 video sequences with different possible orientations and backgrounds. Each video sequence contains 30 frames. So, a total of 15000 video sequences and 45,0000 frames were captured for all sign words. There are 100 labeled folders containing 5 folders each for the different videos each containing 30 more folders for sequence key points. Fig. 3a, Fig. 3b, Fig. 3c, and Fig. 3d show the sequences of the Bangla sign words that are mentioned in Table 1. Those sign words were carefully collected and labeled using the experimental setup as described in the following subsection. The data set consists of a numpy array of keypoints instead of a conventional video or image. Those key points were extracted using the MediaPipe holistic library from the video sequences of selected sign words. This process drastically reduces the storage size of the dataset.

3.3 Model architecture

Our proposed model for detecting Bangla sign language words consists of a deep LSTM-based neural network designed to process sequences of 30 frames with 126 landmarks per step. We use four stacked LSTM layers with tanh activation to capture temporal dependencies, with the final LSTM output passing through fully connected layers

Table 1 The list of 100 sign words that are used in developing the dataset

SL	Bangla sign words	Equivalent English meaning	SL	Bangla sign words	Equivalent English meaning	SL	Bangla sign words	Equivalent English meaning	SL	Bangla sign words	Equivalent English meaning
1	আম	Mango	26	ঘি	Ghee	51	আইন	Law	76	শত্রু	Enemy
2	বই	Book	27	লেখা	Write	52	আকাশ	Sky	77	সাপ	Snake
3	পানি পান করা	Drinking water	28	পরে	Later	53	আচ্ছা	Okay	78	স্কাউট	Scout
4	কলম	Pen	29	খেলা করা	Play	54	উন্নতি	Improvement	79	হেলিকপ্টার	Helicopter
5	টেবিল	Table	30	খাটো	Short	55	কুরআন	Quran	80	আমরা	We
6	কাঁধ	Shoulder	31	গান করা	Sing	56	কুরবানি	Sacrifice	81	ক্যামেরা	Camera
7	চামড়া	Skin	32	ইচ্ছা করা	Wish	57	ক্ষমা চাওয়া	Apologize	82	গাড়ি	Car
8	ধন্যবাদ	Thank you	33	শোনা	Listen	58	খাওয়া	Eat	83	ঘুমানো	Sleep
9	আগামীকাল	Tomorrow	34	চাওয়া	Want	59	গরু	Cow	84	টেলিফোন	Telephone
10	অসুস্থ	Sick	35	লাফ দেয়া	Jump	60	গ্রেফতার	Arrest	85	তুমি	You
11	হাত	Hand	36	বাংলাদেশ	Bangladesh	61	চশমা	Glasses	86	দাও	Give
12	চিন্তা	Thoughts	37	স্কুল	School	62	জানা	Know	87	না	No
13	মেয়ে	Girl	38	লম্বা	Tall	63	জেল	Prison	88	নাম	Name
14	দাদা	Grandfather	39	নিশ্চিত	Sure	64	দরিদ্র	Poor	89	নামায	Namaz
15	শিক্ষক	Teacher	40	বিশ্বাস	Faith	65	ঘড়ি	Watch	90	ফুল	Flower
16	পড়া	Study	41	শহর	City	66	দাঁড়াও	Stand Up	91	চিৎকার করা	Shout
17	মানুষ	Human	42	মসজিদ	Mosque	67	পুলিশ	Police	92	বন্ধু	Friend
18	ভাই	Brother	43	হাসপাতাল	Hospital	68	প্রশ্ন	Question	93	বাবা	Father
19	চাচা	Uncle	44	নাচা	Dance	69	বাঁশি	Flute	94	বিমান	Airplane
20	চাল	Rice	45	শিশু	Child	70	বালক	Boy	95	ব্যায়াম	Exercise
21	ডিম	Egg	46	টাকা	Money	71	বাড়ি	Home	96	ভালো	Good
22	ঝাল	Spicy	47	পৃথিবী	Earth	72	বোকা	Fool	97	মা	Mother
23	চা	Tea	48	চেয়ার	Chair	73	বড়	Big	98	যুদ্ধ	War
24	ফল	Fruit	49	সাগর	Sea	74	সঞ্চয়	Savings	99	সমর্থন	Support
25	সবুজ	Green	50	ধাক্কা দেয়া	Push	75	উপদেশ দেয়া	Advise	100	সুন্দর	Beautiful

for classification. To enhance performance, we incorporate batch normalization for stable training and dropout to reduce overfitting. The final dense layer, with softmax activation, classifies inputs into one of 100 possible Bangla sign language words. With approximately 4.97 million trainable parameters, our model is optimized using the Adam optimizer. Figure 4 and Table 2 illustrate the architecture and parameters of the proposed model.

3.4 Experimental setup

To train and analyze the deep learning model, we needed to do several tasks as mentioned earlier. For this, we built a system that is capable of running the whole process efficiently. The setup was divided into two different categories, i.e., hardware setup and code environment setup. To meet the computational power requirement, we built a machine with an Intel i5-9400 as the central processing unit, an Nvidia 1650 super as the graphics processing unit, and 8.00GB of random access memory on a Windows 10 operating system. We needed a good camera in the system for collecting the required video sequences for the dataset. Every hand gesture has to be captured clearly. The key points we are trying to collect for training purposes will not be accurate if the camera does not collect all of those key points. For the data collection, we used the 12-megapixel iPhone 15 Pro Max front camera. The camera was set to give us a 1280 pixel by 720 pixel resolution. However, for real-time

Fig. 3 **a** Tracked sequences of BdSL sign words (1–30) using MediaPipe holistic in the data collection phase. **b** Tracked sequences of BdSL sign words (31–60) using MediaPipe holistic in the data collection phase. **c** Tracked sequences of BdSL sign words (61–90) using MediaPipe holistic in the data collection phase. **d** Tracked sequences of BdSL sign words (91–100) using MediaPipe holistic in the data collection phase

use, any normal camera, even a 640 × 480 pixel camera, is good to go. We used the Logitech c170 webcam model to test the model in real time. The webcam was able to support the system pretty well, and in good lighting conditions,

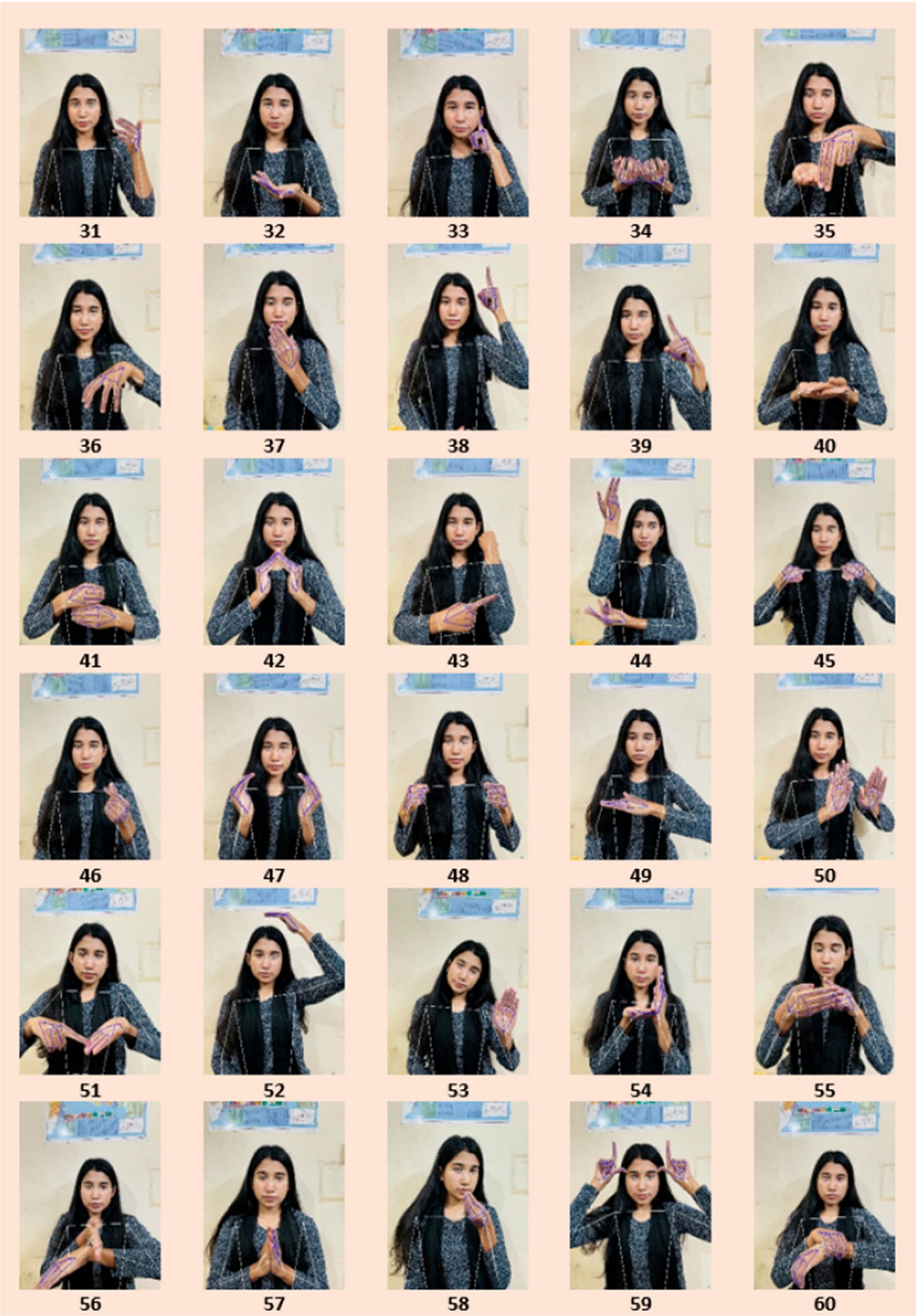

Fig. 3 (continued)

the performance was top-notch. For code environment setup, we used Jupyter Notebook as an integrated development environment. Tensorflow 2.18.0, MediaPipe Holistic, OpenCV 4.5.5, Scikit Learn, etc. were used for importing and implementing necessary functions, and Keras was in the backend. The Python programming language was used for developing the pipeline for training and testing purposes.

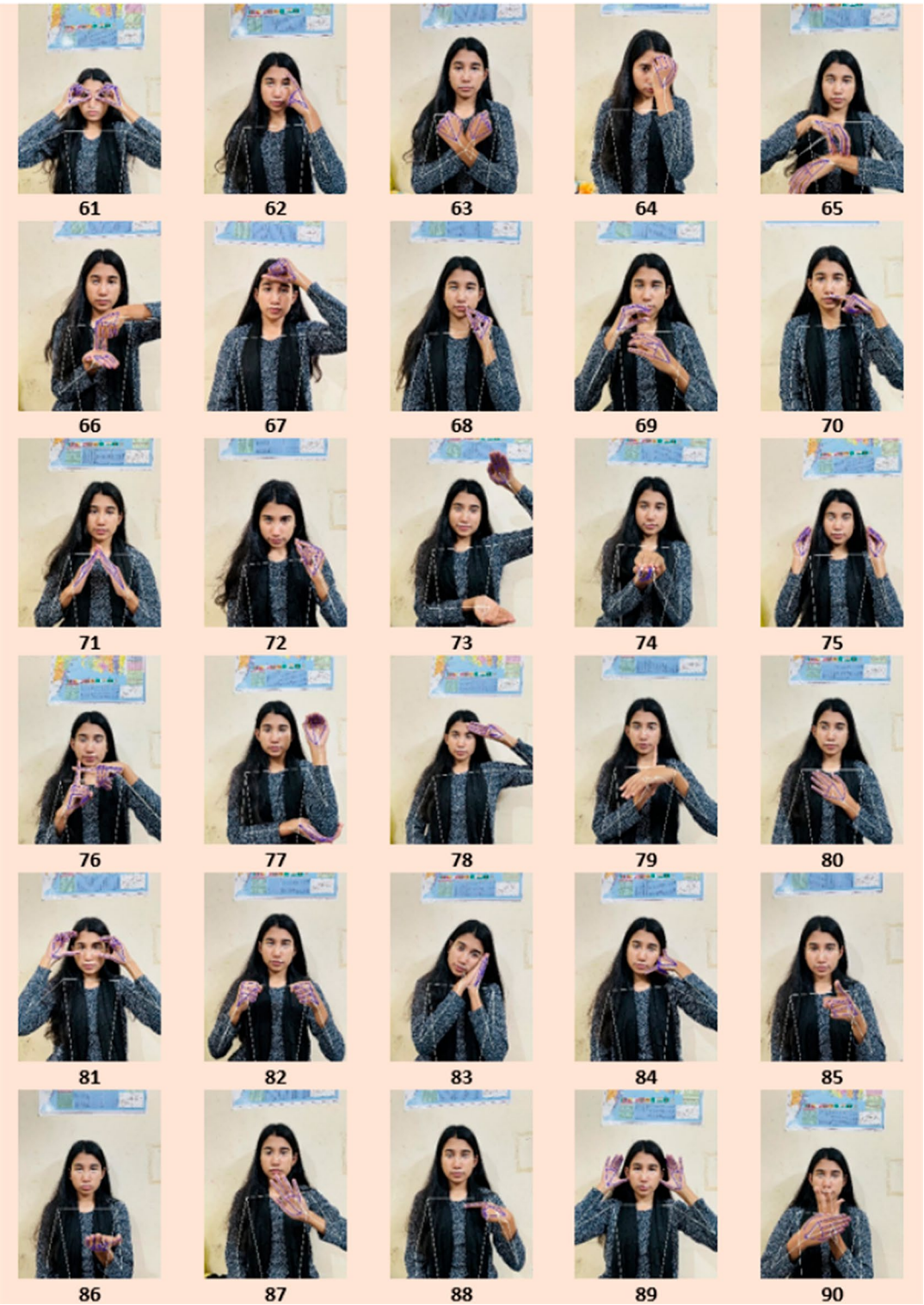

Fig. 3 (continued)

3.5 Model implementation

The important part of the system is the real-time detection of Bangla sign words. Several steps have to be followed for the recognition and translation of a waved sign. The architecture of the real-time translation process using the system is summarized in the Fig. 5. The steps are described below.

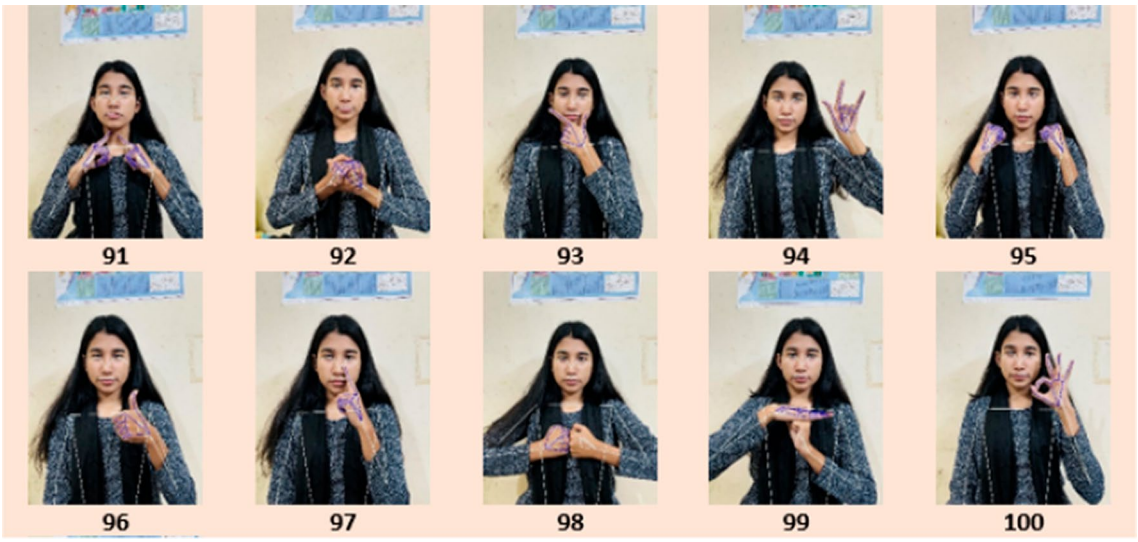

Fig. 3 (continued)

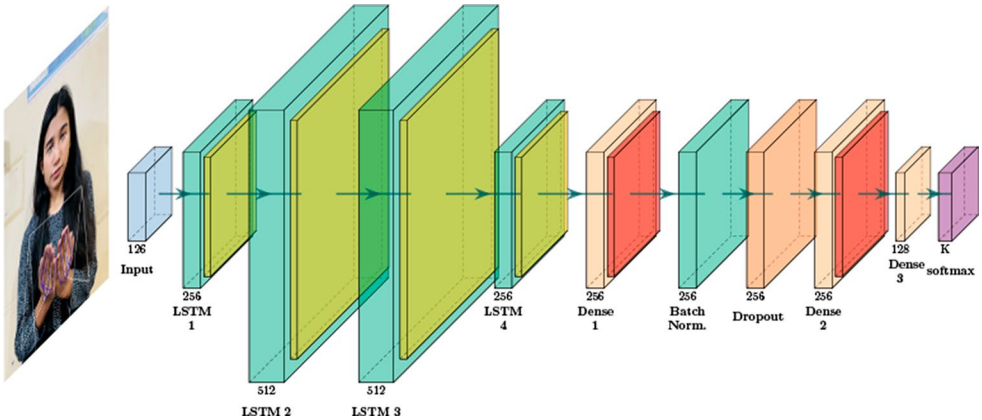

Fig. 4 Architecture of the proposed model

Table 2 The architecture of the proposed model

Layer (Type)	Output Shape	Param #
LSTM 1 (LSTM)	(None, 30, 256)	392,192
LSTM 2 (LSTM)	(None, 30, 512)	1,574,912
LSTM 3 (LSTM)	(None, 30, 512)	2,099,200
LSTM 4 (LSTM)	(None, 256)	787,456
Dense 1 (Dense)	(None, 256)	65,792
Batch Norm. (BatchNormalization)	(None, 256)	1024
Dropout (Dropout)	(None, 256)	0
Dense 2 (Dense)	(None, 128)	32,896
Dense 3 (Dense)	(None, 100)	12,900
Total params:		14,898,094 (56.83 MB)
Trainable params:		4,965,860 (18.94 MB)
Non-trainable params:		512 (2.00 KB)
Optimizer params:		9,931,722 (37.89 MB)

- A user will wave a sign.
- A webcam will capture those sign patterns sequentially.
- Those frames will go through the OpenCV interface. The sign patterns will be detected by estimating hand and posing landmarks. And the key points will be extracted.
- The key points will be fed into the system pipeline to compare with the trained model.
- Certain words will be predicted sequentially as the waved sign.
- Words will be displayed on the user's screen in Bangla font.

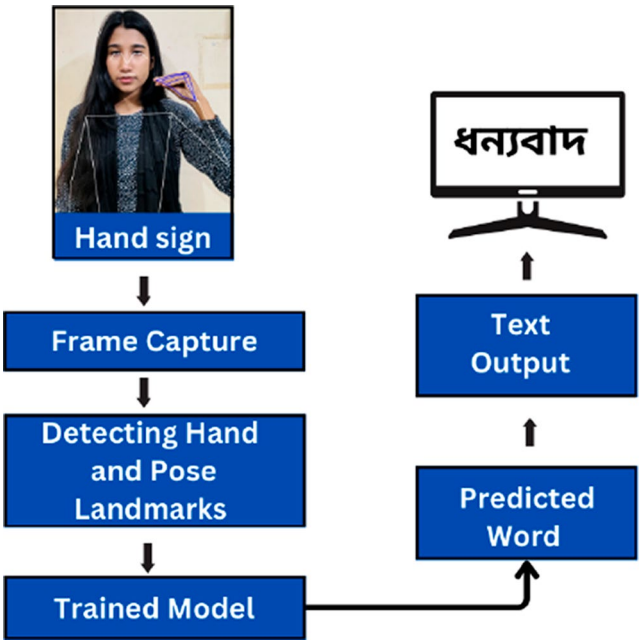

Fig. 5 System implementation in real-time

4 Results and analysis

This section describes experimental results and quantitative analysis of our deep learning model and comparison with other models in terms of performance. It also covers the performance metrics that is used in the experiment.

4.1 Evaluation metrics

Evaluation metrics are used to assess the model's ability to make accurate predictions on new data and to compare the performance of different models. It is important to choose an appropriate evaluation metric for the specific problem being addressed, as different metrics are sensitive to different aspects of model performance. The performance evaluation metrics used for our deep learning model are described below:

4.1.1 Accuracy

Accuracy is a common performance metric that is used to evaluate the effectiveness of a machine learning model. It is defined as the fraction of correct predictions made by the model out of all predictions made and is represented by (1). In the case of a sign language translation model, accuracy refers to the fraction of signs that the model correctly translates into the corresponding text or spoken language.

$$\text{Accuracy} = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}}. \tag{1}$$

4.1.2 Confusion matrix

A confusion matrix is a table that is used to describe the performance of a classification model (or "classifier") on a set of test data for which the true values are known. True negatives (TN) are the number of cases where the classifier correctly predicted the negative class. False positives (FP) are the number of cases where the classifier predicted the incorrect class. False negatives (FN) are the number of cases where the classifier failed to predict the correct class. True positives (TP) are the number of cases where the classifier predicts the correct class.

4.1.3 AUC

AUC is a performance metric used to evaluate the effectiveness of a binary classification model. AUC measures the model's ability to distinguish between positive and negative examples. The equation for calculating the area under the curve is given by

$$AUC = \int_{-\infty}^{\infty} TPR(FPR)dFPR. \tag{2}$$

Equation (2) represents the integral of the true positive rate (TPR) as a function of the false positive rate (FPR) over the entire range of possible FPR values. The AUC ranges from 0 to 1, with a value of 0.5 representing a classifier that performs no better than random guessing and a value of 1 representing a perfect classifier. AUC is a popular metric because it is independent of the classification threshold and is therefore not sensitive to the choice of the threshold.

4.2 Quantitative analysis

The BdSL dataset we developed in this experiment contains 1500 video sequences (450000 frames) of 100 dynamic hand signs. The dataset was split into training (80%), validation (10%) and test (10%) datasets for the experiment. The process was set to have 500 epochs with a batch size of 32. After the training process was completed, the experimental results showed that the model had 89.40% training, 85.47% validation and 88.33% test accuracy. Figure 6 shows the variation of accuracy score on training, validation and test datasets over 500 epochs. The AUC score was 0.9856 on test dataset, which is shown in Fig. 7 and so the model is good enough to distinguish between almost all positive and negative classes. Figure 8 represents a 100 × 100 confusion matrix of the model, which consists of hundred sign

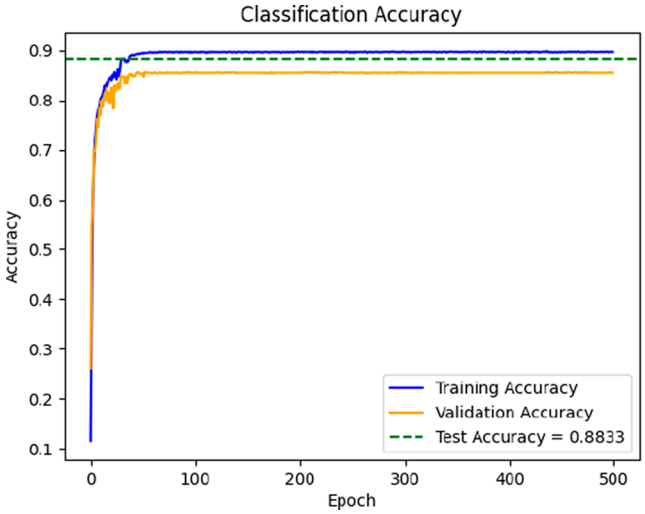

Fig. 6 Training, validation and test accuracy of the model

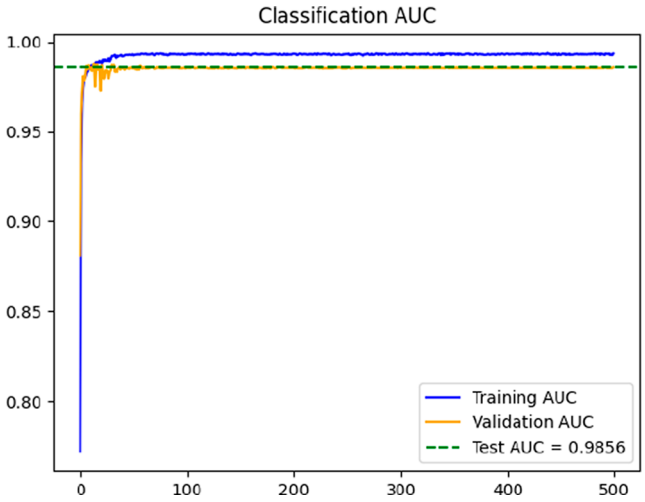

Fig. 7 AUC vs Epoch score of the model

words. The rows of the matrix render the prediction score of a predicted word, and the columns are there to portray the prediction score of an actual class. From the matrix, we can see the individual true positive prediction rate of each class. Each class represents a distinct set of sign words from the "Bangla Sign Language Dictionary", which are sequentially presented in Table 1.

4.2.1 Classification report

To evaluate the ability of the model to correctly classify different signs, the precision, recall, F1 score and support was calculated for each of the 100 signs. The full classification report is illustrated in Table 3

4.3 Comparative analysis

Table 4 shows a comparison among different models as well as our proposed model in terms of performance. Sadik et al. (Sadik et al., 2019) used SVM for developing a BdSL recognition model without dynamic background. Therefore, the algorithm can only be used for static hand signs and is not applicable in real-time applications. Rahaman et al. (Rahaman et al., 2020) worked with static images of alphabets with different backgrounds that can interpret hand signs spelled BdSL into text words. Dipon Talukder et al. (Talukder & Jahara, 2020) used YOLOv4 to develop a similar model that can also translate finger-spelled BdSL sign characters and generate words and eventually sentences. Both models achieve good accuracy, but a user has to spell all the words while using them in real time, which is not practically usable on the user's end. Another work was done by Shafiqul Islalm et al. (Islalm et al., 2019) which involves the recognition of Bangla basic characters and numerals. It achieved 99.80% accuracy for combined uses of characters and numerals. Similarly, Kanchon et al. (Podder et al., 2020) used a different combination of techniques to develop a working model that can detect the Bangla sign alphabet, whereas (Podder et al., 2022) identifies both the Bangla sign alphabet and numerals. However, in our proposed method, we have shown Bangla sign word detection, which is much more convenient than the outcome of the alphabet and numeric detection in real-time applications. Paromita Urmee et al. (Urmee et al., 2019) emphasized the speed of response while recognizing the sign gestures with Xception augmentation. A user has to position his hand within a certain region so that they can detect the hand sign of a certain BdSL character. As we can see, the recent research in this field is overwhelmed with BdSL sign character recognition. For example, character level and finger spelled signs are helpful for the learning purpose of certain sign languages. At the user end, word-level sign recognition is much more convenient and useful while communicating in daily life. M. Islam et al. (Islam et al., 2022) perceived the need for word-level BdSL research and proposed transfer learning techniques to detect 11 BdSL sign words. The performance of the techniques mainly depends on the static image processing, and this method is not yet ready to be deployed in real-time applications. As we used dynamic gesture patterns of Bangla sign words to develop the model, it can detect the moving hand sign sequences and translate them into text in real-time. Our proposed LSTM-based model for word-level Bangla sign language translation is motivated by their ability to effectively capture temporal features and sequential patterns in sign language videos. Unlike conventional models that primarily focus on recognizing

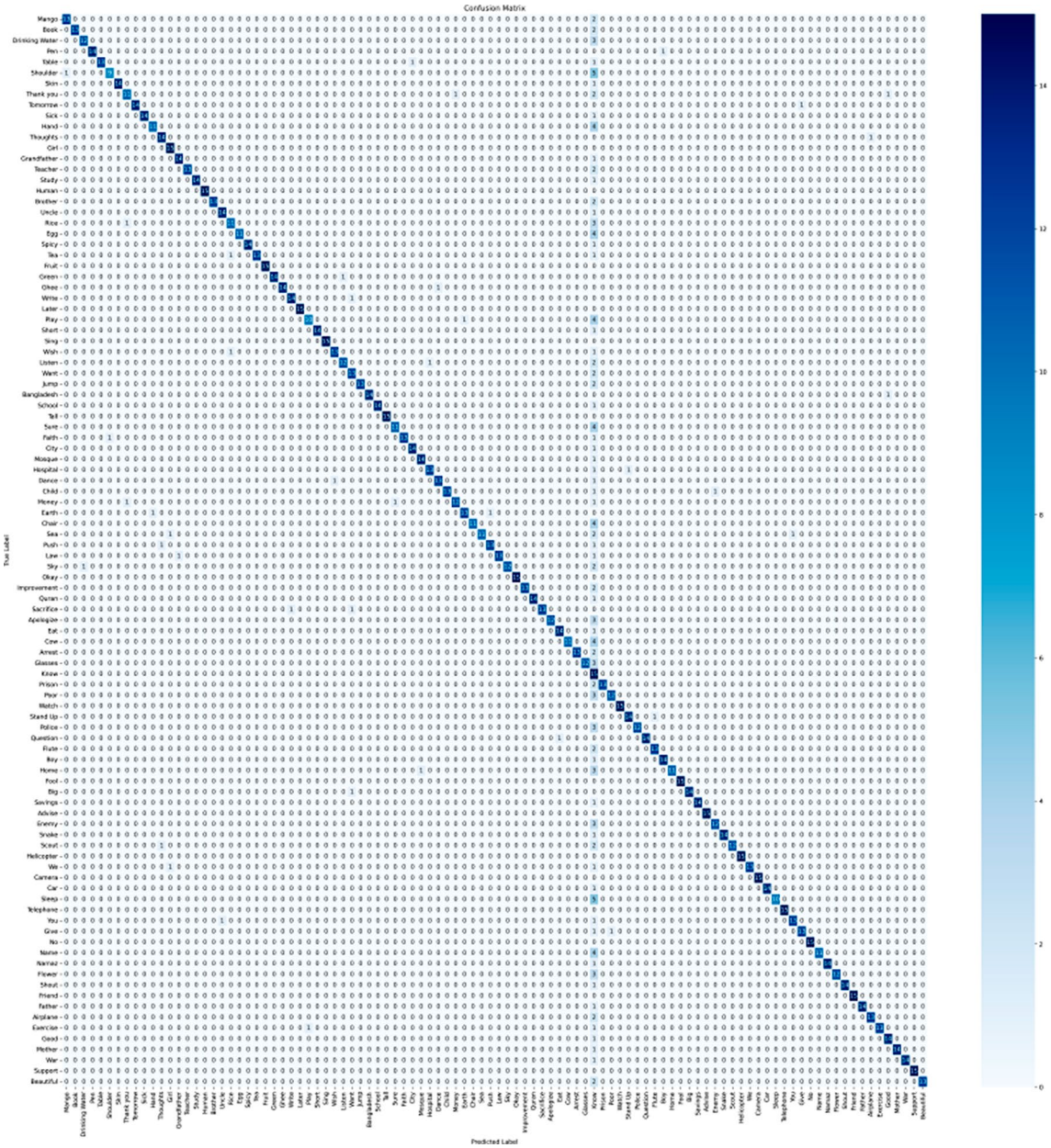

Fig. 8 Confusion matrix of the trained model

individual letters or static gestures, LSTMs are well-suited for modeling dynamic hand movements, transitions, and contextual variations across frames. While alternative deep learning approaches may achieve slightly higher accuracy, they often overlook the sequential nature of sign gestures, treating them as isolated spatial features rather than structured motion sequences. Our proposed model preserves the temporal structure of signs, enabling a more context-aware interpretation of word-level gestures. This approach is particularly valuable in real-world applications such as automated sign language translation, assistive communication systems, and human-computer interaction, where accurately capturing the flow and progression of gestures is critical for generating meaningful translations.

Table 3 Precision, recall, F1 score and support of the 100 sign words

S.No	Class	Precision	Recall	F1-Score	Support
1	Mango	0.929	0.867	0.897	15.0
2	Book	1.000	0.867	0.929	15.0
3	Drinking Water	0.923	0.800	0.857	15.0
4	Pen	1.000	0.933	0.966	15.0
5	Table	1.000	0.867	0.929	15.0
6	Shoulder	0.900	0.600	0.720	15.0
7	Skin	1.000	0.933	0.966	
8	Thank you	0.846	0.733	0.786	15.0
9	Tomorrow	0.000	0.933	0.966	15.0
10	Sick	1.000	0.933	0.966	15.0
11	Hand	0.917	0.733	0.815	15.0
12	Thoughts	0.875	0.933	0.903	15.0
13	Girl	0.882	1.000	0.938	15.0
14	Grandfather	0.933	0.933	0.933	15.0
15	Teacher	1.000	0.867	0.929	15.0
16	Study	1.000	0.933	0.966	15.0
17	Human	1.000	1.000	1.000	15.0
18	Brother	1.000	0.867	0.929	15.0
19	Uncle	0.933	0.933	0.933	15.0
20	Rice	0.846	0.733	0.786	15.0
21	Egg	1.000	0.733	0.846	15.0
22	Spicy	1.000	0.933	0.966	15.0
23	Tea	1.000	0.867	0.929	15.0
24	Fruit	1.000	1.000	1.000	15.0
25	Green	1.000	0.933	0.966	15.0
26	Ghee	1.000	0.933	0.966	15.0
27	Write	0.933	0.933	0.933	15.0
28	Later	1.000	1.000	1.000	15.0
29	Play	0.909	0.667	0.769	15.0
30	Short	1.000	0.933	0.966	15.0
31	Sing	1.000	1.000	1.000	15.0
32	Wish	0.929	0.867	0.897	15.0
33	Listen	0.923	0.800	0.857	15.0
34	Want	0.813	0.867	0.839	15.0
35	Jump	1.000	0.867	0.929	15.0
36	Bangladesh	1.000	0.933	0.966	15.0
37	School	1.000	0.933	0.966	15.0
38	Tall	1.000	1.000	1.000	15.0
39	Sure	0.917	0.733	0.815	15.0
40	Faith	1.000	0.867	0.929	15.0
41	City	0.933	0.933	0.933	15.0
42	Mosque	0.933	0.933	0.933	15.0
43	Hospital	0.929	0.867	0.897	15.0
44	Dance	0.929	0.867	0.897	15.0
45	Child	1.000	0.867	0.929	15.0
46	Money	0.923	0.800	0.857	15.0
47	Earth	0.929	0.867	0.897	15.0
48	Chair	1.000	0.733	0.846	15.0
49	Sea	1.000	0.733	0.846	15.0
50	Push	0.929	0.867	0.897	15.0
51	Law	1.000	0.867	0.929	15.0
52	Sky	1.000	0.800	0.889	15.0
53	Okay	1.000	1.000	1.000	15.0
54	Improvement	1.000	0.867	0.929	15.0
55	Quran	1.000	0.933	0.966	15.0

Table 3 (continued)

S.No	Class	Precision	Recall	F1-Score	Support
56	Sacrifice	1.000	0.867	0.929	15.0
57	Apologize	1.000	0.800	0.889	15.0
58	Eat	0.933	0.933	0.933	15.0
59	Cow	1.000	0.733	0.846	15.0
60	Arrest	1.000	0.867	0.929	15.0
61	Glasses	1.000	0.800	0.889	15.0
62	Know	0.100	1.000	0.182	15.0
63	Prison	1.000	0.867	0.929	15.0
64	Poor	0.923	0.800	0.857	15.0
65	Watch	1.000	1.000	1.000	15.0
66	Stand Up	0.933	0.933	0.933	15.0
67	Police	1.000	0.800	0.889	15.0
68	Question	1.000	0.933	0.966	15.0
69	Flute	0.929	0.867	0.897	15.0
70	Boy	0.933	0.933	0.933	15.0
71	Home	1.000	0.733	0.846	15.0
72	Fool	1.000	1.000	1.000	15.0
73	Big	1.000	0.933	0.966	15.0
74	Savings	1.000	0.933	0.966	15.0
75	Advise	1.000	1.000	1.000	15.0
76	Enemy	0.923	0.800	0.857	15.0
77	Snake	1.000	0.933	0.966	15.0
78	Scout	1.000	0.800	0.889	15.0
79	Helicopter	1.000	1.000	1.000	15.0
80	We	1.000	0.867	0.929	15.0
81	Camera	1.000	1.000	1.000	15.0
82	Car	1.000	0.933	0.966	15.0
83	Sleep	1.000	0.667	0.800	15.0
84	Telephone	1.000	1.000	1.000	15.0
85	You	0.929	0.867	0.897	15.0
86	Give	0.929	0.867	0.897	15.0
87	No	1.000	1.000	1.000	15.0
88	Name	1.000	0.733	0.846	15.0
89	Namaz	1.000	0.933	0.966	15.0
90	Flower	1.000	0.800	0.889	15.0
91	Shout	1.000	0.933	0.966	15.0
92	Friend	1.000	1.000	1.000	15.0
93	Father	1.000	0.933	0.966	15.0
94	Airplane	0.929	0.867	0.897	15.0
95	Exercise	1.000	0.867	0.929	15.0
96	Good	0.875	0.933	0.903	15.0
97	Mother	1.000	0.933	0.966	15.0
98	War	1.000	0.933	0.966	15.0
99	Support	1.000	1.000	1.000	15.0
100	Beautiful	1.000	0.867	0.929	15.0
Accuracy		0.883			
Macro Avg		0.962	0.883	0.915	1500.0
Weighted Avg		0.962	0.883	0.915	1500.0

Table 4 Comparative study

Researches	Method	Detected Classes	Input Type	Results
Rahaman et al. (Rahaman et al., 2020)	Window-grid vector (WGV)	51 Bangla written characters	Static and Dynamic	Accuracy 95.80%
Shafiqul Islalm et al. (Islalm et al., 2019)	CNN	35 Bangla alphabets and 10 Bangla digits	Static	Accuracy 99.8%
Paromita Urmee et al. (Urmee et al., 2019)	Xception Augmentation	37 Bangla alphabets	Static	Accuracy 98.93%
Sadik et al. (Sadik et al., 2019)	Binary Masking; SVM	10 Bangla alphabets	Static	Accuracy 99.8%
M. M. Islam et al. (Islam et al., 2022)	VGG16	11 BdSL sign words	Static	Accuracy 99.92%
Dipon Talukder et al. (Talukder & Jahara, 2020)	YOLOv4	39 Bangla alphabets and 10 Bangla digits	Static	Accuracy 97.95%
Kanchon et al. (Podder et al., 2020)	Color-coded fingertip; ResNet18	37 Bangla alphabets	Static	Accuracy 99.97%
Kanchon et al. (Podder et al., 2022)	ResNet18 with background; DenseNet201 FPN - MobileNet_V2 without background	38 one-handed and 36 two-handed gestures of Bangla alphabets and 10 Bangla digits	Static	Accuracy 99.99%; Accuracy 99.91%
Proposed method	LSTM	100 BdSL sign words	Dynamic	Accuracy 88.33%; AUC 98.56%

5 Conclusion

Effective communication between speech impaired and non speech impaired people is the main concern of this research. In particular, in Bangladesh, the situation is bad and the communication gap is huge for deaf and mute people. Translating both verbal and sign language at the user end is the only key to the solution. Deep learning can be a promising solution in this regard. Dynamic gesture recognition is very important for this translation system to be feasible. The dataset we created has 100 sign gestures, which may not be enough to continue a conversation. So, we will try to enrich our dataset with all the sign words that the Bangla sign language dictionary has in our future works. Currently, the system is able to show one word on the screen at a time. To make a full sentence from the signs that a user waves can also be promising research in the future. Furthermore, converting the detected text words into speech can be more helpful to understand the sign language and subject to future works. This will help the deaf and mute to communicate with blind people. We think that this work can help to explore the new possibilities of human-computer interaction.

Acknowledgements This work is supported by the project offered by Research \& Publication Cell, University of Chittagong, Bangladesh (No. 03/2023-24/42/2024).

Author contributions Aonmoy Das: Conceptualization, Methodology, Investigation, Writing - Review & Editing; Ananna Dev Aishi: Methodology, Investigation, Writing - Review & Editing; Masbah Uddin Toha: Conceptualization, Methodology, Investigation, Original draft preparation; Md. Fazlul Kader: Investigation, Original draft preparation, Writing - Review & Editing, Supervision.

Funding This study did not receive any funding.

Data availability The dataset (BdSL-LSTM dataset containing 100 Bangla Sign Words, 2025) was generated during the current study.

Declarations

Competing interests The authors declare no competing interests.

References

Aly, S., & Aly, W. (2020). DeepArSLR: A novel signer-independent deep learning framework for isolated arabic sign language gestures recognition. *IEEE Access*, *8*, 83199–83212.

Aly, W., Aly, S., & Almotairi, S. (2019). User-independent american sign language alphabet recognition based on depth image and PCANet features. *IEEE Access*, *7*, 123138–123150.

American-sign-language. n.d., https://www.nidcd.nih.gov/health/american-sign-language. Accessed: 2022-03-12

Bantupalli, K., & Xie, Y. (2018) American sign language recognition using deep learning and computer vision. In *2018 IEEE international conference on big data* (Big Data) (pp. 4896–4899).

Barche, P., Gurugubelli, K., & Vuppala, A. K. (2024). Stockwell-transform based feature representation for detection and assessment of voice disorders. *International Journal of Speech Technology*, *27*(1), 101–119.

BdSL-LSTM dataset containing 100 Bangla Sign Words. *Zenodo* (2025). https://doi.org/10.5281/zenodo.14955703

Cao, Z., Simon, T., Wei, S.-E., & Sheikh, Y. (2017) Realtime multi-person 2D pose estimation using part affinity fields. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 7291–7299).

Cheok, M. J., Omar, Z., & Jaward, M. H. (2019). A review of hand gesture and sign language recognition techniques. *International Journal of Machine Learning and Cybernetics*, *10*(1), 131–153.

Committee, B. S. L. (1994). *Bangla Sign Language Dictionary* (1st edition edn ed.). National Center for Special Education.

Cravotta, A., Prieto, P., & Busà, M. G. (2021). Exploring the effects of restraining the use of gestures on narrative speech. *Speech Communication*, *135*, 25–36.

Greff, K., Srivastava, R. K., Koutnk, J., Steunebrink, B. R., & Schmidhuber, J. (2016). LSTM: A search space odyssey. *IEEE Transactions on Neural Networks and Learning Networks*, *28*(10), 2222–2232.

Haque, P., Das, B., & Kaspy, N. N. (2019) Two-handed bangla sign language recognition using principal component analysis (PCA) and KNN algorithm. In *2019 International conference on electrical, computer and communication engineering (ECCE)* (pp. 1–4).

Hashan, A. M., Dmitrievich, C. R., Valerievich, M. A., Vasilyevich, D. D., Alexandrovich, K. N., & Bredikhin, B. A. (2024). Hyperkinetic dysarthria voice abnormalities: A neural network solution for text translation. *International Journal of Speech Technology*, *27*(1), 255–265.

Hoque, O. B., Jubair, M. I., Islam, M. S., Akash, A.-F., & Paulson, A. S. (2018) Real time bangladeshi sign language detection using faster R-CNN. In *2018 International Conference on Innovation in Engineering and Technology (ICIET)* (pp. 1–6).

Islalm, M. S., Rahman, M. M., Rahman, M. H., Arifuzzaman, M., Sassi, R., & Aktaruzzaman, M. (2019) Recognition Bangla sign language using convolutional neural network. In *2019 International conference on innovation and intelligence for informatics, computing, and technologies (3ICT)* (pp. 1–6).

Islam, M. M., Uddin, M. R., AKhtar, M. N., & Alam, K. R. (2022). Recognizing multiclass static sign language words for deaf and dumb people of bangladesh based on transfer learning techniques. *Information in Medicine Unlocked*, *33*, 101077.

Kang, I., Goy, A., & Barbastathis, G. (2021). Dynamical machine learning volumetric reconstruction of objects' interiors from limited angular views. *Light: Science & Applications*, *10*(1), 1–21.

Khan, P., Kader, M. F., Islam, S. R., Rahman, A. B., Kamal, M. S., Toha, M. U., & Kwak, K.-S. (2021). Machine learning and deep learning approaches for brain disease diagnosis: Principles and recent advances. *IEEE Access*, *9*, 37622–37655.

Kudrinko, K., Flavin, E., Zhu, X., & Li, Q. (2021). Wearable sensor-based sign language recognition: A comprehensive review. *IEEE Reviews in Biomedical Engineering*, *14*, 82–97.

Lyu, Y., & Huang, X. (2018). Road segmentation using CNN with GRU. arXiv preprint arXiv:1804.05164.

MediaPipe Holistic. n.d. https://google.github.io/media-pipe/solutions/holistic. Accessed 14 March 2022.

Pires, I. M., Hussain, F., Marques, G., & Garcia, N. M. (2021). Comparison of machine learning techniques for the identification of human activities from inertial sensors available in a mobile device after the application of data imputation techniques. *Compututers in Biology and Medicine*, *135*, 104638.

Podder, K. K., Chowdhury, M., Mahbub, Z. B., & Kadir, M. (2020). Bangla sign language alphabet recognition using transfer learning based convolutional neural network. *Bangladesh Journal of Scientific Research*, 31–33.

Podder, K. K., Chowdhury, M. E., Tahir, A. M., Mahbub, Z. B., Khandakar, A., Hossain, M. S., & Kadir, M. A. (2022). Bangla sign language (BdSL) alphabets and numerals classification using a deep learning model. *Sensors*, *22*(2), 574.

Rahaman, M. A., Jasim, M., Ali, M., & Hasanuzzaman, M. (2020). Bangla language modeling algorithm for automatic recognition of hand-sign-spelled Bangla sign language. *Frontiers in Compututer Science*, *14*(3), 1–20.

Raj, R. D., & Jasuja, A. (2018) British sign language recognition using HOG. In *2018 IEEE international students' conference on electrical, electronics and computer science (SCEECS)* (pp. 1–4).

Rastgoo, R., Kiani, K., & Escalera, S. (2021). Sign language recognition: A deep survey. *Expert Systems with Applications*, *164*, 113794.

Rosero-Montalvo, P. D., Godoy-Trujillo, P., Flores-Bosmediano, E., Carrascal-García, J., Otero-Potosi, S., Benitez-Pereira, H., & Peluffo-Ordóñez, D. H. (2018) Sign language recognition based on intelligent glove using machine learning techniques. In *2018 IEEE third Ecuador technical chapters meeting (ETCM)* (pp. 1–5).

Sadik, F., Subah, M. R., Dastider, A. G., Moon, S. A., Ahbab, S. S., & Fattah, S. A. (2019) Bangla sign language recognition with skin segmentation and binary masking. In *2019 IEEE international WIE conference on electrical and computer engineering (WIECON-ECE)* (pp. 1–5).

Sanzidul Islam, M., Sultana Sharmin Mousumi, S., Jessan, N. A., Shahariar Azad Rabby, A., & Akhter Hossain, S. (2018) Ishara-Lipi: The first complete multipurposeopen access dataset of isolated characters for bangla sign language. In *2018 international conference on Bangla speech and language processing (ICBSLP)* (pp. 1–4).

Shanta, S. S., Anwar, S. T., & Kabir, M. R. (2018) Bangla sign language detection using SIFT and CNN. In *2018 9th international conference on computing, communication and networking technologies (ICCCNT)* (pp. 1–6).

Supančič, J. S., Rogez, G., Yang, Y., Shotton, J., & Ramanan, D. (2018). Depth-based hand pose estimation: Methods, data, and challenges. *International Journal of ComputerVision*, *126*(11), 1180–1198.

Talukder, D., & Jahara, F. (2020) Real-time bangla sign language detection with sentence and speech generation. In *2020 23rd international conference on computer and information technology (ICCIT)* (pp. 1–6).

Urmee, P. P., Mashud, M. A. A., Akter, J., Jameel, A. S. M. M., & Islam, S. (2019) Real-time bangla sign language detection using Xception model with augmented dataset. In *2019 IEEE international WIE conference on electrical and computer engineering (WIECON-ECE)* (pp. 1–5).

Van Houdt, G., Mosquera, C., & Nápoles, G. (2020). A review on the long short-term memory model. *Artificial Intelligence Review*, *53*(8), 5929–5955.

What is British sign language. n.d.. https://www.british-sign.co.uk/what-is-british-sign-language. Accessed 12 March 2022.

Yu, Y., Si, X., Hu, C., & Zhang, J. (2019). A review of recurrent neural networks: LSTM cells and network architectures. *Neural Computation*, *31*(7), 1235–1270.

Zhu, Y., Lu, W., Gan, W., & Hou, W. (2021). A contactless method to measure real-time finger motion using depth-based pose estimation. *Computers in Biology and Medicine*, *131*, 104282.