IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 7 December 2022, accepted 29 December 2022, date of publication 9 January 2023, date of current version 17 January 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3235368

RESEARCH ARTICLE

Dynamic Hand Gesture Recognition Using Multi-Branch Attention Based Graph and General Deep Learning Model

ABU SALEH MUSA MIAH, MD. AL MEHEDI HASAN, AND JUNGPIL SHIN, (Senior Member, IEEE)
School of Computer Science and Engineering, The University of Aizu, Aizu-Wakamatsu, Fukushima 965-8580, Japan
Corresponding author: Jungpil Shin (jpshin@u-aizu.ac.jp)

This work was supported by the Competitive Research Fund of The University of Aizu, Japan.

ABSTRACT The dynamic hand skeleton data have become increasingly attractive to widely studied for the recognition of hand gestures that contain 3D coordinates of hand joints. Many researchers have been working to develop skeleton-based hand gesture recognition systems using various discriminative spatial-temporal attention features by calculating the dependencies between the joints. However, these methods may face difficulties in achieving high performance and generalizability due to their inefficient features. To overcome these challenges, we proposed a Multi-branch attention-based graph and a general deep-learning model to recognize hand gestures by extracting all possible types of skeleton-based features. We used two graph-based neural network channels in our multi-branch architectures and one general neural network channel. In graph-based neural network channels, one channel first uses the spatial attention module and then the temporal attention module to produce the spatial-temporal features. In contrast, we produced temporal-spatial features in the second channel using the reverse sequence of the first branch. The last branch extracts general deep learning-based features using a general deep neural network module. The final feature vector was constructed by concatenating the spatial-temporal, temporal-spatial, and general features and feeding them into the fully connected layer. We included position embedding and mask operation for both spatial and temporal attention modules to track the node's sequence and reduce the system's computational complexity. Our model achieved 94.12%, 92.00%, and 97.01% accuracy after evaluation with MSRA, DHG, and SHREC'17 benchmark datasets, respectively. The high-performance accuracy and low computational cost proved that the proposed method outperformed the existing state-of-the-art methods.

INDEX TERMS Dynamic hand gesture recognition, machine learning, spatial-temporal attention, hand skeleton points, temporal-spatial attention, deep learning.

I. INTRODUCTION

Research on hand gesture recognition has been increasing daily since many real-life applications like human-computer interaction, nonverbal communication, controlling a wheelchair, abnormal behaviour monitoring, and sign language recognition [1], [2], [3], [4], [5], [6]. Previous work on hand gesture recognition has been divided into two categories based on the data collection procedure: vision-based and sensor-based systems. Since the sensor-based system is difficult because of the carrying sensor, researchers focus on the vision-based system because it only uses a camera, and that is easy to carry. Based on the input data modality, vision-based research work can be divided into two categories: image-based research, which uses full image pixels, and skeleton-based research, which uses only joint information. RGB or RGB-D images are common input for an image-based method for extracting the recognition features. In comparison, skeleton-based methods predict hand gestures based on 2D or 3D coordinates of hand joints. The skeleton sequence is not affected by the limitations of the RGB video and does not consist of color information. Moreover,

The associate editor coordinating the review of this manuscript and approving it for publication was Sudipta Roy.

Johansson et al. have proved that key joints of the gesture carry highly potential information about human motion [7]. Furthermore, each skeleton joint represents a point of the human body in three dimensions coordinates. Among the significant reasons this dataset is valuable to researchers is that it contains higher-level semantic information with a small amount of memory and adapts easily to dynamic systems [8], [9], [10]. Currently, many low-cost depth cameras are available in the market, like Intel RealSense, and Oak-D, which are easy to use for collecting skeleton gesture information and made great progress in gesture recognition research [11], [12]. Based on the skeleton-based dataset, many researchers proposed conventional methods for designing a powerful feature descriptors model for recognizing hand gestures [13], [14], [15]. The main problems of the conventional approach are less performance accuracy and limitations of capabilities for generalization. Researchers applied deep learning techniques to overcome the challenges and improve performance accuracy by directly converting joint coordinates into tensors that feed neural networks [16], [17], [18]. They first produced the feature with the neural network, which is learned by the deep learning network for training. Some other researchers transformed their input skeleton into a meaningful format like a graph, a point sequence or a pseudo image using graph topologies or traversal rules. Then this data format is directly fed into the deep learning method such as CNN, GCN, RNN, or LSTM to extract effective features for improving the network architecture performance [8], [19], [20].

Moreover, until now, there is some uncertainty about whether the hand-crafted extracted features and rules are the optimal choice of joint global dependencies for the model. However, learning global agencies transformer has produced success in the natural language processing (NLP) field, which mainly includes the self-attention mechanism [21], [22], [23]. They reported that better parallelizability and global dependency could be learned with minimum computational complexity among the element. In addition, the attention-based model does not require information about the intrinsic relationship among the joints. Another suitability of the attention model is that there is a limited number of joints in the hand gesture dataset. With minimum computational cost, it is possible to discover useful patterns from the hand skeleton dataset. The main drawback of these models nevertheless considers the spatial and temporal structure of the sequential hand skeleton dataset. Recently, many researchers applied a graph-based spatial-temporal attention model to recognize skeleton-based hand gestures [9], [24], [25], [26], [27], [28]. Although they achieved good performance, the main drawback is the lack of flexibility and sub-optimal performance because of the fixed graph structure, which may be difficult to capture variance and dynamics across different actions. To overcome the challenges, more recently, researchers have worked to develop a dynamic hand-skeleton dynamic graph-based spatial-temporal model to recognize hand gestures [24], [29], [30].

Although they overcome the optimality problem with their model, their performance accuracy is not satisfactory. Moreover, their model may be faced difficulties in achieving satisfactory performance or the same performance all the time because of the inefficiency of the extracted feature. In addition, they only extracted spatial features and then temporal features, and there is no explanation about the vice versa features or if the combination of the other general deep learning features. To overcome the challenges, we are inspired to extract all possible kinds of features from the hand skeleton dataset with the dynamic graph-based attention model, including spatial, temporal attention and general deep learning information. To do this study, we proposed Multi-Branch Attention Based Graph and General Deep Learning model for hand gesture recognition using a skeleton dataset to overcome the mentioned challenges. We developed the architecture by following the dynamic graph-based attention-based mechanism, including spatial, temporal and deep learning information. To convert from original nonsequential skeleton information to sequential information, we used a general neural network, considered the initial feature. Then we employed three branches to extract all possible features with spatial-temporal, temporal-spatial and general deep neural network branches. We considered the spatial-temporal, temporal-spatial branch as a graph-based deep neural network branch that used a position embedding technique to generate the unique markers for every point before each attention block, which helped the attention model feed data sequentially. We utilized a masking operation in each attention block to reduce the computational complexity because two individual pieces of information would decrease the efficiency of the system. The main purpose of the third layer is to recover the missing feature value and solve inefficient signal propagation in the fully connected layer. As we fused the three kinds of features, which combined all possible kinds of features of the hand skeleton, it became an efficient and quicker process compared to the existing system. The significant contribution of this study is as follows:

- We proposed a Multi-Branch Attention Based Graph and General Deep Learning Model to recognize dynamic skeleton-based hand gestures.
- We used several principles in designing spatial-temporal, temporal-spatial and general deep neural network models. The first branch produces spatial-temporal features based on spatial attention through a temporal attention block, and the second produces temporal-spatial features based on temporal attention through a spatial attention block. The third branch carries the general deep learning network features, and finally, we fused three features vector to generate the effective final feature vector.
- Finally, we conducted a comprehensive validation of our system with the three-dynamic skeleton-based hand gesture dataset and achieved superior performance over the state-of-art method considerably within minimum time. The models and code of the proposed model

were uploaded into GitHub to make it public, which is available at https://www.github.com/musaru/Graph-and-General-DNN.

This paper we organized as follows, relevant literature review provided in Section II. Section III is described the benchmark dataset of hand skeletons used to develop this work. The proposed multi-branch spatial-temporal attention model is described in Section IV. Section V described the experimental results and different evaluation scenarios. Section VI concludes the paper, including some future work.

II. RELATED WORK

Hand joint skeleton information-based hand gesture recognition has recently been widely used in the computer vision domain but is still considered a challenging task. The traditional approach, like machine learning and traditional feature extraction method, mainly focuses on developing effective feature descriptors [15], [31], [32], [33], [34]. Ohn-Bar et al. l. proposed a set of feature generators from a skeleton dataset by including a histogram of oriented gradients (HOG) algorithm with the descriptor and employed linear SVM after converting the feature to a 2D array using HOG again [15]. Many other feature extractors have also been proposed by researchers, like the covariance matrix for skeleton joint location [34], joint location, joint angles, 3D geometric relationships between [35], and intraclass variance [36]. Hand geometric configuration for capturing hand shape variation was proposed by Smedt et al., which is used to extract spatial-temporal motion features of hand parts from the whole Euclidean space [37]. They achieved 82.50% and 80.11% accuracy for the 14 and 28 gestures of the DHG dataset after applying SVM machine learning on the Riemannian-based trajectory features. Smedt et al. extracted features based on the fisher vectors and skeleton-based geometric technique, then applied SVM to the concatenated features, and achieved 83.00% and 80.00% accuracy for DHG dataset 14 gesture and 28 gesture sequentially [13]. They extracted three features, namely the shape of connected joints (SoCJ), histogram of hand directions (HoHD), and histogram of wrist rotations (HoWR) and combined them to make the final feature vector. Smedt et al. also applied the fisher vector and shaped the connected feature for the SHREC'17 dataset with the SVM classifier and achieved 88.24% and 81.90% for 14 gestures and 28 gestures sequentially [14]. The advantage of the work is that they demonstrated the superiority of 3D skeleton information over depth-based approaches, but the drawback is they did not consider the amplitude of the gesture and temporal pyramid representation may lose some information. Chen et al. proposed a motion feature extractor by combining articulated movement of the finger and motion feature from global hand movement for extracting bone angle and applying RNN for classification. They evaluated their model with the DHG dataset and achieved 84.68% and 80.32% accuracy for the 14 and 28 classes, respectively [16]. Also, some researchers employed deep neural networks like CNN on the hand joints skeleton data for recognizing hand gestures and significantly improved [14], [16], [18], [23], [27], [32]. Many researchers used other networks with CNN, like an RNN-based approach that transforms the skeleton data into sequential data using traversal rules and feeds into LSTM for training and prediction [9], [17], [18], [38], [39]. Lin C et al. developed a fusion model by combining skeleton LSTM and Res-C3D network for recognizing abnormal hand gestures [39]. Lai et al. incorporated a CNN and an RNN deep learning model for recognizing skeleton-based hand gestures and achieved 85.61% for the DHG 14 gesture dataset [40].

Ma et al. employed an unscented kalman filter (UKF) to reduce the noise and include LSTM for classification [25]. They focused on the noisy dataset by revising the noise in the hand skeleton data and achieved 85.92% and 80.44% accuracy for the 14 and 28 gestures of the DHG dataset sequentially. Nunez et al. proposed a combination of CNN and LSTM models for recognizing a temporal 3D pose, and they achieved 85.46% and 81.10% accuracy for the 14 and 28 gestures of the DHG dataset, respectively [17]. Chen et al. employed an augmented network based on motion (MFA-Net) for recognizing hand gestures, and they achieved 85.75% and 81.10% for the DHG dataset and 91.31% and 86.55% accuracy for the 14 and 28 gestures of the SHREC'17 dataset respectively [26]. They extracted features using a variations auto-encoder from finger and global motion and then fed them into 3 RNNs. Ma et al. proposed a modified memory-augmented neural al network, namely gesture recognition using an enhanced network (GREN) and LSTM architecture, to recognize hand gestures as a short learning algorithm that aims to improve the system's efficiency [23]. They achieved 82.29% and 82.03% for the DHGD dataset and 79.17% for the MSRA dataset. Handwriting-inspired features (Hif3d) are proposed by Boulahia et al. for a 3D skeleton-based gesture classification and achieved 90.48% for 14 gestures and 80.48% accuracy for the 28 gestures of the DHG dataset [28]. Recently, researchers have focused on utilizing self-attention mechanisms to increase the efficiency and performance accuracy of the vision-based hand gesture recognition task by reducing the long-range distance [41], [42]. Vaswani et al. first applied a self-attention network to establish a semantic relationship among words [21].

Query, Key and Values, which multiply the Query with Key in the first stage, divide by the key's dimension and finally apply the SoftMax function to produce the weight vector [22], [30]. After that, it is also employed for detection, semantic segmentation, and relational modelling research work [43], [44], [45]. Currently day, many researchers combined spatial-temporal attention with various architectures like CNN [39], [46], [47], [48], [49], RNN and soft-attention instead of hidden RNN [50], and memory attention networks (MANS) [31]. Song et al. applied a spatial-temporal attention mechanism through RNN and LSTM, where they used individual joints as the main information [51]. Hou et al. employed spatial-temporal attention by combining

with residual connection and temporal convolutional neural network (STA-Res-TCN) to recognize skeleton-based human gestures [27]. They extracted features from the different levels of attention mechanism and applied CNN for individual time steps and finally achieved 89.20% and 85.00% for 14 and 28 gestures of the DHG dataset, respectively. They also evaluated the model SHREC17 dataset and achieved 93.60% and 90.70% accuracy for 14 and 28 gestures sequentially. Recently, a graph convolutional neural network (GCNN) was used by many researchers for gesture recognition [8], [9], [29], [32]. Also, the existing system produced a good performance in some cases but still faced some generalisation problems and sometimes difficulties in achieving high performance for more datasets. To overcome the challenges, we employed here a Multi-Branch Attention Based Graph and General Deep Learning model to recognize hand gestures. We first employed a deep neural network and then employed a spatial-temporal and temporal-spatial branch to produce node and edge features for spatial and temporal domains. To increase the system's generalization, we extracted general deep learning features and concatenated the three extracted features to produce the final feature vector. To reduce the computational cost, we used a spatial-temporal mask and achieved 94.12% accuracy for the MSRA dataset, then 92.00% and 88.78% accuracy for the DHG dataset. In the same way, they achieved 97.01% and 92.78% accuracy for the 14 and 28 gestures sequentially for the SHREC'17 dataset. Our study is more efficient in general, as it does not require hand-crafted transformation rules, and it produced high performance compared to the existing method by a significant margin.

III. DATASET DESCRIPTION

We studied nine open sources of skeleton-based datasets to evaluate the proposed model, namely: MSRA [52], DHG [13], SHREC17 [14], Florence 3-D action [53], UTKinetic [54], UCF-Kinetic [55], NTU [56], NYU [57], ICVL [58], NVGesture [59] which are considered as the benchmark dataset. Among them, Florence 3-D action, UTKinetic, UCF-Kinetic, and NTU datasets are human action datasets. NYU datasets are collected only for binary data, and NVGesture and ICVL contain only RGB and Depth information. Since our objective of the proposed model is to recognize skeleton-based hand gestures, we selected the most recently used skeleton-based hand gesture datasets namely: MSRA, DHG and SHREC17, which have almost similar characteristics in terms of the hand skeleton key points and the number of samples. The details of the uses skeleton dataset are given below [5].

3D Skelton data sequence can be defined as a vector by following Equation (1).

$$S = (P_1, P_2, P_3, \ldots, P_n)^T \tag{1}$$

Here, S represent the skeleton data sequence, P_j represents a multivariate time sequence, T represent the transpose of a matrix and each component of the sequence we

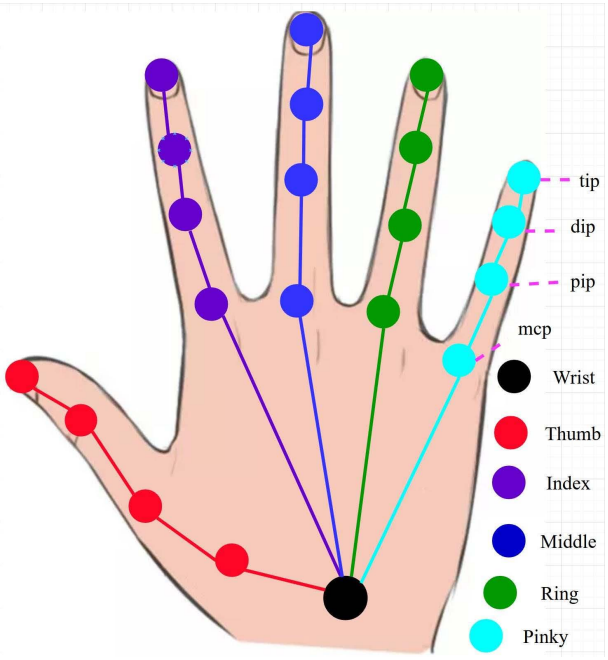


FIGURE 1. Twenty-one joints of MSRA dataset with right-hand skeleton.

can be written as $P_j = \left(P_j(t)\right)_{t \in N}$ which contained three univariate sequence components like the following Equation (2).

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

Here, x, y, and z coordinates are represented by X, Y, and Z for j-th joint, respectively. In addition, $P_j(t)$ represents the position of the j-the skeletal joint. Every joint contains a precise or distinct articulation of the hand of the physical world. From each t time frame, 21 joints for the MSRA dataset and 22 joints for the DHG and SHREC'17 dataset are collected in 3D space by Intel Creative Interactive Camera with their position $P_i = (X_i, Y_i, Z_i) \in \mathbb{R}^3, \forall i \in [1; N]$, where N=21 and 22.

A. MSRA DATASET

One of the hand joint skeleton-based gesture datasets is the MSRA, which is the most challenging publicly available sequence data [52]. This dataset was recorded from 9 participants based on 17 right-hand gestures using Intel Creative Interactive Camera. Each gesture is manually chosen by following the American sign language gesture focusing on the figure articulation's span as much as possible. The dataset contains 490 to 500 frames for each gesture, and for 17 gestures, it is composed of 76500 frames. There are 21 joints as 3D world space or skeleton information in each frame and also collected 2d depth images as well. Among the 21 joints, each finger consists of four joints and one in the palm for the MSRA dataset. The name of the 21 joints is shown in Figure 1. This dataset is considered challenging because of the viewpoint variation.

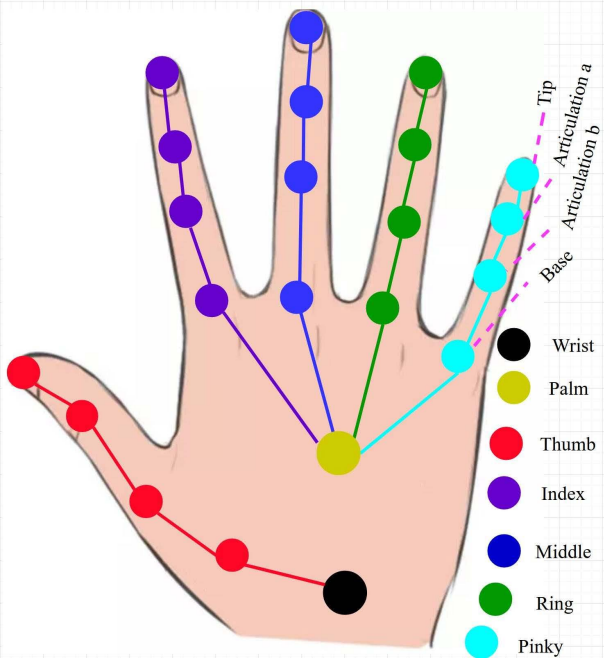


FIGURE 2. Twenty-two joints for the DHG and SHREC'17 dataset from the right-hand skeleton.

B. DHG DATASET

DHG is a publicly available dynamic and one of the challenging datasets for hand gestures, which contains a sequence of 14 right-hand gestures with various finger configurations [13]. For each gesture, the dataset was collected using Intel Real sense SDK and five times from 20 participants in two ways of finger configuration. By following the procedure, they collected a total of 2800 video sequences, and the length of each video contains 20 to 70 frames. Each frame is considered a 3D world space and a full hand skeleton formed with 22 skeleton joints. Figure 2 shows the name and position of the 22-hand skeleton. Some gestures consist of hand movements called coarse gestures, and some other gestures are composed of the shape of a hand, called fine gestures. Among the 14 gestures, nine coarse and five fine-grained gestures are reported. Also, the DHG dataset contains depth image skeleton information, but in our experiment, we used only the skeleton information for gesture recognition. Table 1 shows the name and types of the gestures.

C. SHREC'17 DATASET

Another challenging skeleton-based hand gesture dataset name is the SHREC'17 dataset [14]. This dataset is the same as the DHG dataset, and the Intel Reals Sense SDK was also used and collected from 27 participants. Data were collected 1 to 10 times from each participant in a 2-finger configuration, with a total of 2800 video sequences. Depending on the number of fingers, labels from this dataset are categorized as 14 labels or 28 labels. In addition, among the gestures, some gesture consists of only one finger, and some gesture is composed of a whole hand. For each gesture, a 2D and 3D hand representation was also collected with the depth image for each scene and each time step. Although this dataset contains 2D depth images and 3D hand skeleton information, we used only 3D hand skeleton information in this study. The 22 hands skeleton points name of this dataset is shown in Figure 2 and the name shown in Table 1.

TABLE 1. Name of the 14 gestures for the DHG and SHREC'17 dataset.

No.	Name of the Gesture and Tag name	Gesture Type
G-1	Grab (G)	Fine
G-2	Tap (T)	Coarse
G-3	Expand (E)	Fine
G-4	Pinch (P)	Fine
G-5	Rotation clockwise (RC)	Fine
G-6	Rotation counter-clockwise (RCC)	Fine
G-7	Swipe right (SR)	Coarse
G-8	Swipe left (SL)	Coarse
G-9	Swipe up (SU)	Coarse
G-10	Swipe down (SD)	Coarse
G-11	Swipe x (SX)	Coarse
G-12	Swipe + (S+)	Coarse
G-13	Swipe v (SV)	Coarse
G-14	Shake (Sh)	Coarse

IV. PROPOSED METHODOLOGY

The main goal of the demonstrated MSRA, DHG and SHREC'17 dataset was (1) full hand skeleton and depth information-based dynamic hand gesture recognition, (2) evaluating the efficiency of the hand gesture recognizer based on the number of the finger in the gesture [29]. However, the main objective of our study is not the same as theirs because our study is to achieve high performance in hand gesture recognition with minimum time and cost using only 3D hand skeleton information comparing the still image and video-related hand gesture recognizer. Another objective of our study is to extract all possible features, including a small deep neural network as a skip connection. The purpose of the NN2 is to resolve the missing value problem and improve the performance and efficiency of the model by combining general features with others. Our proposed study is demonstrated in Figure 3(a). We designed a Multi-Branch Attention Based Graph and General Deep Learning Model to recognize dynamic skeleton-based hand gestures. We used two graph-based neural network channels in our multi-branch architectures and one general neural network channel. In graph-based neural network channels, one channel first uses the spatial attention module and then the temporal attention module. On the other hand, the second channel of the graph-based neural network section first used the temporal attention module and then the spatial attention module. The graph-based network branches can be defined with the spatial-temporal and temporal-spatial branches. All three branches took input from the output of an NN1, shown in Figure 3(b). Firstly, NN1 takes the skeleton data points as input for each node and projects the input hand joints 3D coordinate into an initial feature node F_1 that is 128 dimensions. All three branches took F_1 as

input, where the first and second branches embedded the output of NN1 with the corresponding spatial and temporal position to track the sequence correctly. The first branch produces the spatial features with 256 dimensions as a node feature and is projected into a 128-dimension using the neural network NN1, then embedded with temporal position and fed into the temporal attention model, which produces the spatial-temporal feature with 256 dimensions. After that, we projected the 256-dimensional node feature into 128 using NN1 and denoted by F_{ST}. Figure 4(a) shows the spatial-temporal F_{ST} feature extraction mechanism. In the second branch, we follow the reverse sequence of the first branch, where we first fed the initial feature F_1 into the temporal attention model and then fed the temporal feature into the spatial attention model and produced the temporal-spatial feature vector after projecting in the NN1 which is denoted by F_{TS}.

Figure 4(b) shows the temporal-spatial F_{TS} feature extraction mechanism. The 3rd branch also took the F_1 as an input, and after applying the general deep neural network NN2, which is shown in Figure 3(c), it produced a general feature F_G. After that, we concatenated the spatial-temporal, temporal-spatial and general features according to Equation (3) and produced the final feature vector of the proposed architecture F_{Final}. Lastly, we fed the average pool feature vector of the concatenated node features into the fully connected layer for classification.

$$F_{Final} = concate[F_{ST}, F_{TS}, F_R] \tag{3}$$

A. GRAPH-BASED DEEP NEURAL NETWORK BRANCH

We considered two among the three branches as the graph-based deep neural network branch because we used the attention-based mechanism for computing the representation of every joint node of the hand skeleton as a graph node by following its neighbours. The self-attention approach helps us learn an adaptive and dynamic local summary of the neighbour node to improve the prediction, then change to multi-head attention by repeating itself. Extracting a spatial-temporal [29] and temporal-spatial domain feature is the primary purpose of these two branches to build a long sequence for learning the most important part of the hand skeleton. To modify the unified graph, we need to extract spatial and temporal domain features which are dynamically optimized by the different actions. Both graph-based branches took input from the output of NN1 and produced the spatial-temporal and temporal-spatial features after encoding with the spatial and temporal attention model. In both branches, we employed position embedding and masking operations for each attention at spatial and temporal domains to improve performance accuracy and efficiency.

1) SKELETON-BASED GRAPH INITIALIZATION

The structure of the hand skeleton data naturally looks like a graph when we consider it a graph. A hand gesture video sequence containing T frames to represent the hand skeleton and the total N number of 3D hand skeleton joints can be recorded from each of the frames. We assumed a fully connected graph is constructed from the sequence of hand skeleton joints of a frame which is considered as $G = (V, E)$. Let the set of the node denoted by the $V = \{v_{(t,i)} | t = 1, \ldots, T, i = 1, \ldots, N\}$ of the graph and i-th hand joints of the time steps t is contained by the node $v_{(t,i)}$. The feature of the node $v_{(t,i)}$ is contained by the $f_{(t,i)}$ and feature of all nodes are written by $F = \{f_{(t,i)} | t = 1, \ldots, T, i = 1, \ldots, N\}$. The main concept of the feature extraction procedure from the 3D coordinate is that each node connects with other nodes and itself, where we considered three kinds of edges: spatial, temporal, and self-connected edge [27]. We explained the mathematical concept for a set of edges E as follows:

- The connection of two different nodes at the same time step is known as a spatial edge which is defined by $v(t, i) \rightarrow v(t, j)(i \neq j)$.
- The connection of two different nodes at different time steps, known as the temporal edge, is defined by $v(t, i) \rightarrow v(k, j)(t \neq k)$.
- The same node is connected with itself, known as a self-connected edge which is defined by $v(t, i) \rightarrow v(t, i)(t, i)$.

Here, the frame sequence is represented by t and k; the joint skeleton sequence is represented by i and j, respectively.

2) POSITION EMBEDDING

The recurrent network like GRUs and LSTM sequentially process the input, whereas our architecture is one kind of transformer that will not process the skeleton joint sequentially. We used positional embedding here to maintain the sequence of the joint information since there is no built-in notion of the sequence in the transfer. Each skeleton joint of the hand gesture is composed of a tensor for feeding the deep neural networks. For each node, there are no pre-defined structures or orders for showing the node's identity, and it's impossible to identify the corresponding node's hand gesture name. We need to provide unique markers or identifiers for every node to identify the gesture name of the corresponding node. We propose a spatial, temporal position encoding technique to generate the gesture information according to joint information. We use the sine and cosine functions by following [30], [31], [63], and [64] with various frequencies to encode the position number for each node as the encoding functions:

$$P_E(p, 2i) = sin\left(p/1000^{2i/C_{in}}\right)$$
$$P_E(p, 2i+1) = cos\left(p/1000^{2i/C_{in}}\right) \tag{4}$$

Here, $P_E(p, 2i)$ represents the sin function position encoding for the even index, $P_E(p, 2i + 1)$ represent the cos function position encoding for the odd index, the position encoding vector dimension is represented by i, and p denotes the position of each element. According to [63] and [64], the input hand skeleton contains space and time information,

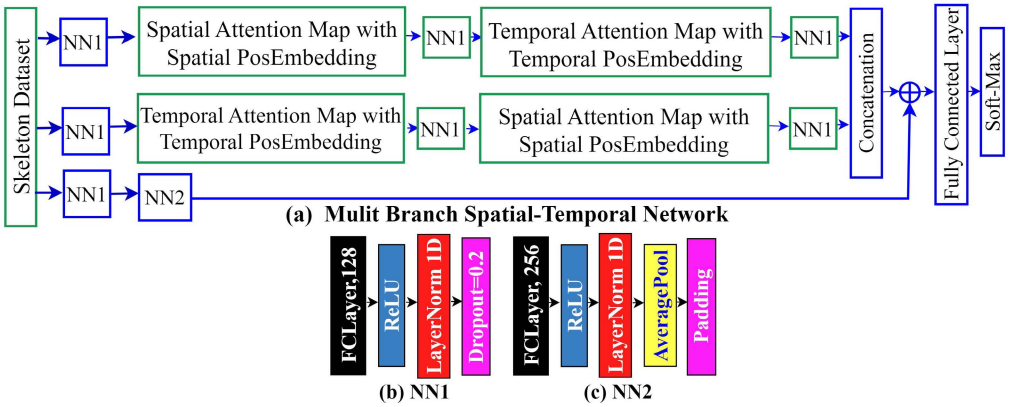


FIGURE 3. Proposed working flow architecture.

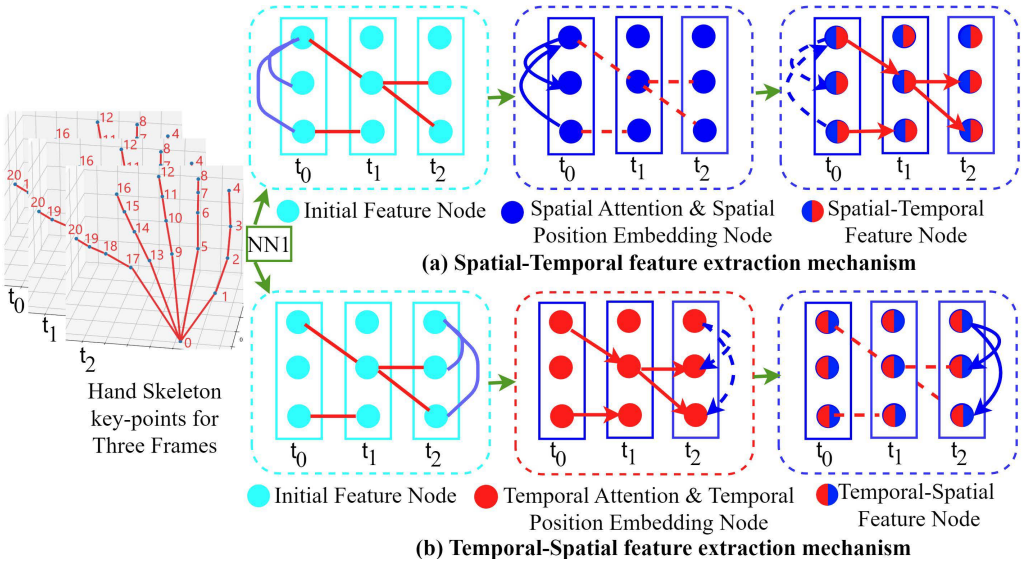


FIGURE 4. Spatial-Temporal and Temporal-Spatial feature extraction working procedure.

and one of the important strategies of the position embedded important strategies is to unify the spatial and temporal information and encode them sequentially. The spatial position embedding comprises the N vectors, where each individual vector consists of a hand joint. We applied spatial position encoding by joining all joints in a single frame by encoding sequentially. On the other hand, temporal position embedding is composed of individual vectors, and each vector represents the corresponding node's hand skeleton graph. We encoded them by encoding the same joints in different frames. Lastly, we added the position information with the output of the NN1 network, which is considered an initial feature of a specific node and fed into the proposed architecture after being embedded with the associated position vector. We added the feature vector with the embedding position, which is shown in the following Equation (5) and (6):

$$\bar{f}_{ST(t,i)} = A_T\left(P^T_{t,i} + A_S\left(f_{(t,i)} + P^S_{(t,i)}\right)\right) \tag{5}$$

$$\bar{f}_{TS(t,i)} = A_S\left(P^S_{t,i} + A_T\left(f_{(t,i)} + P^T_{(t,i)}\right)\right) \tag{6}$$

Here, spatial-temporal and temporal-spatial feature is represented by $\bar{f}_{ST(t,i)}$, and $\bar{f}_{TS(t,i)}$ for a specific node $v_{(t,i)}$, respectively. In the equation, $f_{(t,i)}$ represents the initial feature, A_T represent the output of temporal attention, A_S represent the output of spatial attention. The i-th hand joint of the t frame is represented by $P^S_{(t,i)}$ and $P^T_{(t,i)}$ where the embedding dimension is the same as the input $f_{(t,i)}$ dimension.

3) SPATIAL-TEMPORAL ATTENTION MODULE

The proposed approach consists of spatial-temporal, temporal-spatial attention and general deep neural network branches. Attention-based branches comprise the two-attention model with spatial embedding and two attention models with temporal embedding. In the first branch, the spatial attention block took the input from the output node of NN1 and updated them with the encoding spatial information with the spatial attention block; then, it is fed into temporal attention for updating with the temporal attention block and produced the spatial-temporal feature. In the same way, the second branch produced the temporal-spatial feature by the

reverse procedure of the first branch. In all cases, we applied a multi-head attention mechanism [21], [29], [30], which is visualised in Figure 5. Consider $f_{(t,i)}$ is the initial feature of a node $v_{(t,i)}$ of a hand skeleton, which is used as the input value of an attention layer. There are multi-heads in the attention mechanism, and let m-th attention head first apply the fully-connected layer for mapping query, key and value vectors with the $f_{(t,i)}$ input features. The mapping procedure was performed using the following formulas:

$$Q^m_{(t,i)} = W^m_Q f_{(t,i)}, K^m_{(t,i)} = W^m_K f_{(t,i)}, V^m_{(t,i)} = W^m_V f_{(t,i)} \quad (7)$$

Here query, key and value nodes are represented by $Q^m_{(t,i)}$, $K^m_{(t,i)}$, and $V^m_{(t,i)}$ respectively. The weight metrics of the fully connected layer for the m-th spatial or temporal attention model are denoted by W^m_Q, W^m_K and W^m_V for query, key and value respectively. The spatial, temporal, and self-connected edge weights are calculated in two stages. In the first stage, simultaneously calculates the dot-product between the query and the key vectors [21], [29], [30]. Using a SoftMax activation function normalize the output of the dot product in the second stage. The following formulas in the Equation (8) is execute the above two steps:

$$u^m_{(t,i)\to(t,j)} = \frac{\langle Q^m_{(t,i)}, K^m_{(t,i)} \rangle}{\sqrt{d}}$$
$$\alpha^m_{(t,i)\to(t,j)} = \frac{exp\left(u^m_{(t,i)\to(t,j)}\right)}{\sum_{n=1}^{N} exp\left(u^m_{(t,i)\to(t,n)}\right)} \quad (8)$$

where d represents the dimension of the key vectors and scaled dot products between $v\,(t,i)$ and $v\,(t,j)$ nodes are represented by $u^m_{(t,i)\to(t,j)}$; inner product operation is represented by $<\cdot>$, and attention operation is represented by $\alpha^m_{(t,i)\to(t,j)}$, which is extracted effective information from $v\,(t,i)$ to $v\,(t,j)$ node. In this stage, we can determine whether the attention will be considered spatial or temporal using masking operations by assigning a value of edges. We block the temporal domain information passing by assigning 0 weights for all temporal edges to consider spatial attention and vice versa. Consequently, a weighted skeleton graph is produced by the spatial attention block by considering a hand joint for the same time frame, and the attention head calculates from the node $V_{(t,i)}$ using Equation (9):

$$\bar{f}^m_{(t,i)} = \sum_{j=1}^{N} \left(\alpha^m_{(t,i)\to(t,j)} . V^m_{t,j}\right) \quad (9)$$

Here, $\alpha^m_{(t,i)\to(t,j)}$, and $\bar{f}^m_{(t,i)}$ represents the attention operation and the output of the attention. The attention operation $\alpha^m_{(t,i)\to(t,j)}$ is worked as either spatial or temporal attention for the $V_{(t,i)}$ node based on the masking operation. Moreover, the main idea of spatial attention is to calculate the relationship between two nodes and information passing among the nodes within the same time steps. In addition, according to the learned edge weights, their aggregates and the received information. Equation (9) repeated itself.

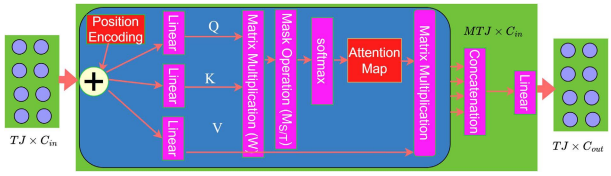


FIGURE 5. Attention map architecture with masking operation.

M times for producing the multi-head attention of spatial or temporal domain are considered multiple feature vectors. Finally, all the attention head outputs concatenate according to Equation (10) and make a single feature vector as $\bar{f}_{(t,i)}$ which is considered the feature vector for the node $V_{(t,i)}$ and we considered spatial attention feature A_S:

$$\bar{f}_{(t,i)} = Concat[\bar{f}^1_{(t,i)}, \bar{f}^2_{(t,i)}, \bar{f}^3_{(t,i)}, \ldots, \bar{f}^M_{t,i}] \quad (10)$$

Here, spatial or temporal attention features for single-head and multi-head are represented by $\bar{f}^i_{(t,i)}$ and $\bar{f}_{(t,i)}$ respectively, and M is the total number of heads in multi-head attention, which is 8 in our study. In the first branch, the spatial attention A_S model learns the weighted skeleton graphs and produces node features by encoding multiple types of structural information. The spatial attention feature is considered as the input feature for the temporal attention A_T and employed the described multi-head attention procedure in the temporal domain and produced the spatial-temporal feature information. In the same way, in the second branch, the temporal attention A_T models learn weighted skeleton graphs and produce node features by encoding multiple types of structural information and then feeding it into the spatial attention A_S and employed the described multi-head attention procedure in the temporal position embedding domain.

4) SPATIAL-TEMPORAL MASK OPERATION

In the proposed architecture, we employed the attention block's spatial and temporal masking operation to cut down the computational cost. In spatial attention, the block mask operator assigns 1 for the spatial position and 0 for others. In the same way, the temporal attention block mask operator contains 1 for temporal value and 0 for other positions. After performing the mask operation, it reduces the data block's size and cuts down the system's computational cost. The concept of attention block is first to calculate three fully connected layers for query, key and values vectors. Then among the query vector and key vector, it calculated the dot product and was divided by the dimension of the key vector. Before the SoftMax activation function, we employed mask operation for both spatial and temporal domains to block unnecessary domains' edges by assigning 0. In Figure 6, we illustrated our masking operation [29], [30]. In the previous section, we discussed the attention where we computed a query matrix Q and K key matrix. Each row of the Q matrix contained the query vector for each node, and each row of the K contained the key for each node. Then we computed the edge weight W matrix using scaled dot

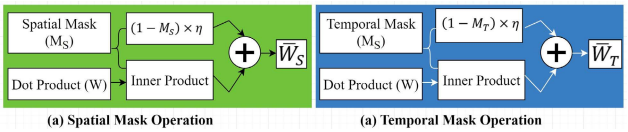


FIGURE 6. Spatial-temporal masking operation.

products by applying the following Equation (11):

$$W = Q \bigotimes K^T \tag{11}$$

Here, W, T, and $\bigotimes$ represent the weight matrix, transpose of the key matrix and matrix multiplication between the query and transpose of the key matrix. The edge weights W can contain a spatial or temporal edge depending on the setting value in each element of the masking matrix. Here, in the first stage, we proposed spatial mask operations to set the value in W that contains the temporal edge to η, where the value of the η is near zero and keeps other values unchanged. After applying the spatial mask, we got the output W_S that contained the spatial edge, and W_T contained the temporal edge. The following Equation (12) calculate the spatial edge and temporal edge:

$$\bar{W}_S = \phi\left(W \odot M_S + (1 - M_S) \times \eta\right) \tag{12}$$

Here, $\bar{W}_S$, $\odot$, ϕ, and $\times$ represent the spatial attention edges, element-wise dot product, Softmax function and multiplication operation sequentially. In addition, W, M_S and η represent weights matrix, spatial mask, and a number close to negative infinity. The mechanism of the mask operation with the weight matrix is if the edges are self-connected or spatial, then it's 1; otherwise, 0. At this work, we assign $-9 \times [10]^5$ for the η. The SoftMax activation normalizes the weights based on the spatial edges because the value of the eta is near zero. Consequently, all temporal edges are set to 0 at W_S. Here, M_S represent the spatial mask containing one if edge represents self-connected or spatial edge otherwise 0. The edge weight calculation formula in the spatial domain of Equation (8) is successfully implemented by Equations (11) and (12). However, masking output W_S a matrix can be applicable for computing the node feature described in Equation (12) based on the matrix multiplication with the value vectors matrix. In the same way, we employed temporal mask operation according to Equation (13) where we used M_T instead M_S for computing the weight matrix W_T in the temporal domain.

$$\bar{W}_T = \phi\left(W \odot M_T + (1 - M_T) \times \eta\right) \tag{13}$$

Here, $\bar{W}_T$, $\odot$, ϕ, and $\times$ represent the temporal attention edges, element-wise dot product, Softmax function and multiplication operation sequentially. In addition, W, M_T, and η represent weights matrix, temporal mask, and a number close to negative infinity. According to the previous discussion, this matrix contains if it is temporal or self-connected edge; otherwise, 0. The main goal of mask operation is to increase the efficiency of the system by reducing computational complexity.

B. GENERAL DEEP NEURAL NETWORK BRANCH

In our study, the general deep neural network branch is used as an alternative path to reach the output of NN2 to concatenate with spatial-temporal and temporal-spatial features. In the NN1, we first employed a fully connected layer along with the relu function, then normalised with layer normalization and dropout layer were used to reduce the overfitting and produced the initial feature F_1. In the NN2 taken output of NN1 as input with three dimensions where a fully connected layer produced 256 dimensions after applying layer normalization, then we employed an average pooling layer to produce an average vector, and finally, a padding layer was used for maintaining the output dimension general feature vector from the NN2. This branch effectively solves the missing data problems and converges problems for exploding gradient and vanishing gradient, which face difficulties in the other branch [62], [63].

V. EXPERIMENTS

We evaluated a comprehensive validation of our system with the three-dynamic skeleton-based hand gesture dataset here. Our proposed system has three channels; two are graph-based neural network channels, and one is a general neural network channel. In graph-based neural network channels, one channel first used the spatial attention module and then the temporal attention module. On the other hand, the second channel of the graph-based neural network section first used the temporal attention module and then the spatial attention module. Finally, we fused them, and after average pooling, we applied a fully connected layer as the final layer for classification.

A. EXPERIMENTAL CONFIGURATION OF TRAINING AND TESTING

We implemented our architecture in the PyTorch platform in the study's NVIDIA 8GB GPU machines. We randomly selected eight frames for each video as the input. First, we subtract every input frame sequence by the first frame palm position based on the previous work; then, we employed some data augmentation techniques by following previous work like shifting, scaling, time interpolation and adding noise. In the compiling section, we used Adam optimizer as an optimizer method with the.001 learning rate for training the model, where batch size was set to 32 and dropout rate was set to 0.1 and 0.2 [64].

B. EXPERIMENTAL SETUP AND IMPLEMENTATION PROTOCOLS

We selected the most recently used three skeleton-based hand gesture famous datasets, MSRA [52], DHG [13], and SHREC17 [14] dataset, to evaluate the proposed model. DHG and SHREC contain 2800 video sequences for 14 and 28 gestures, and 3D coordinates of 22 joints are extracted from each frame. MSRA dataset is collected for 17 gestures and 500/600 frames for each of the gestures in 76500 frames,

TABLE 2. Performance accuracy (%) for MSRA dataset for 17 gesture.

Subject name	Accuracy (%)
Subject-1	100
Subject-2	94.11
Subject-3	94.11
Subject-4	94.11
Subject-5	100
Subject-6	94.17
Subject-7	88.24
Subject-8	100
Subject-9	82.35
Average	94.12

where 21 joints are extracted from each frame. There were 9, 20 and 27 subjects for MSRA, DHG, and SHREC datasets, respectively. We used all three datasets to evaluate our model with a cross-validation procedure: leave-one-out cross-validation (LOOCV). According to the procedure, we sequentially selected n-1 subject information for training for each experiment and the remaining subject for testing. There are nine subjects in the MSRA dataset; keeping one subject dataset for testing, we trained the model with the remaining nine subject datasets. There are 20 subjects in the DHG dataset; we took one subject for testing and the remaining 19 for training. In the same way, among 27 subject datasets for the SHREC'17 dataset, we considered 26 subject datasets for training, and the remaining one was considered as a testing dataset. The overall accuracy of all gestures is reported here.

C. EXPERIMENTAL RESULT

The performance accuracy of the proposed model with three benchmark datasets is demonstrated in this section. SectionV-C1 demonstrated the performance for the MSRA dataset; then Section V-C2 showed the performance for the DHG and the SHREC'17 datasets.

1) EVALUATION WITH MSRA DATASET

In the first stage, we evaluated our proposed system with the MSRA dataset, where eight subject datasets were used for the training and the remaining subjects dataset for the evaluation. Table 2 shows the performance accuracy of the MSRA dataset, where we reported nine individual subject performance accuracy and average accuracy among nine subjects as well. We got maximum accuracy of 100% for subject 1, subject five and subject eight, and minimum accuracy got 82.35% accuracy at subject nine and a 94.12% average accuracy for the nine subjects.

2) EVALUATION WITH DHG DATASET AND SHREC'17 DATASET

Secondly, to evaluate the proposed model with another dataset, namely DHG-14/28, we trained the model using 19 subject datasets and tested it using the remaining subject for each experiment. Accordingly, we repeated it 20 times and used different subjects for both DHG-14 and DHG-28. In the same way, for the SHREC'17-14/28 dataset, we trained using 26 subjects' information and tested it on the remaining ones. We repeated it 27 times accordingly for testing different subjects for both 14 and 28 gestures of the SHREC-17 dataset. Table 3 demonstrated the performance accuracy of the ten subjects for both DHG and SHREC'17 datasets for 14 and 28 gestures. For the DHG dataset with 14 gestures, we got a 97.64% maximum accuracy at subject ten and minimum accuracy of 83.36% at subject 2. In the same way 28 classes of the DHG dataset, we got maximum accuracy of 95.05% at subject ten and minimum accuracy of 75.00% at subject 2. For the SHREC'17 dataset with 14 classes, we got 99.76% accuracy in subject 9, whereas minimum accuracy of 94.00% got in subject 4. For 28 classes of the SHREC'7 dataset, we got a maximum accuracy of 98.56% at subject 10, whereas the minimum accuracy was 87.81% at subject 4. The average accuracy for all 20 and 27 subjects is demonstrated in Table 5 and Table 6 for comparison.

TABLE 3. Performance accuracy for DHG and SHREC'17 dataset for 14 and 28 gestures.

Subject name	DHG Dataset Accuracy (%)		SHRACE'17 Accuracy (%)	
	14 Class	28 Class	14 Class	28 Class
Subject-1	85.00	87.02	99.16	95.00
Subject-2	83.36	75.00	97.33	89.00
Subject-3	96.77	85.71	97.49	92.49
Subject-4	92.63	87.80	94.00	87.81
Subject-5	91.26	91.38	96.05	94.09
Subject-6	90.23	87.57	99.10	91.38
Subject-7	91.10	85.20	98.40	95.74
Subject-8	94.36	91.25	96.66	95.00
Subject-9	93.57	94.15	99.76	95.05
Subject-10	97.64	95.07	99.50	98.56

D. COMPARISON WITH STATE-OF-THE-ART METHOD

We compared our evaluation performance with the state-of-art model for all datasets to prove the superiority of the proposed system. Since we are using graph-based and general neural network modules to extract features and fuse them before feeding them to the classification module, we are getting good accuracy over the existing state-of-the-art model. In the Section V-D1, V-D2, and V-D3 showed the comparison for MSRA, DHG and SHREC17 datasets, respectively.

1) COMPARISON OF MSRA DATASET

Our model produced good performance accuracy for the MSRA dataset by comparing the stat-of-the-art model shown in Table 4. The state-of-the-art model proposed by Ma et al. employed an enhanced neural al network, GREN and LSTM architecture to recognize hand gestures using a skeleton dataset based on the augmented neural network with one short learning memory [23]. The main goal of their idea is to improve performance accuracy, minimize prediction error, and remove unnecessary hyperparameter updating. Their model aims to design a network that can effectively combine and share the feature between dissimilar classes and experiment with their model in different ways. Based on the

TABLE 4. State-of-the-art comparison of the MSRA dataset for 17 gestures.

Method Name	Class	Accuracy [%]
LSTM [23]	17	72.92
Green [23]	17	79.17
Proposed Method	17	94.12

skeleton information, they employed an LSTM network that achieved 72.92% accuracy and achieved 79.17% accuracy with the green network. On the other side, our proposed study achieved 94.12% accuracy, which is more than 10.00% of the existing method.

2) COMPARISON OF DHG DATASET

In Table 5, the proposed study is compared with the various state-of-the-art method for the DHG dataset for both 14 and 28 gestures. It demonstrated that the proposed study outperforms most state-of-the-art techniques and achieves comparable performance accuracy with DG-STA [29] and STA-GCN [8]. Although some existing methods used depth and skeleton both information, such as joint angles and HOG2 (JAHOG) [15] approaches, ASJT [37], SoCJ + HoHD + HoWR [13], NIUKF-LSTM [25], CNN+RNN [39], our study only relies on the only skeleton. Our method generated an average accuracy of 20 subjects at 92.00% for the 14-gesture, which is higher than the advanced algorithm. In the case of 28 gestures, it achieved 88.78% average accuracy for 20 subjects, which is also higher than the existing performance accuracy. JASHOG [15] applied joint similarity with [31] and achieved 83.35% and 76.53% accuracy for 14 and 28 gestures sequentially. In [16], they employed motion features augmented with RNN (MARNN), achieving 84.68% and 80.32% for 14 and 28 gestures sequentially. Smedt et al. also show satisfactory performance accuracy, but they showed a problem for incorrect joint locations during closed hands [13]. They combined geometric features with the multi-level representation of the fisher vector that the temporal pyramid ensures to achieve the feature for the SVM classifier. LSTM-based technique, although achieved better performance compared to the hand-crafted features such as CNN + LSTM [17], NIUKF-LSTM [25], Green [23], and MFA-Net [26]. Ma et al. employed LSTM to handle noisy skeleton data by integrating a spatial type of Kalman filter namely: nested interval unscented Kalman filter (NIUKF), then achieved 8.92% and 80.44% accuracy for the 14 and 28 gestures of the DHG dataset. sequentially [25]. Nunez et al. produced accuracy by combining CNN and LSTM, where they focused on spatial-temporal feature extraction from the skeleton features and achieved higher performance than hand-crafted features [17]. For recognizing hand-gesture using the skeleton hand joint motion feature augmented network (MFA-Net) model is proposed by Chen et al. and achieved 85.75% and 81.10% for the DHG dataset for 14 and 28 gestures sequentially [26]. Another technique employed by Ma et al. based on the GREN and LSTM architecture to recognize hand gestures

TABLE 5. State-of-the-art comparison of the DHG dataset.

Methods	Accuracy(%) (14 Gestures)	Accuracy(%) (28 Gestures)
JAHOG [15]	83.85	76.53
GREN [23]	82.29	82.03
ASJT [37]	82.50	80.11
SoCJ + HoHD + HoWR [13]	83.07	80.00
MARNN [16]	84.68	80.32
CNN+RNN [39]	85.46	74.19
NIUKF-LSTM [25]	84.92	80.44
CNN+LSTM [17]	85.46	74.19
MFA-Net [26]	85.75	81.04
Res-TCN [27]	86.90	83.60
STA-Res-TCN [27]	89.20	85.00
Boulahia [28]	90.48	80.48
STA-GCN [8]	91.20	87.10
DG-STA [29]	91.00	88.00
Proposed Method	92.00	88.78

achieved 82.29% and 82.03% for the DHGD skeleton dataset [23]. Res-TCN, STA-Res-TCN [27], STA-GCN [8] and DG-STA [29] are applied attention-based architecture for recognizing hand gestures based on skeleton information. How et al. employed a spatial-temporal attention-based neural network: STA-Res-TCN, for extracting features from different levels of attention block for each time step and achieved 89.20% and 85.00% for 14 and 28 gestures sequentially for the DHG skeleton hand gesture dataset [27]. Boulahia et al. extracted Hif3d for gesture classification and achieved 90.48% for 14 gestures and 80.48% for the 28 gestures of the DHG dataset [28]. Chen et al. employed the DG-STA approach to improve accuracy and reduce the computational cost for hand gesture recognition and achieved 91.00% and 88.00% accuracy for the DHG dataset [29]. Unlike existing work, our proposed architecture focuses on multiple branches for producing multiple feature vectors generated by the parallel architecture, which also preserves the dynamic hand gesture properties. Moreover, replacing some branches of the proposed architecture can easily be compatible with the existing state-of-the-art system like DG-STA [29]. Moreover, our study's main focus is to fully explore prior and future work composition. The table's contents have demonstrated that our proposed method's performance is higher than the existing method in this factor.

3) COMPARISON OF SHREC'17 DATASET

The comparison Table 6 demonstrated that our model outperforms most of the state-of-the-art methods for the SHREC'17 dataset for both 14 and 28 gesture cases and comparable performance with DG-STA [29] and STA-GCN [8]. As shown in Table 6, our study achieved 97.01% for 14 and 92.78% accuracy for the 28 gestures, which is average for 27 subjects and outperformed all existing methods for both experiment settings. Specifically, our method improved the accuracy of 14 gestures by 3.40% and 2.78% for the 28 gestures once we compared them with the existing best-performance DG-STA [29] methods and more than 5.40% with more recent work by STA-GCN [8]. Although some existing methods used depth and skeleton, both information among

TABLE 6. State-of-the-art comparison of the SHREC'17 dataset.

Method	Accuracy (%) (14 Gestures)	Accuracy (%) (28 Gestures)
HON4D [31]	78.53	74.03
SMTRM [32]	79.61	62.00
SoCJ + HoHD + HoWR [14]	88.24	81.90
Res-C3D [40]	89.52	-
MFA-Net [26]	91.31	86.55
Res-TCN [27]	91.10	87.30
STA-Res-TCN [27]	93.60	90.70
STA-GCN [8]	92.27	87.70
DG-STA [29]	94.40	90.00
Proposed Method	97.01	92.78

them, a histogram-based method based on depth sequence (HON4D) [31], shape analysis of motion trajectories on Riemannian manifold (SMTRM) [32] for hand gesture classification, SoCJ + HoHD + HoWR [14], while our study only relies on an only skeleton. In the case of the SHREC17 dataset, MFA-Net produced 91.31% and 86.55% accuracy for 14 and 28 gestures sequentially [23], [27] . Res-TCN, STA-Res-TCN [27], STA-GCN [8], and DG-STA [29] are applied attention-based architecture for recognizing hand gestures based on skeleton information. Among the attention-based model, STA-Res-TCN achieved 93.60% and 90.70% accuracy for 14 and 28 gestures [27] whereas DG-STA [29] approach to improve accuracy and reduce the computational cost of hand gesture recognition and achieved 94.40% and 90.00% accuracy for sequentially the 14 and 28 gestures. Our proposed method mainly focuses on parallelly producing multiple features from multiple branches of the parallel architecture, which preserves the properties of dynamic hand gestures. In addition, the proposed study can be compatible with the existing attention-based method discarding some branches and modules [27], [28], [29]. Moreover, our study's primary focus is to fully explore prior and future work composition. The table's contents have demonstrated that our proposed method's performance is higher than the existing method in this factor.

VI. CONCLUSION

We employed an attention-based Multi-Branch Attention Based Graph and General Deep Learning approach for recognizing hand gestures based on the study's 3D hand skeleton data points. Our method provided a multi-branch graph-based deep neural network and general deep neural network model with masking operation for learning spatial and temporal domain information and produced a potential feature vector for classification. We employed two branches of graph-based neural networks where the first branch took input from the output of the neural network NN1, and after encoding with spatial and temporal attention, it produced the spatial-temporal. In the same way, the second branch produced a temporal-spatial feature by following the reverse sequence of the first branch, which is concatenated with the output of the general deep neural network branch and applied to the average pooling layer. Finally, a fully connected layer is applied to learn node and edge weight for classification. Since we are using graph-based and general neural network modules to extract features and fuse them before feeding them to the classification module, our proposed model is getting good accuracy over the existing state-of-the-art model for all three datasets. In the table, we demonstrated the experimental result for three datasets and the effectiveness of our proposed architecture. In the future, we plan to collect 3D hand skeleton information from ourselves from more gestures to develop a sign language-based communication system.

ABBREVIATIONS

NN1	Deep Neural Network-1.
NN2	Deep Neural Network-2.
MSRA	Microsoft Research Asia.
DHG	Dynamic Hand Gesture.
SHREC	A name of the Data Collection Contest.
RGB-D	Red, Green Blue with Depth.
CNN	Convolutional Neural Network.
SVM	Support Vector Machine.
HON4D	Histogram-based Method Based on Depth Sequence.
Res-C3D	3D Convolutional Neural Networks with Residual Architecture.
MFA-Net	Motion Feature Augmented Network.
DHGD	Dynamic Hand Gesture Depth.
SHREC2017	3D Shape Retrieval Contest 2017.
MANS	Memory Attention Networks.
GCNN	Graph Convolutional Neural Network.
DG-STA	Dynamic Graph-based Spatial Temporal Attention.
ASJT	Analysing Set-of-Joints Trajectories.
NIUKF-LSTM	Nested Interval Unscented Kalman Filter LSTM.
GCN	Graph Convolutional Network.
RNN	Recurrent Neural Network.
NLP	Natural Language Processing.
OAK-D	OpenCV AI Kit with Depth.
RNNG	Recurrent Neural Network Grammar.
LSTM	Long Short-Term Memory.
HOG	Histogram of Oriented Gradients.
JAHOG	Joint angle HOG.
STA- GCN	Spatial Temporal Attention with Graph Convolutional Network.
SoCJ	Shape of Connected Joints.
HoHD	Histogram of Hand Directions.
HoWR	Histogram of Wrist Rotations.
UKF	Unscented Kalman Filter.
GREEN	Gesture Recognition using an Enhanced Network.
Hif3d	Handwriting-inspired Features.
STA-Res-TCN	Spatial-Temporal Attention by combining with Residual Connection and Temporal Convolutional Neural Network.
MARNN	Motion Features Augmented with RNN.

SMTRM	Shape Analysis of Motion Trajectories on Riemannian Manifold.
Res-TCN	Residual Connection with Temporal Convolutional Neural Network

REFERENCES

[1] A. S. M. Miah, J. Shin, M. A. M. Hasan, and M. A. Rahim, "BenSignNet: Bengali sign language alphabet recognition using concatenated segmentation and convolutional neural network," *Appl. Sci.*, vol. 12, no. 8, p. 3933, Apr. 2022.

[2] A. S. M. Miah, J. Shin, M. Al M. Hasan, M. A. Rahim, and Y. Okuyama, "Rotation, translation and scale invariant sign word recognition using deep learning," *Comput. Syst. Sci. Eng.*, vol. 44, no. 3, pp. 2521–2536, 2023.

[3] M. A. Rahim, A. S. M. Miah, A. Sayeed, and J. Shin, "Hand gesture recognition based on optimal segmentation in human-computer interaction," in *Proc. 23rd IEEE Int. Conf. Knowl. Innov. Invention (ICKII)*, Aug. 2020, pp. 163–166.

[4] M. A. Khan, M. Mittal, L. M. Goyal, and S. Roy, "A deep survey on supervised learning based human detection and activity classification methods," *Multimedia Tools Appl.*, vol. 80, no. 18, pp. 27867–27923, Jul. 2021.

[5] G. Devineau, F. Moutarde, W. Xi, and J. Yang, "Deep learning for hand gesture recognition on skeletal data," in *Proc. 13th IEEE Int. Conf. Autom. Face Gesture Recognit. (FG)*, May 2018, pp. 106–113.

[6] S. S. Rautaray and A. Agrawal, "Vision based hand gesture recognition for human computer interaction: A survey," *Artif. Intell. Rev.*, vol. 43, no. 1, pp. 1–54, Jan. 2015.

[7] G. Johansson, "Visual perception of biological motion and a model for its analysis," *Perception Psychophysi.*, vol. 14, no. 2, pp. 201–211, 1973.

[8] S. Yan, Y. Xiong, and D. Lin, "Spatial temporal graph convolutional networks for skeleton-based action recognition," in *Proc. 32nd AAAI Conf. Artif. Intell.*, 2018, pp. 7444–7452.

[9] C. Si, Y. Jing, W. Wang, L. Wang, and T. Tan, "Skeleton-based action recognition with spatial reasoning and temporal stack learning," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 103–118.

[10] L. Shi, Y. Zhang, J. Cheng, and H. Lu, "Two-stream adaptive graph convolutional networks for skeleton-based action recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 12026–12035.

[11] M. Oberweger, P. Wohlhart, and V. Lepetit, "Training a feedback loop for hand pose estimation," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 3316–3324.

[12] M. Oberweger and V. Lepetit, "DeepPrior++: Improving fast and accurate 3D hand pose estimation," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops (ICCVW)*, Oct. 2017, pp. 585–594.

[13] Q. De Smedt, H. Wannous, and J.-P. Vandeborre, "Skeleton-based dynamic hand gesture recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2016, pp. 1–9.

[14] Q. D. Smedt, H. Wannous, J.-P. Vandeborre, J. Guerry, B. L. Saux, and D. Filliat, "Shrec'17 track: 3d hand gesture recognition using a depth and skeletal dataset," in *Proc. 3DOR-10th Eurographics Workshop 3D Object Retr.*, 2017, pp. 1–6.

[15] E. Ohn-Bar and M. M. Trivedi, "Joint angles similarities and HOG2 for action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, Jun. 2013, pp. 465–470.

[16] X. Chen, H. Guo, G. Wang, and L. Zhang, "Motion feature augmented recurrent neural network for skeleton-based dynamic hand gesture recognition," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2017, pp. 2881–2885.

[17] J. C. Núñez, R. Cabido, J. J. Pantrigo, A. S. Montemayor, and J. F. Vélez, "Convolutional neural networks and long short-term memory for skeleton-based human activity and hand gesture recognition," *Pattern Recognit.*, vol. 76, pp. 80–94, Apr. 2018.

[18] P. Zhang, C. Lan, J. Xing, W. Zeng, J. Xue, and N. Zheng, "View adaptive recurrent neural networks for high performance human action recognition from skeleton data," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2117–2126.

[19] Z. Qiu, T. Yao, and T. Mei, "Learning spatio–temporal representation with pseudo-3D residual networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 5533–5541.

[20] Y. Min, Y. Zhang, X. Chai, and X. Chen, "An efficient PointLSTM for point clouds based gesture recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 5761–5770.

[21] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–11.

[22] Z. Dai, Z. Yang, Y. Yang, J. Carbonell, Q. V. Le, and R. Salakhutdinov, "Transformer-XL: Attentive language models beyond a fixed-length context," 2019, *arXiv:1901.02860*.

[23] C. Ma, S. Zhang, A. Wang, Y. Qi, and G. Chen, "Skeleton-based dynamic hand gesture recognition using an enhanced network with one-shot learning," *Appl. Sci.*, vol. 10, no. 11, p. 3680, May 2020.

[24] A. Bigalke and M. P. Heinrich, "Fusing posture and position representations for point cloud-based hand gesture recognition," in *Proc. Int. Conf. 3D Vis. (3DV)*, Dec. 2021, pp. 617–626.

[25] C. Ma, A. Wang, G. Chen, and C. Xu, "Hand joints-based gesture recognition for noisy dataset using nested interval unscented Kalman filter with LSTM network," *Vis. Comput.*, vol. 34, nos. 6–8, pp. 1053–1063, Jun. 2018.

[26] X. Chen, G. Wang, H. Guo, C. Zhang, H. Wang, and L. Zhang, "MFA-Net: Motion feature augmented network for dynamic hand gesture recognition from skeletal data," *Sensors*, vol. 19, no. 2, p. 239, Jan. 2019.

[27] J. Hou, G. Wang, X. Chen, J.-H. Xue, R. Zhu, and H. Yang, "Spatial–temporal attention Res-TCN for skeleton-based dynamic hand gesture recognition," in *Proc. Eur. Conf. Comput. Vis. (ECCV) workshops*, Sep. 2018, pp. 1–15.

[28] S. Y. Boulahia, E. Anquetil, F. Multon, and R. Kulpa, "Dynamic hand gesture recognition based on 3D pattern assembled trajectories," in *Proc. 7th Int. Conf. Image Process. Theory, Tools Appl. (IPTA)*, Nov. 2017, pp. 1–6.

[29] Y. Chen, L. Zhao, X. Peng, J. Yuan, and D. N. Metaxas, "Construct dynamic graphs for hand gesture recognition via spatial–temporal attention," 2019, *arXiv:1907.08871*.

[30] L. Shi, Y. Zhang, J. Cheng, and H. Lu, "Decoupled spatial–temporal attention network for skeleton-based action-gesture recognition," in *Proc. Asian Conf. Comput. Vis.*, Nov. 2020, pp. 1–16.

[31] C. Li, C. Xie, B. Zhang, J. Han, X. Zhen, and J. Chen, "Memory attention networks for skeleton-based action recognition," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 9, pp. 4800–4814, Sep. 2022.

[32] K. Thakkar and P. J Narayanan, "Part-based graph convolutional network for action recognition," 2018, *arXiv:1809.04983*.

[33] S. Lu, D. Metaxas, D. Samaras, and J. Oliensis, "Using multiple cues for hand tracking and model refinement," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2003, p. 443.

[34] M. E. Hussein, M. Torki, M. A. Gowayyed, and M. El-Saban, "Human action recognition using a temporal hierarchy of covariance descriptors on 3D joint locations," in *Proc. 23rd Int. Joint Conf. Artif. Intell.*, 2013, pp. 1–7.

[35] R. Vemulapalli, F. Arrate, and R. Chellappa, "Human action recognition by representing 3D skeletons as points in a lie group," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 588–595.

[36] J. Wang, Z. Liu, Y. Wu, and J. Yuan, "Mining actionlet ensemble for action recognition with depth cameras," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 1290–1297.

[37] Q. D. Smedt, H. Wannous, and J.-P. Vandeborre, "3D hand gesture recognition by analysing set-of-joints trajectories," in *2nd Int. Workshop UHA3DS*, Springer, 2016, pp. 86–97.

[38] S. Li, W. Li, C. Cook, C. Zhu, and Y. Gao, "Independently recurrent neural network (IndRNN): Building a longer and deeper RNN," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5457–5466.

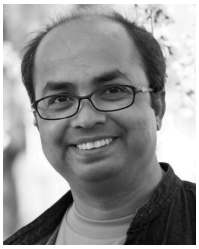
[39] C. Si, W. Chen, W. Wang, L. Wang, and T. Tan, "An attention enhanced graph convolutional LSTM network for skeleton-based action recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1227–1236.

[40] K. Lai and S. N. Yanushkevich, "CNN+RNN depth and skeleton based dynamic hand gesture recognition," in *Proc. 24th Int. Conf. Pattern Recognit. (ICPR)*, Aug. 2018, pp. 3451–3456.

[41] Y. Tian, X. Peng, L. Zhao, S. Zhang, and D. N. Metaxas, "CR-GAN: Learning complete representations for multi-view generation," 2018, *arXiv:1806.11191*.

[42] J. Liu, A. Shahroudy, D. Xu, and G. Wang, "Spatio–temporal LSTM with trust gates for 3D human action recognition," in *Proc. Eur. Conf. Comput. Vis.* Amsterdam, The Netherlands: Springer, 2016, pp. 816–833.
[43] A. Santoro, D. Raposo, D. G. Barrett, M. Malinowski, R. Pascanu, P. Battaglia, and T. Lillicrap, "A simple neural network module for relational reasoning," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–10.
[44] H. Hu, J. Gu, Z. Zhang, J. Dai, and Y. Wei, "Relation networks for object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3588–3597.
[45] J. Fu, J. Liu, H. Tian, Y. Li, Y. Bao, Z. Fang, and H. Lu, "Dual attention network for scene segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3146–3154.
[46] Z. Yang, Y. Li, J. Yang, and J. Luo, "Action recognition with spatio–temporal visual attention on skeleton image sequences," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 8, pp. 2405–2415, Aug. 2018.
[47] C. Li, Q. Zhong, D. Xie, and S. Pu, "Skeleton-based action recognition with convolutional neural networks," in *Proc. IEEE Int. Conf. Multimedia Expo Workshops (ICMEW)*, Jul. 2017, pp. 597–600.
[48] Q. Ke, M. Bennamoun, S. An, F. Sohel, and F. Boussaid, "A new representation of skeleton sequences for 3D action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3288–3297.
[49] C. Li, Q. Zhong, D. Xie, and S. Pu, "Co-occurrence feature learning from skeleton data for action recognition and detection with hierarchical aggregation," 2018, *arXiv:1804.06055*.
[50] F. Baradel, C. Wolf, and J. Mille, "Human action recognition: Pose-based attention draws focus to hands," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops (ICCVW)*, Oct. 2017, pp. 604–613.
[51] S. Song, C. Lan, J. Xing, W. Zeng, and J. Liu, "An end-to-end spatio-temporal attention model for human action recognition from skeleton data," in *Proc. AAAI Conf. Artif. Intell.*, vol. 31, 2017, pp. 1–8.
[52] X. Sun, Y. Wei, S. Liang, X. Tang, and J. Sun, "Cascaded hand pose regression," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 824–832.
[53] L. Seidenari, V. Varano, S. Berretti, A. Del Bimbo, and P. Pala, "Recognizing actions from depth cameras as weakly aligned multi-part bag-of-poses," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, Jun. 2013, pp. 479–485.
[54] L. Xia, C.-C. Chen, and J. K. Aggarwal, "View invariant human action recognition using histograms of 3D joints," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Workshops*, Jun. 2012, pp. 20–27.
[55] C. Ellis, S. Z. Masood, M. F. Tappen, J. J. LaViola, and R. Sukthankar, "Exploring the trade-off between accuracy and observational latency in action recognition," *Int. J. Comput. Vis.*, vol. 101, no. 3, pp. 420–436, 2013.
[56] A. Shahroudy, J. Liu, T.-T. Ng, and G. Wang, "NTU RGB+D: A large scale dataset for 3D human activity analysis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1010–1019.
[57] J. Tompson, M. Stein, Y. Lecun, and K. Perlin, "Real-time continuous pose recovery of human hands using convolutional networks," *ACM Trans. Graph.*, vol. 33, no. 5, pp. 1–10, Sep. 2014.
[58] D. Tang, H. J. Chang, A. Tejani, and T.-K. Kim, "Latent regression forest: Structured estimation of 3D articulated hand posture," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 3786–3793.
[59] P. Molchanov, X. Yang, S. Gupta, K. Kim, S. Tyree, and J. Kautz, "Online detection and classification of dynamic hand gestures with recurrent 3D convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 4207–4215.
[60] X. Chen, G. Wang, H. Guo, and C. Zhang, "Pose guided structured region ensemble network for cascaded hand pose estimation," *Neurocomputing*, vol. 395, pp. 138–149, Jun. 2020.
[61] A. S. M. Miah, M. A. M. Hasan, J. Shin, Y. Okuyama, and Y. Tomioka, "Multistage spatial attention-based neural network for hand gesture recognition," *Computers*, vol. 12, no. 1, p. 13, 2023.
[62] H. Mahmud, M. M. Morshed, and M. Kamrul Hasan, "A deep learning-based multimodal depth-aware dynamic hand gesture recognition system," 2021, *arXiv:2107.02543*.
[63] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," 2017, *arXiv:1710.10903*.
[64] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.

ABU SALEH MUSA MIAH received the B.Sc. and M.Sc. degrees in computer science and engineering from the Department of Computer Science and Engineering, University of Rajshahi, Rajshahi, Bangladesh, in 2014 and 2015, respectively. He is currently pursuing the Ph.D. degree with the School of Computer Science and Engineering, The University of Aizu, Japan, under a scholarship from the Japanese Government (MEXT). He became a Lecturer and an Assistant Professor at the Department of Computer Science and Engineering, Bangladesh Army University of Science and Technology (BAUST), Saidpur, Bangladesh, in 2018 and 2021, respectively. He has authored and coauthored more than 20 publications published in widely cited journals and conferences. His research interests include CS, ML, DL, HCI, BCI, and neurological disorder detection.



MD. AL MEHEDI HASAN received the B.Sc., M.Sc., and Ph.D. degrees in computer science and engineering from the Department of Computer Science and Engineering, University of Rajshahi, Rajshahi, Bangladesh, in 2005, 2007, and 2017, respectively. He became a Lecturer, an Assistant Professor, an Associate Professor, and a Professor at the Department of Computer Science and Engineering, Rajshahi University of Engineering and Technology (RUET), Rajshahi, in 2007, 2010, 2018, and 2019, respectively. He has coauthored more than 100 publications published in widely cited journals and conferences. His research interests include bioinformatics, artificial intelligence, pattern recognition, medical image, signal processing, machine learning, computer vision, data mining, big data analysis, probabilistic and statistical inference, operating systems, computer networks, and security.

JUNGPIL SHIN (Senior Member, IEEE) received the B.Sc. degree in computer science and statistics and the M.Sc. degree in computer science from Pusan National University, South Korea, in 1990 and 1994, respectively, and the Ph.D. degree in computer science and communication engineering from Kyushu University, Japan, in 1999, under a scholarship from the Japanese Government (MEXT). He was an Associate Professor, a Senior Associate Professor, and a Full Professor at the School of Computer Science and Engineering, The University of Aizu, Japan, in 1999, 2004, and 2019, respectively. He has coauthored more than 300 published papers for widely cited journals and conferences. His research interests include pattern recognition, image processing, computer vision, machine learning, human–computer interaction, non-touch interfaces, human gesture recognition, automatic control, Parkinson's disease diagnosis, ADHD diagnosis, user authentication, machine intelligence, as well as handwriting analysis, recognition, and synthesis. He is a member of ACM, IEICE, IPSJ, KISS, and KIPS. He has served as the program chair and a program committee member for numerous international conferences. He serves as an Editor for IEEE journals and Sensors (MDPI) and a reviewer for several major IEEE and SCI journals.