Pattern Analysis and Applications (2024) 27:37
https://doi.org/10.1007/s10044-024-01229-4

INDUSTRIAL AND COMMERCIAL APPLICATION

Spatial–temporal attention with graph and general neural network-based sign language recognition

Abu Saleh Musa Miah[1] · Md. Al Mehedi Hasan[2] · Yuichi Okuyama[1] · Yoichi Tomioka[1] · Jungpil Shin[1]

Received: 28 April 2023 / Accepted: 31 December 2023 / Published online: 4 April 2024

Abstract

Automatic sign language recognition (SLR) stands as a vital aspect within the realms of human–computer interaction and computer vision, facilitating the conversion of hand signs utilized by individuals with significant hearing and speech impairments into equivalent text or voice. Researchers have recently used hand skeleton joint information instead of the image pixel due to light illumination and complex background-bound problems. However, besides the hand information, body motion and facial gestures play an essential role in expressing sign language emotion. Also, a few researchers have been working to develop an SLR system by taking a multi-gesture dataset, but their performance accuracy and time complexity are not sufficient. In light of these limitations, we introduce a spatial and temporal attention model amalgamated with a general neural network designed for the SLR system. The main idea of our architecture is first to construct a fully connected graph to project the skeleton information. We employ self-attention mechanisms to extract insights from node and edge features across spatial and temporal domains. Our architecture bifurcates into three branches: a graph-based spatial branch, a graph-based temporal branch, and a general neural network branch, which collectively synergize to contribute to the final feature integration. Specifically, the spatial branch discerns spatial dependencies, while the temporal branch amplifies temporal dependencies embedded within the sequential hand skeleton data. Further, the general neural network branch enhances the architecture's generalization capabilities, bolstering its robustness. In our evaluation, utilizing the Mexican Sign Language (MSL), Pakistani Sign Language (PSL) datasets, and American Sign Language Large Video dataset which comprises 3D joint coordinates for face, body, and hands that conducted experiments on individual gestures and their combinations. Impressively, our model demonstrated notable efficacy, achieving an accuracy rate of 99.96% for the MSL dataset, 92.00% for PSL, and 26.00% for the ASLLVD dataset, which includes more than 2700 classes. These exemplary performance metrics, coupled with the model's computationally efficient profile, underscore its preeminence compared to contemporaneous methodologies in the field.

Keywords Human–computer interaction (HCI) · Sign language recognition (SLR) · Spatial–temporal attention (STA) · Temporal–spatial attention(TSA) · Skeleton point · Mexican Sign Language(MSL) · ASLLVD · PSL · Pose estimation

1 Introduction

Sign language serves as a means of communication for the hard of hearing and deaf (HHD) community, enabling them to express their ideas and needs. It encompasses various

Abu Saleh Musa Miah, Md. Al Mehedi Hasan and Jungpil Shin have contributed equally to this work.

✉ Jungpil Shin
jpshin@u-aizu.ac.jp

Abu Saleh Musa Miah
d8231105@u-aizu.ac.jp

Md. Al Mehedi Hasan
mehedi_ru@yahoo.com

Yuichi Okuyama
okuyama@u-aizu.ac.jp

Yoichi Tomioka
ytomioka@u-aizu.ac.jp

[1] School of Computer Science and Engineering, The University of Aizu, Aizuwakamatsu, Fukushima 965-8580, Japan

[2] Department of Computer Science and Engineering, Rajshahi University of Engineering & Technology, Rajshahi 6204, Bangladesh

gestures involving different body parts, including hands, fingers, head, body, and facial expressions, facilitating daily interactions within the HHD community [1]. Globally, approximately 70 million individuals are part of the HHD community [2], with around 4.2 million residing in Mexico [3] and approximately 3 million in Bangladesh [4]. Unfortunately, many members of the HHD community lack proficiency in sign language, leading to communication difficulties and challenges in accessing essential services such as healthcare, socialization, education, and employment [5]. While human sign language interpreters could potentially alleviate these challenges, finding qualified interpreters is daunting and often accompanied by significant costs.

An effective solution to bridge the gap between the HHD community and the general population is the application of computer vision, particularly by developing an automatic sign language recognition system. Currently, researchers explore two main domains for sign language recognition [6]: the sensor-based approach [7] and the vision-based approach [8]. In the vision-based approach, researchers typically focus on two modalities: image-based information [9, 10] and skeleton-based information [3, 11]. The image-based approach involves using RGB or RGB-D format input images to extract image-based features for classification. However, this method encounters challenges related to varying lighting conditions and occlusions. In contrast, the skeleton-based modality relies on sequences of 3D or 2D coordinates of hand skeletons to predict hand gestures. This approach is more robust compared to the image-based approach. The 3D hand joint data are typically obtained using affordable depth cameras like Microsoft Kinect, Intel RealSense, or Microsoft Oak-D, which enhance hand pose estimation accuracy [3, 10]. Some researchers have employed traditional methods to extract powerful features from hand joints [12–14], achieving high performance in some instances but still facing limitations in generalization capabilities. Recently, researchers have turned to deep learning algorithms to improve performance accuracy [15–17]. They concatenate joint coordinates into tensors and train neural networks to learn hand features from the data. However, they often overlook the temporal and spatial aspects of the skeleton dataset.

As a result, deep learning-based CNN algorithms tend to capture holistic scenes, which may not be ideal for sign language recognition. While this holistic approach works well for general hand gesture and activity recognition tasks due to scene patterns, it may not be relevant for sign language recognition (SLR). Yan et al. focused their efforts on extracting hand skeleton joint information using the OpenPose framework to facilitate sign language recognition (SLR) [18]. Subsequently, they applied a graph-based spatial and temporal feature method, ST-GCN, to recognize sign language gestures. While their methodology exhibited promising results in sign language recognition, it was limited to utilizing only hand skeletons. While beneficial in addressing challenges related to varying lighting conditions and complex backgrounds, this approach falls short of fully conveying the meaning of signs. The prevailing consensus emphasizes that comprehending sign language accurately requires considering the entire body, encompassing body and facial pose information in conjunction with hand gestures. The position of the hands in relation to the entire body is crucial for interpreting sign language effectively. Furthermore, including whole-body information can offer valuable insights into the signer's communication and convey emotions and expressions among individuals with profound hearing and speech impairments. Many researchers have recognized the importance of incorporating body and facial expression information alongside hand information [3, 19–21].

Yan et al. introduce a custom graphical layout that recognizes the 20 ASL dataset classes [19]. They mainly enhanced the who especially extended the method. This enhanced approach incorporated body and hand landmarks, resulting in commendable performance accuracy. Solis et al. also delved into the realm of sign language recognition, employing RNN and LSTM algorithms to recognize 30 MSL words [3]. Their approach included facial, body, and hand-joint information. However, while their efforts showed promise, they faced challenges in terms of computational complexity and performance accuracy. Recently, many researchers applied the self-attention mechanism mainly employed in the NLP field [20–24, 24–27] to classify hand skeleton information. Moreover, the attention-based neural network is known for accurately detecting hand gestures compared to CNN and LSTM, where the self-attention model usually reduces the longer-term dependencies by running the attention mechanism in parallel. One of the most crucial challenges associated with static graph-based neural network systems lies in the extensive message passing required in each operation, resulting in a substantial number of feature aggregation and updating operations. This overreliance on node feature aggregation can lead to feature over-smoothing, as noted in previous work [28]. Over time, the nodes in the graph can produce redundant or similar information, diminishing the contribution of graph construction-based classification compared to conventional CNNs.

To address these issues, Hammadi et al. introduced a graph-based 3DCNN algorithm designed to represent skeleton information [29–31]. While their model effectively captured long-range dependencies using spatial attention, they did not explore temporal attention or address the temporal context suppression of gestures. Moreover, their performance accuracy and computational complexity fell short of expectations. To rectify the limitations related to performance and computational complexity in whole-body-based sign language recognition (SLR), our study presents a novel approach: the spatial and temporal attention model integrated with a general neural network-based SLR system. Notably, static graph construction methods can encounter challenges in extracting effective features in both spatial and temporal domains.

Our key contributions are outlined as follows:

- We proposed a spatial and temporal attention model with a general neural network-based SLR system, including face, body, and hands, instead of relying solely on hand information. Our main idea is to implement a fully connected graph derived from the entire body skeleton to dynamically learn node and edge features.
- We employed several principles of the multi-attention technique in designing spatial attention, temporal attention, and a general deep neural network model. The first branch generated spatial features using multi-head attention, while the second branch generated temporal features using the same approach. Both the first and second branches incorporated domain-specific mask operations, enhancing the spatial and temporal context representation of gestures. The third branch of the architecture conveys deep learning-based features. Finally, we integrated all these features to create a final feature vector.
- Lastly, we have comprehensively validated our system using three dynamic skeleton-based sign language datasets that include hand, body, and face information. We achieved superior performance compared to state-of-the-art methods while maintaining minimal computational time. One of the primary reasons for this efficiency is that we randomly selected eight frames from the video sequences. We have uploaded the data and code to the following GitHub link: https://github.com/musaru/*Spatial_Temporal_SLR*.

The study we presented here according to the following sequences: literature review based on a recent scientific article we presented in Sect. 2, three datasets of the SLR used to evaluate the system are reported in Sect. 3.1, methodology and materials are described in Sect. 4 performance result of the proposed model with various dataset with state-of-the art comparison are described in 5 . The conclusion and future work are reported in Sect. 6.

2 Related work

There are many methodologies have been developed to analyze the sign language recognizer over the decades based on computer vision [4, 9–11, 32]. In the literature, many researchers developed sensor-based gesture recognition system such as EEG [33–35], hand gloves-based translating approach [36–39], to reduce the computational cost of the system for real-time implementation of the gesture recognition system. Besides this, many researchers proposed machine learning-based methods to accurately detect sign language using specialized sensors such as leap motion and kinetic devices by acquiring 3D motion gesture data [40–42]. Raghuveera et al. proposed local binary patterns and HOG-based feature extraction techniques to extract features from the kinetic dataset for the Indian Sign Language and achieved 78.85% accuracy with SVM approaches [43]. In the same way, Khan et al. proposed a sign language translator for Pakistani and Indian Sign Language-based kinetic device datasets, including color and depth information and achieved 77.00% accuracy [44].

Miah et al. proposed various transformer-based sign language recognition with the kinetic RGB Dataset and achieved a good performance accuracy for the American Sign Language (ASL) dataset [10]. Xiao et al. proposed an LSTM model to translate the kinetic China Sign Language image into the equivalence audio and vice version, and they achieved 79.12% accuracy [45]. Jiang et al. applied a 3D CNN approach to recognize kinetic-based ASL, and they obtained 92.88% accuracy [46]. However, this sensor-based work has been limited to hand processing. Last decades, researchers were again focused on RGB cameras to collect the hand skeleton information of the hand gesture and hand signs [4, 32, 47–51]. Many researchers have developed camera data-based hand gesture recognition using conventional machine learning algorithms. For example, Solis et al. applied Jacobi–Fourier Moments (JFMs) and artificial neural networks (ANN) to classify Mexican Sign Language (MSL) and achieved 95% accuracy [52]. Cervantes et al. extracted 743 features, then applied them to select the potential features and achieved 97% accuracy using SVM algorithms [53].

On the other hand, Adhikary et al. collected hand skeleton information using a Mediapipe through the RGB camera [54]. They employed a random forest algorithm and achieved 97.4% accuracy for the Indian Sign Language (ISL). Recently, SLR researchers have focused on deep learning-based architecture because of the numerous achievements in different fields. Okhan et al. employed inception transformer learning for multimodal SLR, including RGB and optical flow [55]. In the same way, different transfer learning approaches applied by many researchers such as ResNet [56], LSTM, and 3DNCN [57, 58], video transformer with bidirectional LSTM [59]. Also, image-based sign language recognition has many merits, like low cost, easy acquisition, and real-time implementation. However, many limitations exist, such as background complexity, light illumination, orientation, and partial occlusion. In addition, the major drawback of the image-based SLR using the CNN algorithm is the holistic involvement of the scene. This holistic scene may not be a problem for the general hand gesture and activity recognition task because of the pattern of the scene, but that is non-relevant for the SLR. To solve this issue, many researchers used different RGB cameras with spatial types of software, such as OpenPose, Mediapipe, etc., to extract hand joint skeleton information to collect the exact pattern of the hand sign [60, 61].

Shin et al. extracted skeleton information for the ASL using Mediapipe and then applied various approaches for recognizing this and achieved good performance [61]. Xie et al. collected an American Sign Language dataset using an RGB-D camera and achieved 91.35% accuracy after applying the CNN algorithm [62]. Yan et al. extracted only hand skeleton joint information using the OpenPose framework for the SLR [18]. Then, they applied a Graph-based construction-based spatial and temporal feature method, namely ST-GCN, to recognize them. Although the existing methodology produced a good performance in recognizing sign language, they used only hand skeletons; only hand skeleton information can not contain the complete meaning of the sign language. Besides the hand skeleton, the body and facial information of deaf people can play an essential role in collecting extracted expressions and emotions of the hearing and speech impaired. Amorim et al. enhanced the previous network [18], including a custom graphic layout for recognizing the 20 classes of the ASLLVD dataset and achieved good performance accuracy with it [19].

Solis et al. employed RNN and LSTM algorithms to recognize 30 MSL words and achieved 97.00 accuracy [3]. They also included facial and body information and hand joint information, but their computational complexity and performance accuracy were not very good. Recently, many researchers applied the self-attention mechanism mainly employed in the NLP field [20–24, 24–27] to classify hand skeleton information. Moreover, the attention-based neural network is known for accurately detecting hand gestures compared to CNN and LSTM, where the self-attention model usually reduces the longer-term dependencies by running the attention mechanism in parallel.

The mentioned graph-based neural network system is mainly executed by passing the message in each operation, which needs a huge number of feature aggregation and updating operations. The node feature aggregation can lead to the over-smoothing of the feature [28]. In the sequence, graph nodes can produce similar information and graph construction-based classification contribution becomes less than the normal CNN. To overcome the problem, Hammadi et al. proposed a graph-based 3DCNN algorithm to represent the node of the skeleton information [29–31]. The main drawback of this work is that they applied only spatial attention but did not include any experiment or explanation about the temporal feature. In addition, static graph construction may face problems extracting effective features in both spatial and temporal domains.

To overcome the challenges, the study used a spatial and temporal attention model with general neural network-based SLR. Our main idea is to use a fully connected graph to overcome static graph problems in implementing the system; we included attention and neural network-based three branches: spatial attention, temporal attention, and a general neural network branch. In addition, we applied spatial and temporal masking operations in each branch to reduce the computational complexity of the self-attention block. Our system is more efficient than the existing one because we directly model the dependencies of joints with pure attention blocks instead of formulating the skeletal data into images or graphs. In addition, the proposed system is more concise and general because there is no need to design handcrafted transformation rules. Also, it outperforms the previous state-of-the-art methods by a significant margin. Based on our best knowledge, we first use attention to apply the pure attention networks for sign language and propose several improvements to meet the specific requirements.

3 Dataset description

In the study, we are focusing on the whole body information for sign language recognition. Most sign language researchers used only hand information for this, and very few researchers have worked including body and face with hand information. Consequently, we got a few sign language datasets with whole-body information, including body, face, and hand skeleton, from the open source. We described three sign language datasets, including body, face, and hand skeleton. The names of the datasets are MSL [3], ASLLVD [63], and PSL dataset [33]. We selected this dataset in the study because these three skeleton-based sing language datasets contained similar body, face, and hand skeleton characteristics. There are 67 key points in the MSL and PSL dataset, including 21 for one hand, 42 for two hands, and 25 for body and face. The details of the whole body skeleton are written here [64]. We can define the 3D skeleton information as a vector from the video or a frame sequence using the below formula Eq. (1).

$$S = (Pm_1, Pm_2, Pm_3, ..., Pm_n)^T \tag{1}$$

where Pm_i denotes the multivariate time step sequence, S denotes the skeleton data sequences, and the transpose of the matrix is denoted by T. Based on the Eq. (1) we can define the sequence component $P_j = \left(P_j(t)\right)_{t \in N}$, to containing the three coordinates of the skeleton can be explained using the below formula Eq. (2).

$$P_j = \left(X^{(i)}, Y^{(i)}, Z^{(i)}\right) \tag{2}$$

where X, Y, and Z, contained the coordinates value for a specific joint *ith* joint, respectively. Moreover, the position of the *ith* joint can be expressed with the $P_j(t)$. Every frame joint consists of a precise articulation of the hand gesture of the real world. These 67 joints are collected for the t time frame of the MSL and PSL, which are recorded in a 3D space using a WAK-D camera. The position of each skeleton point can be written as $Pm_j = \left(X_j, Y_j, Z_j\right) \in \mathbb{R}^3, \forall j \in [1;N]$, where $N = 67$, for MSL and PSL and 27 for the ASLLVD datasets.

3.1 Mexican Sign Language (MSL)

This dataset was collected using an OAK-D camera with integrating MediaPipe libraries [3]. The dataset is collected from four people with 25 repetitions for 30 different signs in Mexican Sign Language. The details for the dataset are shown in Table 1, and a sample of the MSL dataset is demonstrated in Fig. 1. They collected 3000 samples for the dataset, where 20 videos were for each sign and 20 sequence frames for each video. After that, they extracted 3D skeleton key points for the whole body, including the body, face, and two hand coordinates. Although they collected 468 landmarks for the face, which have a strong correlation, they selected 20 landmarks among those using some feature selection. They selected the most effective facial landmarks around the five places, including two eyes, a mouth, and two eyebrows. For each area, they selected 4 points and, in total, 20 points. They collected 4 points from the upper body, including shoulders, chest, and elbows. The Mediapipe library collects the skeleton key points.

There is no option to extract the chest point, but they calculated it from the average of two shoulders. After that, for the hand, they use all landmarks, 21 points on each hand. Figure 2 demonstrates the 67 key points in the whole body, where 20 were from the face, five from the body, and 42 key points from 2 hands.

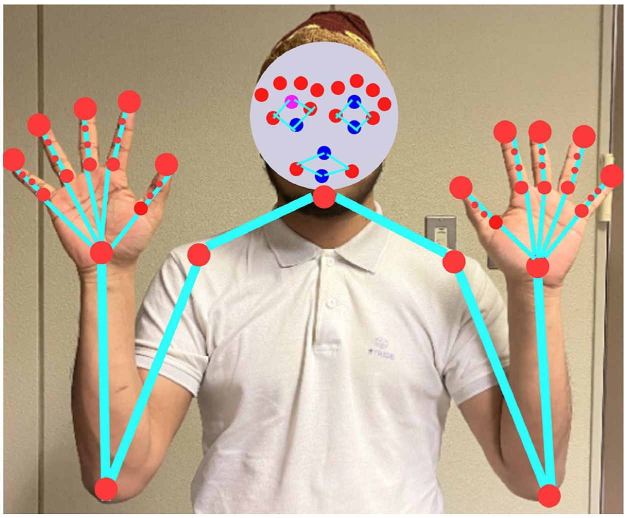

Fig. 2 Illustration showcasing the 67 key points mapped onto the human body for sign language recognition (SLR) and figure collected from [33]. Each key point represents a specific anatomical landmark, distributed across the face, hands, and body, employed to capture the comprehensive gestural language and used as input features for the recognition model

Firstly, they detected each key point in the RGB image and represented this by $P[x, y]$, coordinates. The obtained depth value Z and compute the 3D space coordinates for X, Y, and Z, using the following (3), and (4):

$$X = \left(\frac{XZ}{f}\right) \tag{3}$$

$$X = \left(\frac{YZ}{f}\right) \tag{4}$$

Here, the pixel coordinates of the image are denoted by X and Y, and the focal length and depth are denoted by f and Z, respectively.

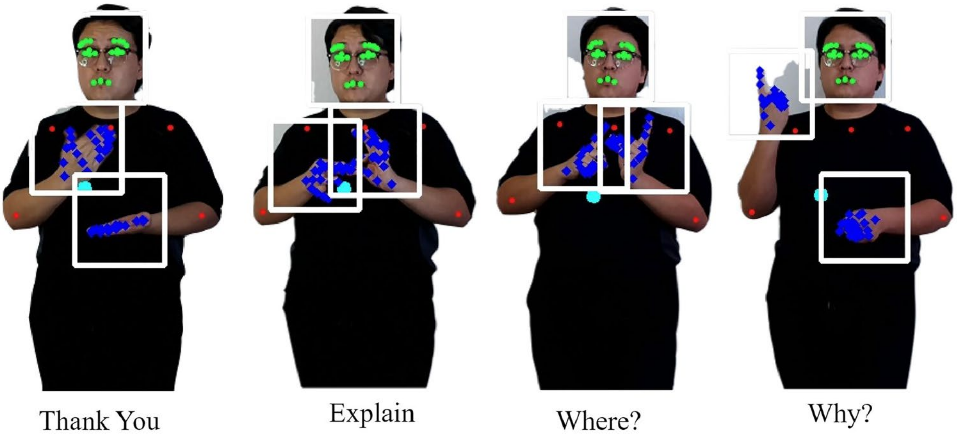

Fig. 1 Sample images extracted from the Mexican Sign Language (MSL) dataset. Each image portrays a distinct hand gesture representative of specific signs within the language. The variations in hand positions, orientations, and shapes are crucial in conveying different meanings [3]

Table 1 Exploring the MSL dataset: featuring 30 gestures across varied sign categories

Type of sign	Sign	No. of hand	Sign type	Sign	No. of hand
Alphabet	A	One		Who?	Two
	B	One	Days of week	Monday	One
	C	One		Tuesday	One
	D	One		Wednesday	One
	J	One		Thursday	One
	K	One		Friday	One
	Q	One		Saturday	One
	X	One		Sunday	One
Questions	What?	Two	Frequent words	Spell	One
	When?	One		Explain	Two
	How much?	Two		Thank you	Two
	Where?	Two		Name	One
	For what?	One		Please	Two
	Why?	One		Yes	One
	What is that?	One		No	One

The labels are subdivided into categories such as alphabets, questions, weekdays, and common words. The gestures utilize either single or bilateral hand information

The skeleton key points they stored in comma-separated values (CSV) were structured with 201 columns and 20 rows. Each row of the file represents a repetition of the individual frames. The coordinate in an order such as the first is five body key points, then 20 facial key points, left hand and right hand.

3.2 Pakistan Sign Language (PSL) OpenPose dataset

PSL comes from Pakistan. Sign language was recorded by the deaf people in Pakistan to make a system for establishing strong communication between the normal and deaf communities. This is the first sign language dataset for Pakistan, which is recorded from the webcam and then extracted whole body key points using the OpenPose system and stored as a JSON file. After that, they annotated the estimated OpenPose and then annotated it with Urdu alphabet labels. Using the open pose, the extracted X, Y, and Z coordinates for 21 left-hand keys, 21 right-hand keys, and 25 from the body and face are based on the Fig. 2. In both cases, this dataset included the alphabet and words, where they considered 12-word signs for the PSL word dataset and 37 characters for the alphabet datasets. Nine people participated in the alphabet and nine in the word dataset. They normalized the dataset by scaling hand and body positions. Figure 2 demonstrated the 67 key points in the whole body, where 20 were from the face, 5 from the body, and 42 key points from 2 hands. The PSL dataset obtained from Kaggle, which is licensed under the GNU General Public License, version 2 (GPL-2) [33]. The dataset has been evaluated for usability with a score of 8.75. The dataset is available at the following link: https://www.kaggle.com/datasets/saadbutt321/pakistan-sign-language-dataset.

3.3 ASLLVD

This is one of the most famous large-scale American Sign Language datasets [29, 63, 65]. The dataset is named ASLLVD, which stands for the American Sign Language Lexicon Video Dataset(ASLLVD), and it comprises 2745 ASL signs recorded from multiple synchronized videos, capturing various angles. The dataset is divided into 7798 training videos and 1950 testing videos [29]. Primarily, this dataset consists of video recordings accompanied by gloss labels, providing start and end times for each sign. Additionally, it includes multiple synchronized videos capturing sign language from various angles, allowing for the observation of hand-shape labels and the morphological and articulatory classification of signs for both hands. In the case of compound signs, the dataset provides annotations for each morpheme. Furthermore, the dataset includes numeric ID labels for sign variants, video sequences in uncompressed-raw format, and camera calibration sequences, all of which facilitate computer vision-based sign language recognition. Figure 3 demonstrates the sample example of the ASLLVD dataset [33].

4 Proposed methodology

The workflow architecture of the proposed model is illustrated in Fig. 4a. It initiates by transforming the characteristics of the skeleton nodes into a new space by applying a

Fig. 3 Sample images from American Sign Language Lexicon Video Dataset (ASLLVD): This dataset encompasses diverse hand gestures, facial expressions, and body postures encoded in a 3D joint skeleton [65]

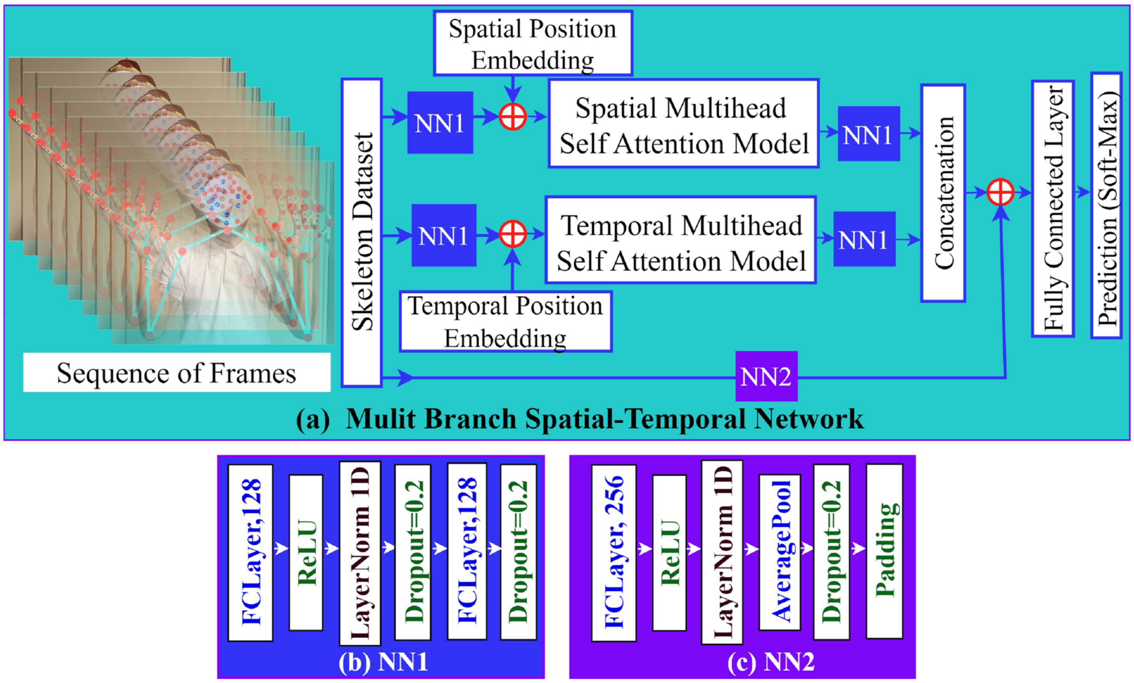

Fig. 4 Schematic representation of the proposed spatial and temporal attention model integrated with a general neural network for SLR. The illustrated workflow encompasses the innovative utilization of a fully connected graph to project skeleton information, strategically bifurcating into three pivotal branches: a graph-based spatial branch, a graph-based temporal branch, and a general neural network branch. The spatial branch astutely identifies spatial dependencies, while the temporal branch brings into focus temporal dependencies residing within sequential hand skeleton data. Concurrently, the general neural network branch enriches the architecture by amplifying its generalization capabilities and robustness. These branches collaboratively weave together, synthesizing and contributing to the final feature integration, thereby guiding the model to interpret and recognize diverse sign language gestures and patterns astutely

fully connected layer within the neural network. We developed the system as an advancement system of the DG-STA [24], dynamic hand gesture recognition system [11]. Our architecture comprises three branches:

- The graph-based spatial branch primarily enhances hand gesture representation within the spatial context using spatial masking with multi-head attention, commonly referred to as the spatial feature [24].
- The graph-based temporal branch focuses on enhancing hand gesture representation within the temporal context using temporal masking with multi-head attention, known as the temporal feature. These features are extracted from the sequence of 3D whole-body skeleton information.
- The general neural network branch. Initially, we employ neural networks NN1 and NN2 to generate a fully con-

nected skeleton graph based on the original 3D joint coordinates of the skeleton, as shown in Fig. 4b and c.

Subsequently, attention-based branches are utilized to learn the edge weights and node features from the previous NN1 block, producing enhanced features for the spatial and temporal domains. Following this, the generated features from each branch are collected and concatenated to create the final feature vector. This concatenated feature is then average-pooled into a vector, and a fully connected layer is applied for classification. Branches (i) and (ii) involve the position embedding, attention model, and masking operation. Let the output of NN1 be denoted as F_1, representing the initial feature. We introduce A_S, as the spatial attention model and A_T, as the temporal attention model, both of which are used to derive spatial and temporal domain features from the skeleton dataset. These attention models take the form of multi-head attention. A_S, initially takes the initial node features F_1, as input, updating them to encode spatial information, resulting in the spatial feature. It is then projected into NN1, producing the enhanced spatial contextual feature within the first branch, denoted as F_s In the second branch A_T takes F_1, as input and updates it to encode temporal information. After being projected into NN1, it produces the temporal feature, denoted as F_t. In the third branch, the general neural network NN2 [11], takes the original hand skeleton dataset as input and generates the output feature F_R. Subsequently, we concatenate the output features from the first, second, and third branches using Eq. (5), resulting in the final output features.

$$F_{\mathrm{final}} = \mathrm{concate}[F_\mathrm{S}, F_\mathrm{T}, F_\mathrm{R}] \tag{5}$$

4.1 Graph-based neural network branch

This study employs two branches as graph-based attention branches to capture spatial and temporal dependencies within the destination grid. One branch is dedicated to extracting spatial information [24], while the other focuses on extracting temporal information, all based on the sequential 3D joint coordinate dataset of the skeleton. The primary objective of these branches is to dynamically adapt the unified graph based on different actions, optimizing both edges and nodes. Consequently, this mechanism allows our model to achieve action-specific graphs. To mitigate the computational complexity associated with the graph branches, we incorporate a masking operation within the attention block, both in spatial and temporal contexts. The primary purpose of these two branches is to enhance spatial and temporal dependency information after encoding within the attention block.

4.1.1 Skeleton-based graph initialization

The input skeleton data consists of T frames in each sequence, with N key points extracted from each frame representing the body, face, and hand skeleton. We consider $G = (V, E)$, as a fully connected graph for a video sequence, where V and E, represent the sequences of skeleton points and connections between these points, respectively [11, 24]. Each node or sequence of skeleton points can be defined as $V = \{v_{(t,p)}, \| \, t = 1, ..., T, p = 1, ..., N\}$.

For a given time frame t and joint skeleton p, it can be represented as $v_{(t,p)}$, with its corresponding feature denoted as $f_{(t,i)}$. Similarly, the features of all nodes for a specific time frame t can be expressed as $F = \{f_{(t,p)}, \|, t = 1, ..., T, p = 1, ..., N\}$. Each node possesses a 3D coordinate, and we can extract three types of edge information using the attention model as follows:

- The graph edge between two nodes within the same time frame is considered a spatial edge, which can be written as $v(t, p) \rightarrow v(t, q)(p \neq q)$.
- The graph edge between different time frames is known as temporal and can be written as $v(t, p) \rightarrow v(k, q)(t \neq k)$.
- If the graph edge is itself, then it is known as a self-connected edge, which can be expressed as $v(t, p) \rightarrow v(t, p)(t, p)$.

4.1.2 Spatial and temporal attention module

One of the main objectives of the attention model is to compute the dependence among the nodes of the adjacency matrix, which is calculated depending on the dynamic association between nodes in the spatial and temporal domains [11, 24].

Normally, an attention-based transformer takes sequential data as input like $X \in R^{(N \times C)}$, which contains the element N and the number of channel C [24]. However, dynamic skeleton datasets usually have a time frame of T with N and C looking like $X \in R^{(N \times T \times C)}$. Time information is crucial in investigating the relationship between time and space. Wang et al. combined the element and time information to ignore the difference between time and space for using the skeleton data as sequential data as follows: $X \in \mathbb{R}^{(\bar{N} \times C)}$, where $\bar{N} = N \times T$ [27]. Because the spatial and temporal dimensions are different, it is not good to treat them equally. We calculated attention-based spatial and temporal information from the skeleton dataset.

Let the initial feature value of a node be $f_{(t,p)}$. Spatial attention of a specific head applied three softmax functions to map the initial feature into the key, query, and value vectors sequentially, as shown in Eq. (6) [24, 25, 66]:

$$Q^n_{(t,p)} = W^n_Q f_{(t,p)}, K^n_{(t,p)} = W^n_K f_{(t,p)}, V^n_{(t,p)} = W^n_V f_{(t,p)} \tag{6}$$

where key, query, value, and corresponding matrix are denoted by Q, K, V, and W. In the processing, a dot product is first calculated between the query and key vectors by the attention mechanism within the same time frame and normalized using a softmax activation function described in Eq. (7).

$$\begin{aligned} u^m_{(t,p)\in(t,q)} &= \frac{\langle Q^m_{(t,p)}, K^m_{(t,p)}\rangle}{\sqrt{d}} \\ \alpha^m_{(t,p)\in(t,q)} &= \frac{\exp\left(u^m_{(t,p)\in(t,q)}\right)}{\sum_{n=1}^{N} \exp\left(u^m_{(t,p)\in(t,n)}\right)} \end{aligned} \tag{7}$$

Here, the dot product between two nodes is denoted by < . >, and dimensions of each matrix K, Q and V, are denoted with d. In the same way, α denotes the attention weight between two nodes, which extracts the important information from them. In this stage, we used a masking operation to determine whether the attention model was used as a spatial or temporal domain. We block all the time information for the spatial domain by assigning 0, for the temporal edges with the spatial masking operation. As a result, the masking operation only passed the spatial domain edges, which is considered the spatial structure of the weighted skeleton graph. Finally, the attention head calculated the weighted sum of the value vectors with the same time steps, which is considered as a spatial feature vector for the specific skeleton node information using the following Eq. (8):

$$\bar{f}^m_{t,p} = \sum_{q=1}^{N} \left(\alpha^m_{(t,p)\in(t,q)} \cdot V^m_{t,q}\right) \tag{8}$$

where $\bar{f}$ denotes the attention feature for a single head for a specific node V, the spatial attention mechanism sends each node to the other simultaneously. It aggregates the received information based on the edge weight. By following the same procedure, it produces eight heads and produces the final feature vector $\bar{f}$, which is the output of the spatial attention model A_S, by following Eq. (9).

$$\bar{f}_{t,p} = \text{Concat}[\bar{f}^1_{t,p}, \bar{f}^2_{t,p}, \bar{f}^3_{t,p}, \ldots, \bar{f}^H_{t,p}] \tag{9}$$

where the number of heads is denoted by H, in our case $H = 8$, we follow the same procedure to extract the enhanced temporal domain feature A_T. In the temporal attention model, we blocked all the spatial edge information by setting 0 using a temporal masking operation. Figure 5 shows the internal structure of the attention model with the masking operation. The spatial or temporal domain depends on the type of masking operation.

4.1.3 Spatial temporal mask operation

The detailed mechanism of the mask operation is demonstrated in Fig. 6, which is mainly used here to reduce the computational complexity is demonstrated [11, 24, 25]. The attention model we reported in the previous Sect. 4.1.2 where for a specific head, we multiplied Q and K matrices to produce weight matrix W, for each node using the following scaled dot products formula Eq. (10):

$$W = Q \bigotimes K^T \tag{10}$$

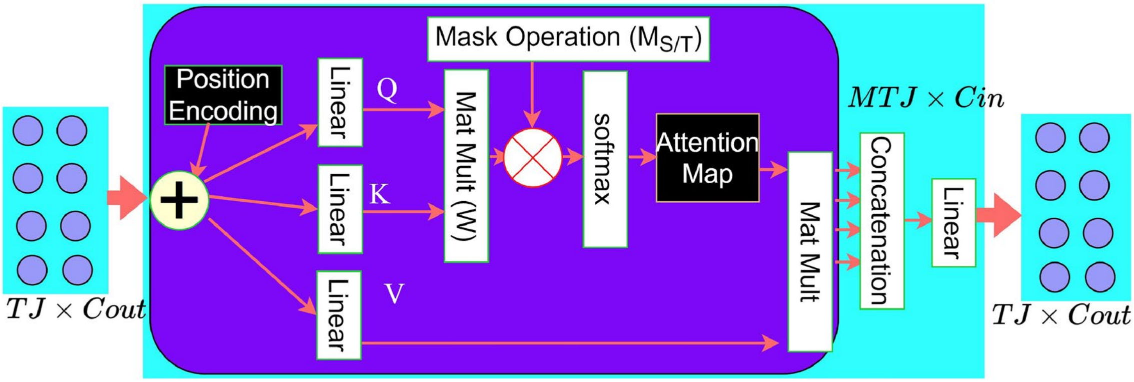

Fig. 5 Illustration of the multi-head self-attention (MHSA) map architecture with masking operation: This architecture derives node dependence in the adjacency matrix from dynamic spatial/temporal domain associations [11]. Firstly, the position embedding is calculated and added to the skeleton input dataset. This then generates key, query, and value vectors (K, Q, V, and their matrices, W) through the linear layer. Following dot product computations, it is normalized with softmax activation. The masking operation negates temporal information for the spatial domain, allowing only spatial domain edges. The attention head calculates the weighted sum of value vectors, which is repeated across eight heads to produce a final feature vector enriched with spatial and temporal context

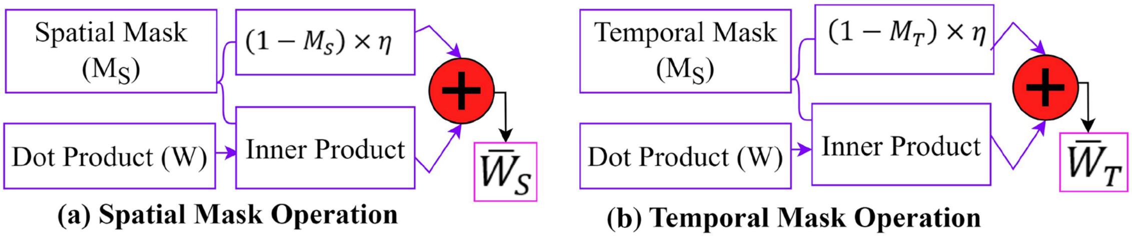

Fig. 6 Spatial–temporal masking operation in attention model. The attention model multiplies matrices Q and K to form weight matrix W via scaled dot products, subsequently manipulating W's elements for spatial mask operation. Essential principles include maintaining spatial domain values while nullifying others by assigning 1 to spatial positions and 0 to temporal, effectively allowing only spatial position values to pass. Manual adjustments of the weight matrix enable the performance of both spatial and temporal masks [11, 24, 25]

In Eq. (10), first, transpose the key matrix, which is denoted by T and then multiply with query Q, which is denoted by $\bigotimes$. Then, we set the value of each element in the weight matrix W for the spatial mask operation.

The main concept of the setting value is to keep the spatial domain unchanged and set 0 for the other element. Here, for the spatial mask cases, we assign 1 for the spatial position and set 0 for the temporal position, aiming to block the temporal value and only pass the spatial position value. We updated the value of the weight matrix by manual setting aiming to perform spatial mask M_{S} and temporal mask M_{T}, using the following formula Eqs. (11) and (12).

$$\bar{W}_S = \phi(\ W \odot M_{\mathrm{S}} + (\ 1 - M_{\mathrm{S}})\ \times \eta) \tag{11}$$

$$\bar{W}_T = \phi(\ W \odot M_{\mathrm{T}} + (\ 1 - M_{\mathrm{T}})\ \times \eta) \tag{12}$$

In the equations, $\odot$, denotes the inner product, η, represents the spatial or temporal edge, and we assign-9×10^5, for this. The mask variable M_{S} is a spatial mask containing one edge spatial or self-connected edge otherwise 0. M_{T} is a temporal mask variable that assigns one if the edge is temporal or self-connected; otherwise, it is 0. Equations (10)–(12) is implemented using Eq. (7) efficiently to compute spatial or temporal edge weight.

4.1.4 Position embedding

The skeleton joint fed into the neural network as a tensor because there are no predefined structures to define the identity of the individual tensor. Traditional methodologies like LSTM and GRUs process the skeleton information sequentially, whereas our architecture is one kind of transformer that will not process the skeleton joint sequentially [11, 24, 25]. By applying the spatial and temporal position embedding, we provided a unique marker for every joint of the body, face and hand skeleton. To do this, we computed a sine and cosine for different frequencies using the following formula defined in Eq. (13):

$$\begin{aligned} P_E(ps, 2p) &= \sin\left(ps/1000^{2p/C_{\mathrm{in}}}\right) \\ P_E(ps, 2p) &= \cos\left(ps/1000^{2p/C_{\mathrm{in}}}\right) \end{aligned} \tag{13}$$

The ps denotes the position of the elements, and p represents the position encoding vector dimension [24, 25]. One of the challenging issues in the skeleton data is that it contains space and time information, and we did not unify these two pieces of information. Still, we extracted spatial and temporal position encoding for this. To do this, the spatial position is considered the N vector, where an individual vector contains a set of joints for a single frame sequentially. Moreover, the temporal position contained $N \times T$, individual vectors, and an individual vector is composed of a corresponding node in the skeleton graph. We encoded these vectors with the same joint in a different frame. After generating the position, we added the output of $NN1$, considered the initial feature vector and fed it into the spatial and temporal block. For the spatial and temporal branch, we followed the following procedure sequentially, shown in Eqs. (14) and (15):

$$\bar{f}_{ST(t,p)} = A_S\left(f_{(t,p)} + P^{\mathrm{S}}_{(t,p)}\right) \tag{14}$$

$$\bar{f}_{TS(t,p)} = A_T\left(f_{(t,p)} + P^{\mathrm{T}}_{(t,p)}\right) \tag{15}$$

Equations (14) and (15) produces the final feature $\bar{f}_{\mathrm{S}(t,p)}$, $\bar{f}_{\mathrm{T}(t,p)}$, of spatial and temporal branches in parallel for a specific node $v_{(t,p)}$.

The $p - th$ hand joint for the $t - th$ time step is represented here by $P^{\mathrm{S}}_{(t,p)}$, and $P^{\mathrm{T}}_{(t,p)}$, in parallel for the spatial and temporal. The dimension of the embedding feature is the same as the initial feature $f_{(t,p)}$.

4.2 General neural network branch

In the third branch, the general neural network, we employed a neural network to project the 3D coordinate of a hand joint into an initial node feature of 128 dimensions, as shown in Fig. 4c. We included a fully connected layer in the network, one ReLU activation, a normalization layer, a global average pooling layer, a dropout layer, and a padding mechanism [11]. A newer version of the fully connected layer in the PyTorch allows acceptance of the N–D input tensor and specific constraints and is applied in the last dimension.

5 Experimental evaluation

We evaluated the proposed multibranch STA model using three datasets: MSL, PSL, and ASLLVD. The MSL dataset contains video sequences for 30 hand gesture signs. In the evaluation, we assessed the performance of individual body parts and their combinations. The PSL dataset consists of 49 class labels, of which 37 derive from the alphabet and 12 represent words. It includes the 3D coordinates of 5 body joints, 20 face points, and 21 joints each for the left and right hands. ASLLVD comprises 2745 ASL sign class labels (used in the study), recorded from multiple synchronized videos, and is divided into 7798 training videos and 1950 testing videos.

5.1 Experimental setting

We implemented the model using the PyTorch platform and divided the dataset into training and testing sets in accordance with a previous paper. The models were trained for 500 epochs on data from 300 patients, utilizing the cross-entropy loss function [67, 68], a learning rate of 0.001, and the Adam optimizer [69]. We randomly selected eight frames for each video sequence as input and applied a data augmentation technique that involved adding noise, shifting, scaling, and time interpolation. Subsequently, we aligned all the skeleton sequences by subtracting them from the palm position of the first frame. Although there have been some predefined training and testing cases in the MSL and ASLLVD datasets, we partitioned the PSL dataset with 80% for training and 20% for testing.

5.2 Ablation study

In our ablation study, we proposed a model structured with two multi-head self-attention (MHSA) modules distributed across two streams: a spatial attention stream and a temporal attention stream. Within these streams, we incorporated a blend of spatial and temporal attention modules supplemented by several neural network (NN) modules. Our primary objective was to determine the optimal count of MHSA modules, given their spatial and temporal contextual characteristics, in tandem with the NN modules to maximize sign language recognition accuracy. A summarized view of our findings, alongside comparisons to existing models, is captured in Table 2. Building upon the foundational concepts from prior studies, notably [19] and [11], we drew insights into the configuration of MHSA and NN modules. For instance, the research presented in [19] utilized an architecture featuring 2 MHSAs paired with 3 NN modules, framing it within a spatial–temporal attention paradigm. Meanwhile, the [11] study engaged 4 MHSAs complemented by 7 NN modules, accentuating the interplay of spatial–temporal attention, temporal–spatial attention, and NN-based residual connections. To address computational efficiency and long-time-dependency challenges, we also explored configurations deploying a single MHSA for spatial attention, complemented by 2 NN, modules and a corresponding configuration for temporal attention. These configurations are seen in the single stream spatial attention (3DGCN) and single stream temporal attention (3DGCN) models, each achieving 96.22% accuracy in MSL and 87.50% in PSL. However, our prime achievement was found with the proposed model. Here, we leveraged 2 MHSAs in a parallel architecture emphasizing both spatial and temporal contextual feature enhancements bolstered by 5 NN modules. This design culminated in an impressive accuracy of 99.96% for MSL and 91.66% for PSL, distinctly outperforming other models.

To provide a clearer context, here's a concise summary of the ablation study demonstrated in Table 2:

Table 2 Strategic ablation study on MHSA and NN module variations for spatial and temporal feature enhancement: This study examines the variations in multi-head self-attention (MHSA) and neural network (NN) module counts across different model architectures

Method Name	No. of MHSA	No. of NN	Accuracy MSL [%]	Accuracy PSL[%]
Single stream spatial–temporal attention(STA) (ST-GCN) [19]	2	3	97.00	90.00
Multi-stream with STA and TSA (Multi-stream ST-GCN) [11]	4	7	99.69	90.00
Multi-stream with STA and STA (Multi-stream ST-GCN)	4	7	99.69	90.00
Single stream spatial attention (First stream)	1	2	96.22	87.50
Single stream temporal attention (Second stream)	1	2	96.22	87.50
Proposed model	2	5	99.96	92.00

Notably, configurations vary, with the single stream spatial–temporal attention (ST-GCN) using 2 MHSAs and 3 NNs, while the multi-stream ST-GCN employs 4 MHSAs and 7 NNs, leading to diverse performance results. The proposed model is configured with 2 MHSAs and 5 NNs that generated high performance

- *Single stream spatial–temporal attention (ST-GCN)* [19]: 2 MHSAs, 3 NNs, performance accuracy of MSL: 97.00%, performance accuracy of PSL: 90.00%
- *Multi-stream with STA and TSA (Multi-stream ST-GCN)* [11]: 4 MHSAs, 7 NNs, performance accuracy of MSL: 99.69%, performance accuracy of PSL: 90.00%
- *Multi-stream with STA and STA (Multi-stream ST-GCN):* 4 MHSAs, 7 NNs, performance accuracy of MSL: 99.69%, performance accuracy of PSL: 90.00%
- *Single stream spatial attention (First stream):* 1 MHSA, 2 NNs, performance accuracy of MSL: 96.22%, performance accuracy of PSL: 87.50%
- *Single stream temporal attention (Second stream):* 1 MHSA, 2 NNs, performance accuracy of MSL: 96.22%, performance accuracy of PSL: 87.50%
- *Proposed model:* 2 MHSAs, 5 NNs, performance accuracy of MSL: 99.96% (Table 3), performance accuracy of PSL: 91.66%

These results unequivocally underscore the superior performance of our proposed configuration compared to existing architectures.

5.3 Performance accuracy

We evaluated the proposed model with three benchmark whole-body pose-based datasets, and the performance of each dataset with state-of-the-art comparison is given below.

5.3.1 Performance accuracy with MSL dataset

We evaluated our model using the Mexican Sign Language dataset, which contains 3D joint coordinates for the body, face, and hands. Table 3 displays the performance accuracy of our proposed model with this dataset, considering both individual and combined features. Initially utilizing only body information, our model achieved 81.33% accuracy, 82.00% precision, and 89.00% recall. When utilizing solely face information, the model yielded 49.11% accuracy, 57% precision, and 52.27% recall. Remarkably, employing only hand information, our model attained 100% accuracy and recall. Further, by combining body information with face information, an accuracy of 80.22% was achieved. Similarly, by integrating hand information with face information and hand information with body information, in both instances, we attained 99.71% accuracy. When combining all three types of information, our model achieved accuracies of 99.96% (Table 3), 99.69% (Table 3), and 99.92% for accuracy, precision, and recall, respectively. This showcases the model's versatile predictive prowess and how various feature integrations influence recognition accuracy.

Table 3 Evaluating varied feature combinations in the proposed model

Combinations of features	Accuracy	Precision	Recall	No. of joint coordinates
All_Features	99.96	99.69	99.92	67
Body	81.33	82.00	89.00	5
Face	49.11	57.00	52.27	20
Hand only	99.71	99.42	99.42	42
Body Face	80.22	78.16	79.07	25
Face Hand	99.71	99.71	99.42	62
Body Hand	99.71	99.71	99.77	47

Highlighting the paramount performance using 'All Features' with accuracy, precision, and recall at 99.96%, 99.69%, and 99.92%, respectively, and utilizing 67 joint coordinates the data also delineates varying results for other feature sets and combinations of 'Body,' 'Face,' 'Hand only,' 'Body Face,' 'Face Hand,' and 'Body Hand' gestures

Table 4 displays a comparison of the proposed system with Existing RNN and the state-of-the-art models for the MSL dataset in terms of individual for same and combined gesture datasets.

The authors of the system [3] employed various model architectures, including recurrent neural networks (RNN) like long short-term memory (LSTM) and gated recurrent units (GRU), to classify the sign language dataset.

As per Table 4, their method achieved a peak performance accuracy of 96.44% for all features, whereas our proposed model attained 99.96% (Table 3) accuracy for the combined feature. While they achieved 96.00% accuracy using only hand information, our model reached 99.71%. In evaluations using only body and face information, their model attained 3.55% and 63.55% accuracy, respectively, while our model achieved sequential accuracies of 81.33% and 49.11%. Furthermore, we crafted a project for the ST-GCN [19] and 3DGCN [29] methods with the aim of evaluating, comparing, and demonstrating the superiority of our proposed model. We implemented the proposed method on the MSL datasets since direct performance results from existing studies were unavailable. This showcases our method's superior performance and effectiveness in sign language recognition,

Table 4 Benchmarking the proposed model's efficacy against state of the art methods on the MSL dataset. The table accentuates the exceptional 99.96% accuracy of our proposed model, utilizing body, face, and hand gesture data, against renowned methods like RNN (96.44%), ST-GCN (96.69%), and 3DGCN (99.11%)

Method name	Dataset name	Gestures	Performance[%]
RNN [3]	MSL	Body, Face, Hand	96.44
ST-GCN [19]	MSL	Body, Face, Hand	96.69
3DGCN [29]	MSL	Body, Face, Hand	99.11
Proposed Model	MSL	Body, Face, Hand	99.96

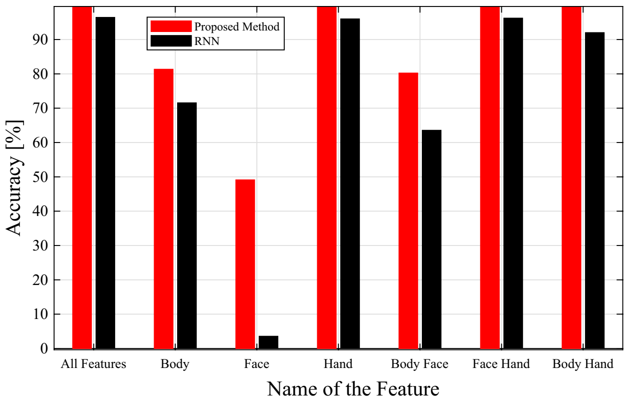

Fig. 7 Gesture-wise comparison of the proposed model versus existing RNN models: existing RNN models, including LSTM and GRU, reached a peak accuracy of 96.44% for sign language classification. Our model achieved 99.96%. Specifically, with just hand information, they scored 96.00% versus our 99.71%. Using only body or face data, they registered 3.55% and 63.55%, while ours recorded 81.33% and 49.11%. Combining all features, they reached 96.44%, compared to our 99.96% (Table 3), underscoring the differential effectiveness between approaches

Table 5 Evaluating the proposed model on the PSL pose dataset: performance metrics varied between 87.50% and 92.00% based on whether we used only ’Hand’ gestures or combined 'Body, Face, Hand’ gestures. Further differentiation was made between ’PSL Word’ and ’PSL Alphabet’ classifications

Model	Dataset type	Gestures	Performance[%]
Proposed model	PSL Word	Hand	87.50
Proposed model	PSL Word	Body, Face, Hand	91.66
Proposed model	PSL Alphabet	Hand	87.50
Proposed model	PSL Alphabet	Body, Face, Hand	92.00

highlighting variations in methodology and results within the MSL dataset. By doing so, we aim to offer a more comprehensive comparison to existing research. As a result, our model consistently achieved high accuracy across different data types.

Figure 7 visualizes the gesture-wise comparison with the existing RNN models.

5.3.2 Performance accuracy with PSL dataset

We employed another whole-body pose-based PSL sign language dataset to assess the proposed model. This dataset comprises two types of data: the word whole-body pose dataset and the Alphabet whole-body pose dataset. Table 5 showcases the performance of the PSL dataset for both word and alphabet cases. Similar to the procedure with the MSL dataset, we initially conducted experiments utilizing only hand pose information, which encompassed 42 key information points for the left and right hands. Our proposed model achieved accuracies of 62.50% and 87.50% for the PSL word and PSL alphabet datasets, respectively, when using only hand information. Subsequently, we experimented with our model using whole-body features, including information from two hands, face, and body. Our model attained an accuracy of 88.00% for the PSL word whole-body information and 92.00% for the whole-body information of the PSL alphabet dataset. We illustrated the label-wise precision, recall, F1-score, and accuracy in Fig. 8, focusing on the sign language word dataset and observing the performance of the alphabet dataset. The illustration reveals a detailed breakdown of precision, recall, F1-score, and accuracy for each label class in the PSL dataset. Figure 8 shows that precision is almost 100.00% for seven classes, 87.00% for one

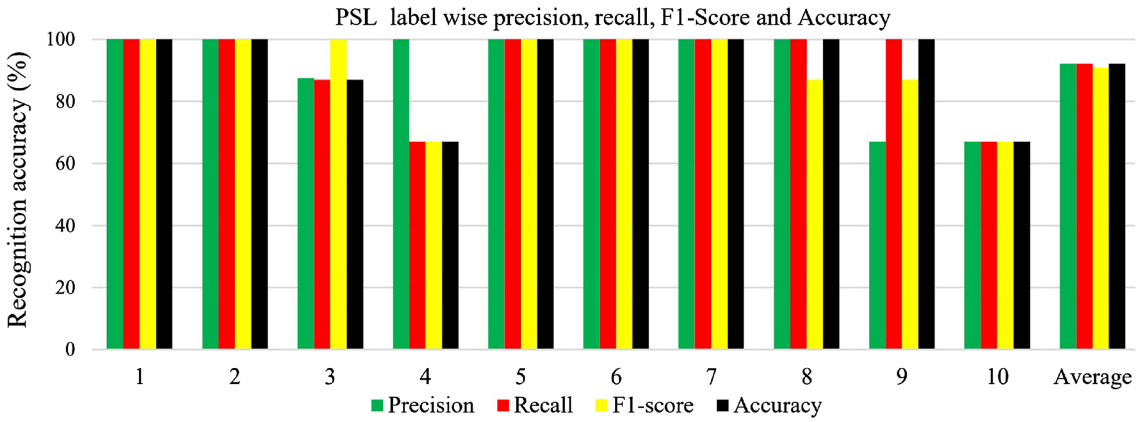

Fig. 8 Label-wise precision, recall, and F1-score for the PSL dataset. Seven classes exhibit nearly impeccable precision, recall, and accuracy, hovering around 100%. A distinct class demonstrates 87% precision, while two others attain 67% in precision and recall, influencing their respective F1 scores. Recall further delineates, securing 84% for an isolated class, and mirroring precision for the remaining seven. The F1-score also distinctly attains 87% for two classes, while preserving nearly 100% for six classes

Table 6 Comparative performance evaluation on the PSL dataset contrasts the performance of the proposed model against ST-GCN and 3DGCN methodologies, each utilizing body, face, and hand gestures on the PSL Alphabet dataset

Method name	Dataset name	Gestures	Performance[%]
ST-GCN [19]	PSL	Body, Face, Hand	90.00
3DGCN [29]	PSL	Body, Face, Hand	87.50
Proposed model	PSL	Body, Face, Hand	92.00

The proposed model notably achieves the highest accuracy at 92.00%, outperforming ST-GCN (90.00%) and 3DGCN (87.50%)

Table 7 Benchmarking against leading models on the ASLLVD dataset: the table showcases a performance comparison, revealing the proposed model's leading accuracy of 26.05% on the ASLLVD dataset, juxtaposed with the results of established methodologies—ST-GCN at 16.48% and 3DGCN at 25.05%

Method name	Dataset name	Performance[%]
ST-GCN [19]	ASLLVD	16.48
3DGCN [29]	ASLLVD	25.05
Proposed model	ASLLVD	26.00

class, and 67.00% for two classes. Recall achieved 67% for two classes, 84.00% for one class, and nearly 100.00% for the remaining seven classes. Similarly, the F1-score achieved 67.00% for two classes, 87.00% for two classes, and nearly 100.00% for the remaining six classes. Lastly, accuracy achieved 67.00% for two classes, 84.00% for one class, and 100.00% for seven classes. Ultimately, the nuanced exploration of these metrics offers an intricate understanding of the model's label-wise performance across various evaluative parameters. A comparison table for this dataset is provided, featuring existing state-of-the-art methods in the field of action recognition, as no published papers focusing specifically on this dataset were found.

Table 6 presents a comparison of our model to state-of-the-art performances on the PSL dataset, providing demonstrative insight into its superior efficacy and precision in gesture recognition compared to established approaches. To validate the superiority of our proposed model, we implemented and evaluated the ST-GCN [19] and 3DGCN [29] methods since direct performance results for the PSL datasets were unavailable in existing studies. As demonstrated in Table 6, our model outperforms the aforementioned methods. This approach broadens the comparative scope with current research.

5.3.3 Performance accuracy with ASLLVD dataset

The performance accuracy of the proposed model, evaluated using the ASLLVD dataset, is displayed in Table 7. Unlike many existing models that were evaluated using only a subset of classes, our evaluations utilized all available classes. The proposed model yielded an accuracy of 26.05%, computed as the average of the maximum batch across all epochs. We juxtaposed the performance of our model with that of ST-GCN [19] and 3DGCN [29]. In the most recent method for ASLLVD data proposed by Hammadi et al., a graph-based convolutional neural network was developed. It integrates a few separable 3DGCN layers and employs only the spatial attention mechanism [29]. While they secured an accuracy of 25.05% for all classes and exhibited strong performance for certain individual classes, the method was not as effective across the entire dataset. Table 7 illustrates the performance accuracy of the ST-GCN model, which achieved 16.48%. When solely utilizing spatial context information with the 3DGCN model, an accuracy of 25.05% was achieved, whereas the proposed model reached an accuracy of 26.00%.

5.4 Disussion

In our study, we employed several technologies to address specific challenges in hand gesture recognition. In summary, our model described encompasses three crucial streams for conducting experiments in skeleton-based hand gesture recognition: joint skeleton information, position embedding, and multi-head self-attention (MHSA), each with a unique role and contribution to the overall framework. Joint skeleton information is paramount for balancing user privacy and data dimensionality reduction, ensuring accurate gesture recognition without forfeiting data protection. Position embedding introduces a spatial and temporal identifier for each joint within the skeletons (body, face, and hand), elevating the precision of the recognition process by characterizing each joint's spatial–temporal dynamics, thus enhancing gesture understanding. MHSA is crucial in capturing spatial dependencies and multi-scale features, enabling the recognition of complex gestures and bolstering recognition capabilities and transparency through transfer learning. Further, the spatial and temporal contexts of MHSA-based features were explored through ablation studies to assess their impact on performance and result accuracy, providing insights into their roles within the architecture. Spatial–temporal masking operation, applied across branches, curtails the computational complexity of the self-attention block, enhances system efficiency, and enables the direct modelling of joint dependencies without translating skeletal data into images or graphs. Lastly, general deep learning-based residual connections leverage CNNs to facilitate autonomous feature extraction, ensuring end-to-end learning and bolstering accuracy in gesture recognition. The inclusion of residual connections mitigates the vanishing gradient problem, aiding the creation of deeper architectural frameworks while stabilizing gradients, altogether showcasing a diverse and multifaceted approach to gesture analysis. Each hand gesture recognition system technology specifically enhances accuracy, efficiency, robustness, and privacy. The proposed model notably outperforms existing state-of-the-art approaches, especially when managing the complexities and challenges of the MSL, PSL, and ASLLVD datasets.

6 Conclusion

In the study, we proposed a spatial and temporal attention model with general neural network-based SLR recognition. Our model consisted of three branches: spatial attention to extracting spatial context representation of the gesture, temporal attention to extracting spatial context representation of the gesture and the general neural network model branch, which included a fully connected layer as a skip connection. The experimental output with three datasets using whole-body features demonstrated that we achieved our goal with the proposed system. It produced a high performance compared to the state-of-the-art model and reduced computational complexity by taking only eight frames as input from the 20 frames. Our model can be considered a general framework that may be used further for other sign language or multi-sign language classification. In the future, we will extend the architecture and experiment with multi-sign language recognition with large-scale datasets.

Declarations

Conflict of interest The authors declare no conflict of interest.

References

1. Obi Y, Claudio KS, Budiman VM, Achmad S, Kurniawan A (2023) Sign language recognition system for communicating to people with disabilities. Proc Comput Sci 216:13–20. https://doi.org/10.1016/j.procs.2022.12.106
2. Manning V, Murray JJ, Bloxs A (2022) Linguistic human rights in the work of the world federation of the deaf. In: The handbook of linguistic human rights. John Wiley & Sons, Ltd, pp 267–280
3. Mejía-Peréz K, Córdova-Esparza DM, Terven J, Herrera-Navarro AM, García-Ramírez T, Ramírez-Pedraza A (2022) Automatic recognition of Mexican Sign Language using a depth camera and recurrent neural networks. Appl Sci 12(11):5523
4. Miah ASM, Shin J, Hasan MAM, Rahim MA (2022) Bensignnet: Bengali sign language alphabet recognition using concatenated segmentation and convolutional neural network. Appl Sci 12(8):3933
5. Zhang Z, Li Z, Liu H, Cao T, Liu S (2020) Data-driven online learning engagement detection via facial expression and mouse behavior recognition technology. J Educ Comput Res 58(1):63–86
6. Rajan RG, Leo MJ (2020) American sign language alphabets recognition using hand crafted and deep learning features. In: 2020 international conference on inventive computation technologies (ICICT). IEEE, pp 430–434
7. Kudrinko K, Flavin E, Zhu X, Li Q (2020) Wearable sensor-based sign language recognition: a comprehensive review. IEEE Rev Biomed Eng 14:82–97
8. Sharma S, Singh S (2020) Vision-based sign language recognition system: a comprehensive review. In: 2020 international conference on inventive computation technologies (ICICT). IEEE, pp 140–144
9. Shin J, Musa Miah AS, Hasan MAM, Hirooka K, Suzuki K, Lee H-S, Jang S-W (2023) Korean Sign Language recognition using transformer-based deep neural network. Appl Sci 13(5):3029
10. Miah ASM, Hasan MAM, Shin J, Okuyama Y, Tomioka Y (2023) Multistage spatial attention-based neural network for hand gesture recognition. Computers 12(1):13
11. Miah ASM, Hasan MAM, Shin J (2023) Dynamic hand gesture recognition using multi-branch attention based graph and general deep learning model. IEEE Access 11:4703
12. Gu Y, Sherrine Wei W, Li X, Yuan J, Todoh M (2022) American Sign Language alphabet recognition using inertial motion capture system with deep learning. Inventions 7(4):112
13. Abdullahi SB, Chamnongthai K (2022) American sign language words recognition of skeletal videos using processed video driven multi-stacked deep LSTM. Sensors 22(4):1406
14. De Smedt Q, Wannous H, Vandeborre JP, Guerry J, Le Saux B, Filliat D (2017) Shrec’17 track: 3d hand gesture recognition using a depth and skeletal dataset. In: 3DOR-10th Eurographics workshop on 3D object retrieval, pp 1–6
15. Li C, Zhang X, Liao L, Jin L, Yang W (2019) Skeleton-based gesture recognition using several fully connected layers with path signature features and temporal transformer module. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, pp. 8585–8593
16. Hou J, Wang G, Chen X, Xue JH, Zhu R, Yang H (2018) Spatial-temporal attention res-TCN for skeleton-based dynamic hand gesture recognition. In: proceedings of the European conference on computer vision (ECCV) workshops, pp 0–0
17. Lai K, Yanushkevich SN (2018) Cnn+ rnn depth and skeleton based dynamic hand gesture recognition. In: 2018 24th International Conference on Pattern Recognition (ICPR). IEEE, pp 3451–3456
18. Yan S, Xiong Y, Lin D (2018) Spatial temporal graph convolutional networks for skeleton-based action recognition. In: proceedings of the AAAI conference on artificial intelligence, vol 32
19. de Amorim, CC, Macêdo D, Zanchettin C (2019) Spatial–temporal graph convolutional networks for sign language recognition. In: artificial neural networks and machine learning–ICANN 2019: workshop and special sessions: 28th international conference on artificial neural networks, Munich, Germany, September 17–19, 2019, Proceedings 28, pp 646–657 Springer
20. Jiang S, Sun B, Wang L, Bai Y, Li K, Fu Y (2021) Skeleton aware multi-modal sign language recognition. In: proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 3413–3423
21. Jiang S, Sun B, Wang L, Bai Y, Li K, Fu Y (2021) Sign language recognition via skeleton-aware multi-model ensemble. arXiv preprint arXiv:2110.06161
22. Dai Z, Yang Z, Yang Y, Carbonell J, Le QV, Salakhutdinov R (2019) Transformer-xl: attentive language models beyond a fixed-length context. arXiv preprint arXiv:1901.02860
23. Fu J, Liu J, Tian H, Li Y, Bao Y, Fang Z, Lu H (2019) Dual attention network for scene segmentation. In: proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 3146–3154
24. Chen Y, Zhao L, Peng X, Yuan J, Metaxas DN (2019) Construct dynamic graphs for hand gesture recognition via spatial-temporal attention. arXiv preprint arXiv:1907.08871

25. Cheng K, Zhang Y, Cao C, Shi L, Cheng J, Lu H (2020) Decoupling gcn with dropgraph module for skeleton-based action recognition. In: European conference on computer vision, Springer, pp 536–553
26. Yan S, Xiong Y, Lin D (2018) Spatial temporal graph convolutional networks for skeleton-based action recognition. In: Thirty-second AAAI Conference on Artificial Intelligence
27. Hou J, Wang G, Chen X, Xue JH, Zhu R, Yang H (2018) Spatial-temporal attention res-TCN for skeleton-based dynamic hand gesture recognition. In: proceedings of the European conference on computer vision (ECCV) workshops, pp 0–0
28. Zhou K, Huang X, Li Y, Zha D, Chen R, Hu X (2020) Towards deeper graph neural networks with differentiable group normalization. Adv Neural Inf Process Syst 33:4917–4928
29. Al-Hammadi M, Bencherif MA, Alsulaiman M, Muhammad G, Mekhtiche MA, Abdul W, Alohali YA, Alrayes TS, Mathkour H, Faisal M (2022) Spatial attention-based 3d graph convolutional neural network for sign language recognition. Sensors 22(12):4558
30. Altuwaijri GA, Muhammad G, Altaheri H, Alsulaiman M (2022) A multi-branch convolutional neural network with squeeze-and-excitation attention blocks for eeg-based motor imagery signals classification. Diagnostics 12(4):995
31. Amin SU, Altaheri H, Muhammad G, Abdul W, Alsulaiman M (2021) Attention-inception and long-short-term memory-based electroencephalography classification for motor imagery tasks in rehabilitation. IEEE Trans Ind Inf 18(8):5412–5421
32. Miah ASM, Hasan MAM, Shin J, Rahim MA, Okuyama Y (2023) Rotation, translation and scale invariant sign word recognition using deep learning. Comput Syst Sci Eng 44(3):2521–2536
33. Miah ASM, Hasan MAM, Nishimura S, Shin J (2024) Sign Language recognition using graph and general deep neural network based on large scale dataset. IEEE Access 9(10):1–1. https://doi.org/10.1109/ACCESS.2024.3372425
34. Miah ASM, Shin J, Hasan MAM, Molla MKI, Okuyama Y, Tomioka Y (2022) Movie oriented positive negative emotion classification from eeg signal using wavelet transformation and machine learning approaches. In: 2022 IEEE 15th international symposium on embedded multicore/many-core systems-on-chip (MCSoC), pp 26–31. https://doi.org/10.1109/MCSoC57363.2022.00014
35. Miah ASM, Shin J, Islam MM, Abdullah Molla MKI (2022) Natural human emotion recognition based on various mixed reality(mr) games and electroencephalography (eeg) signals. In: 2022 IEEE 5th Eurasian conference on educational innovation (ECEI), pp 408–411 https://doi.org/10.1109/ECEI53102.2022.9829482
36. Piskozub J, Strumillo P (2022) Reducing the number of sensors in the data glove for recognition of static hand gestures. Appl Sci 12(15):7388
37. Ruvalcaba D, Ruvalcaba M, Orozco J, López R, Cañedo C (2018) Prototipo de guantes traductores de la lengua de señas mexicana para personas con discapacidad auditiva y del habla. In: Memorias del Congreso Nacional de Ingeniería Biomédica, vol 5, pp 350–353
38. Saldaña González G, Cerezo Sánchez J, Bustillo Díaz MM, Ata Pérez A (2018) Recognition and classification of sign language for spanish. Computación y Sistemas 22(1):271–277
39. Varela-Santos H, Morales-Jiménez A, Córdova-Esparza D-M, Terven J, Mirelez-Delgado FD, Orenday-Delgado A (2021) Assistive device for the translation from Mexican Sign Language to verbal language. Computación y Sistemas 25(3):451–464
40. Hernández EC, Orozco JJM, Lozada DM, Saucedo AZ, Flores AB, López VEB, Raggi SEA (2018) Sistema de reconocimiento de vocales de la lengua de señas mexicana. Pistas Educativas 39(128), Technologico nacional de Mexico
41. Estrivero-Chavez C, Contreras-Teran M, Miranda-Hernandez J, Cardenas-Cornejo J, Ibarra-Manzano M, Almanza-Ojeda D (2019) Toward a Mexican Sign Language system using human computer interface. In: 2019 international conference on mechatronics, electronics and automotive engineering (ICMEAE). IEEE, pp 13–17
42. Unutmaz B, Karaca AC, Güllü MK (2019) Turkish sign language recognition using kinect skeleton and convolutional neural network. In: 2019 27th signal processing and communications applications conference (SIU). IEEE, pp 1–4
43. Raghuveera T, Deepthi R, Mangalashri R, Akshaya R (2020) A depth-based Indian sign language recognition using microsoft kinect. Sādhanā 45(1):1–13
44. Khan M, Siddiqui N (2020)Sign language translation in urdu/hindi through microsoft kinect. In: IOP conference series: materials science and engineering, vol 899. IOP Publishing, p 012016
45. Xiao Q, Qin M, Yin Y (2020) Skeleton-based Chinese Sign Language recognition and generation for bidirectional communication between deaf and hearing people. Neural Netw 125:41–55
46. Jing L, Vahdani E, Huenerfauth M, Tian Y (2019) Recognizing american sign language manual signs from rgb-d videos. arXiv preprint arXiv:1906.02851
47. Gutiérrez MM, Rojano-Cáceres JR, Patiño IEB, Pérez FJ (2016) Identificación de lengua de señas mediante técnicas de procesamiento de imágenes. Adv Intell Technol Appl 121(1):121–129
48. Solís F, Martínez D, Espinoza O (2016) Automatic Mexican Sign Language recognition using normalized moments and artificial neural networks. Engineering 8(10):733
49. Pérez LM, Rosales AJ, Gallegos FJ, Barba AV (2017) LSM static signs recognition using image processing. In: 2017 14th international conference on electrical engineering, computing science and automatic control (CCE). IEEE, pp 1–5
50. Morales EM, Aparicio OV, Arguijo P, Armenta RÁM, López AHV (2019) Traducción del lenguaje de señas usando visión por computadora. Res Comput Sci 148(8):79–89
51. Martinez-Seis B, Pichardo-Lagunas O, Rodriguez-Aguilar E, Saucedo-Diaz E-R (2019) Identification of static and dynamic signs of the Mexican Sign Language alphabet for smartphones using deep learning and image processing. Res Comput Sci 148(11):199–211
52. Solís F, Toxqui C, Martínez D (2015) Mexican sign language recognition using Jacobi–Fourier moments. Engineering 7(10):700
53. Cervantes J, García-Lamont F, Rodríguez-Mazahua L, Rendon AY, Chau AL (2016) Recognition of Mexican Sign Language from frames in video sequences. In: international conference on intelligent computing. Springer, pp 353–362
54. Adhikary S, Talukdar AK, Sarma KK (2021) A vision-based system for recognition of words used in Indian Sign Language using mediapipe. In: 2021 sixth international conference on image information processing (ICIIP), vol 6. IEEE, pp 390–394
55. Pigou L, Van Den Oord A, Dieleman S, Van Herreweghe M, Dambre J (2018) Beyond temporal pooling: recurrence and temporal convolutions for gesture recognition in video. Int J Comput Vision 126:430–439
56. Chen X, Gao K (2018) Denseimage network: video spatial-temporal evolution encoding and understanding. arXiv preprint arXiv:1805.07550
57. Liu Y, Jiang D, Duan H, Sun Y, Li G, Tao B, Yun J, Liu Y, Chen B (2021) Dynamic gesture recognition algorithm based on 3d convolutional neural network. Comput Intell Neurosci 2021:4828102
58. Al-Hammadi M, Muhammad G, Abdul W, Alsulaiman M, Bencherif MA, Alrayes TS, Mathkour H, Mekhtiche MA (2020) Deep learning-based approach for sign language gesture recognition with efficient hand gesture representation. IEEE Access 8:192527–192542

59. Qin W, Mei X, Chen Y, Zhang Q, Yao Y, Hu S (2021) Sign language recognition and translation method based on vtn. In: 2021 international conference on digital society and intelligent systems (DSInS). IEEE, pp 111–115
60. Martínez-Gutiérrez ME, Rojano-Cáceres JR, Benítez-Guerrero E, Sánchez-Barrera HE (2019) Data acquisition software for sign language recognition. Res Comput Sci 148(3):205–211
61. Shin J, Matsuoka A, Hasan MAM, Srizon AY (2021) American sign language alphabet recognition by extracting feature from hand pose estimation. Sensors 21(17):5856
62. Xie B, He X, Li Y (2018) RGB-D static gesture recognition based on convolutional neural network. J Eng 2018(16):1515–1520
63. Athitsos V, Neidle C, Sclaroff S, Nash J, Stefan A, Yuan Q, Thangali A (2008) American Sign Language lexicon video dataset (asllvd). CVPR 2008, In: workshop on human communicative behaviour analysis (CVPR4HB)
64. Devineau G, Moutarde F, Xi W, Yang J (2018) Deep learning for hand gesture recognition on skeletal data. In: 2018 13th IEEE international conference on automatic face & gesture recognition (FG 2018), pp 106–113. https://doi.org/10.1109/FG.2018.00025
65. Neidle C, Thangali A, Sclaroff S (2012) Challenges in development of the American Sign Language lexicon video dataset (asllvd) corpus. In: 5th workshop on the representation and processing of sign languages: interactions between Corpus and Lexicon, LREC. Citeseer
66. De Smedt Q, Wannous H, Vandeborre J-P (2016) Skeleton-based dynamic hand gesture recognition. In: proceedings of the IEEE conference on computer vision and pattern recognition workshops, pp 1–9
67. Cover TM (1999) Elements of information theory. Wiley
68. Brownlee J (2019) Probability for machine learning: discover how to harness uncertainty with Python. Machine Learning Mastery
69. Kingma DP, Ba J (2014) Adam: a method for stochastic optimization. arXiv preprint arXiv:1412.6980

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.