Multimed Tools Appl (2018) 77:32063–32091
https://doi.org/10.1007/s11042-018-6199-7

Indian sign language recognition using graph matching on 3D motion captured signs

D. Anil Kumar[1] · A. S. C. S. Sastry[1] · P. V. V. Kishore[1] · E. Kiran Kumar[1]

Received: 5 November 2017 / Revised: 29 March 2018 / Accepted: 23 May 2018/
Published online: 14 June 2018

Abstract A machine cannot easily understand and interpret three-dimensional (3D) data. In this study, we propose the use of graph matching (GM) to enable 3D motion capture for Indian sign language recognition. The sign classification and recognition problem for interpreting 3D motion signs is considered an adaptive GM (AGM) problem. However, the current models for solving an AGM problem have two major drawbacks. First, spatial matching can be performed on a fixed set of frames with a fixed number of nodes. Second, temporal matching divides the entire 3D dataset into a fixed number of pyramids. The proposed approach solves these problems by employing interframe GM for performing spatial matching and employing multiple intraframe GM for performing temporal matching. To test the proposed model, a 3D sign language dataset is created that involves 200 continuous sentences in the sign language through a motion capture setup with eight cameras.The method is also validated on 3D motion capture benchmark action dataset HDM05 and CMU. We demonstrated that our approach increases the accuracy of recognizing signs in continuous sentences.

Electronic supplementary material The online version of this article (https://doi.org/10.1007/s11042-018-6199-7) contains supplementary material, which is available to authorized users.

✉ D. Anil Kumar
danilmurali@kluniversity.in

A. S. C. S. Sastry
ascssastry@kluniversity.in

P. V. V. Kishore
pvvkishore@kluniversity.in

E. Kiran Kumar
kiraneepuri@kluniversity.in

[1] Biomechanics and Vision Computing Research Center, Department of Electronics and Communications Engineering, K.L.E.F(Deemed-to-be-University), Green Fields, Vaddeswaram, Guntur (DT), AP, India

Keywords 3D sign language · 3D motion capture · Spatial graph matching · Temporal graph matching · Distance measures

1 Introduction

Motion capture has penetrated fields that involve human actions such as movies, animation, medicine, and sports to capture complex human motions in a three-dimensional (3D) space. The motion capture setup is a complex technology that produces 3D spatial points and time-varying 3D skeleton models for analysis. 3D trajectories and velocities of the 3D skeleton points provide an approximately accurate model of human actions in real time. However, many processing levels are required in the physical and virtual pre- and post-processing methods to generate usable data.

The sign language involves a complex and nonlinear motion of hands and fingers. Moreover, at times, the head and torso are referred to for explaining a particular sign. Machine translation of the sign language is researched extensively in two aspects—instrumentation and signal processing, and computer vision. For the first aspect, the instrument used is a pair of sensor gloves; the finger movements produce one-dimensional (1D) vectors for classification by using the sensor gloves. The second aspect involves a two-dimensional (2D) camera sensor. We believe that 2D time-varying approaches produce superior processing performance compared with 1D approaches. Sign language is known as a visual language model because it involves hand shapes and movements with involving the head, face, and torso. However, the sensor-glove-based approach cannot recognize the signs that involve hand movements with respect to the head, face, and torso.

The sign recognition problems of using 2D video approaches are still being researched extensively. In our previous study that involved the use of these approaches, we noted problems pertaining to the camera sensor resolution, blur camera focus, variation in lighting conditions, changes in the background, use of non-contrasting colors, occlusions, and movement emptiness. A 3D-based approach can eliminate most of these drawbacks that influence machine interpretation for sign language recognition (SLR). In this study, we demonstrated the use of a 3D motion capture technology to recognize gestures of the Indian sign language.

Graphs are a powerful tool for representing structured 3D data. However, graph construction from 3D data is complex for tasks such as human motion retrieval. The complexity pertains to the spatial characterization of the joint pairs in motion to form similarity metrics on the vertex and edge pairs in two consecutive motion sequences. This study focuses on recognizing signs from a continuous dataset containing a sequence of signs exploiting the above characterizations.

The continuous 3D dataset is recorded using a Vicon motion capture setup with eight cameras and a video camera involving the RGB color model. The 3D dataset contains 200 signs that form meaningful sentences known as continuous sign frames (CSFs). Each frame in the sentence dataset is represented with a graph of 57 spatial points known as vertices and 56 joint pairs known as edges. The intergraph matching algorithm (IGM) is applied on two consecutive frames, and a threshold is set to extract high-motion sequences from sign videos. The same process is applied for query sign frames (QSFs) for enabling motion frame separation. QSFs are the testing input 3D sign video sequence.

An adaptive graph matching (AGM) algorithm is proposed for obtaining the QSFs in the CSFs. IGM calculates the similarity between corresponding vertices and edges for consecutive QSFs and CSFs. However, for sign language 3D data, the IGM model produces negative matching because of the small variations between signs in most of the cases. This problem is solved by using an AGM model. In this model, each vertex and edge in a QSF and CSF are matched both spatially and temporally.

First, temporal matching is conducted for finding the signs in the CSF dataset. AGM is applied on the first three frames and last three frames of the QSF and the entire CSF dataset. This provides match locations in the CSF dataset. These locations contain a start frame and end frame to be matched in the CSF dataset. We can obtain multiple start and end frames on the basis of the number of times the QSF might appear in the CSF dataset. These multiple CSF's are grouped to form a group CSF (GCSF) with which the QSF is matched by using a spatial AGM.

Each spatial graph of the QSF is matched on the basis of both vertex and edge with every graph's vertex and edge in the GCSF. This makes our proposed spatial AGM independent of the number of frames in the QSF and GCSF.

2 Literature review

Last few years saw a surge in research findings related to Human action recognition. The early days of human action recognition is burdened by computing power, database unavailability and complex algorithms for extracting humans from 2D videos [36]. Few approaches used spatial and temporal tracking information as a feature for classification along with spatial shape features [14]. However, 3D model based approaches have reported good recognition rates [21]. The models used were hand build in 3D animation and inducing real action information to the models is a difficult task. Moreover, it is not possible to define all action class models beforehand and develop a regression model to match real time 2D object movements to 3D models [44].

Hence, sensor based approaches [32] are gaining ground with the availability of low cost sensors such as accelerometers. In [34], mobile based accelerometer data is used to detect human actions such as walking, running, jumping, jogging etc. Further, in [33], human activities such as walking up, down, climbing etc are recognized using a dataset of 30 subjects. The activities are performed by wearing a smart phone on the human body. Various data modelling methods such as probabilistic interval based models [31] and data fusion models [11] are popular and are considered state–of–the–art.

SLR has transformed with technology from 1D and 2D to 3D models in the last two decades. The 1D SLR is based on 1D signals acquired from hand gloves [26] and classified using signal processing methods [24, 25]. Recently, researchers started using the leap motion sensor [35] to extract 1D signals of finger movements and estimate the related gestures of the sign language using the hidden Markov models (HMMs).

The faster 1D models produce superior recognition rates when the emphasis is only on the signs involving the hand movement. However, the sign language involves signs involving the head, torso, and facial expressions with the hand movements and shapes [39]. 2D video data of signs produces relatively more information compared with 1D data gloves. By using 2D SLR methods, one can obtain all the elements of a visual language with a constraint on speed and classification accuracy. Moreover, for 2D SLR, HMMs are the most widely researched classifiers for continuous and discrete versions of the sign language [6]. Further studies on 2D SLR models, and the corresponding research challenges can be found in [28, 41, 43]. The other challenge encountered by researchers is the conversion of the detected signs into meaningful sentences [41]. Figure 1 displays the challenging problems in 2D SLR, such as hand tracking, occlusions on hands and face, background lighting, varying signer backgrounds, and camera sensor dynamics for processing.

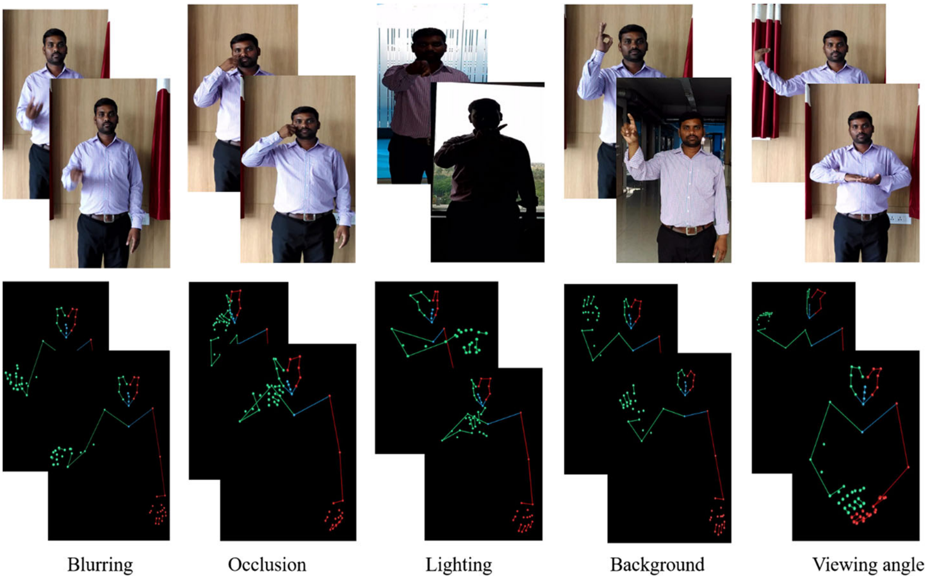


Fig. 1 Challenges in 2D SLR processing and elimination of the problems by using in 3D SLR methods

The 3D SLR model solves all the problems observed during 2D SLR. However, using the 3D SLR model induces new challenges such as large data sets, 2D–3D integration, sign and non-sign differentiation, 3D shape analysis, and 3D point classification. Due to the availability of 3D depth sensors in the previous decade such as Microsoft Kinect and similar RGB-D sensors, SLR has evolved to a new level.

Kinect sensors capture 3D depth images that are sometimes combined with RGB color video data to form RGB-D video images. Recently, the 3D SLR [1, 3, 30, 42] is explored to a considerable extent by using these sensors. Moreover, 3D data from Kinect sensors consists of hand trajectories [10], orientations, and velocities [13] of a single depth image. Features such as 3D body joint locations [37] and Finger-Earth Mover's Distance (FEMD) [49] are used for sign classification.

The features from 3D gestures are classified using HMMs [24], deep convolutional neural networks [12], weighted dynamic time warping [19], and Euclidian distance measures [4]. The discriminative exemplar coding by using 3D Kinect data classifies sign videos on the basis of exemplars learned from the discrimination at the frame level and individual video level [42]. The experiment employed a set of 2000 American sign language videos with features such as color, depth, and skeleton information. However, this model cannot appropriately select the exemplar sign frames from the background clutter.

Literature suggests the extensive use of Kinect for SLR with optimal accuracies for detecting signs. However, Kinect sensors still has problems with occlusions, cannot accurately perform multi-object sensing, and is signer dependent. The 3D motion capture technology (3D Mocap) [40] eliminates most of the capture related problems such as background motion, multiple movements, lighting changes, and occlusions [16].

3D motion capture data analytics is a currently emerging research field, and most researches use the analytics as a validation tool rather than an analysis tool. In the last few years, research on analytics is picking pace, and 3D data analytics is the most challenging problem [47]. Motion features such as trajectories, velocities, and angles between markers are used for the classification of human motion [8]. The analysis uses a limited 3D dataset for classifying less than 10 motions from a large set of features generated. A large dataset of 3D human motions for use in applications such as sports, dance, and gait can be found in a previous study [48]. For 3D SLR, the shapes, trajectories, and angles change abruptly with less scope for periodicity in the motions.

Utilizing geometric relationships between 3D motion data to identify motions in a 3D video sequence [2, 17, 38] is by far the most successful analytical model technique. However, these models use static data for finding the geometric relationship in marker joint spatial data. 3D motion capture produces spatiotemporal joint data [29] for analysis and requires a model to define this relationship among joints in the sequence of frames.

Human motion recognition method specified in [29] is recognized by representing 3D human joint data by using undirected graphs $g\,(v, e)$, where v represents the vertex and e is the edge that represents the path between two consecutive vertices. This model was used in the present study on SLR by using 3D motion capture. Computer vision researchers discovered the efficient use of graphs in representing image objects for shape matching and motion segmentation on both 2D and 3D data sequences [18]. In [7], the human motion in each frame is represented by a graph, and the matching similarity is calculated between training and testing data. In [45], a hyper-graph matching (GM) algorithm recognizes human motions from by using the spatiotemporal features extracted from graphs. Graph-based techniques are researched using adaptive graph kernels (AGK) in [29]; these techniques include the Kuhn–Munkres GM algorithm [46] and dynamic programing [9] for 3D human motion matching. Graph kernels have received extensive appreciation from researchers for 3D continuous data [15].

The 3D motion retrieval problems are accurately addressed using AGM, as discussed above. However, most of the researches address problems in the temporal domain by using temporal pyramid structures. The problems pertaining to the use of temporal pyramids for SLR are related to the video length of a sign. Each sign video in the QSF has a different length from the same sign in the CSF dataset. Moreover, the use of fixed length windows for temporal pyramid construction provides negative results for 3D SLR. Hence, we designed a sign location identification algorithm and multiple frame matching between QSFs and the CSF datasets for improving sign extraction. Compared with the temporal pyramid model, the sign recall accuracy has improved when the aforementioned methods are used. Figure 2 displays the flowchart of the proposed 3D SLR process.

3 3D GM for SLR

3.1 Preparing 3D sign graphs

A graph $g\,(v, e)$ is a set of connected points in 3D space $v \to \mathrm{R}\,(x, y, z)$. Moreover, $e \to \mathrm{R} = d\left(x_i, y_j, z_k\right)$ represents the pair-wise distances of vertices, where $i, j, k \to I$ identify joint vertex pairs. The 3D motion capture environment offers a natural inclination toward graph theoretical analysis. A graph with 3D data is denoted with a two-tuple $g = \{v, e\}$. The features for vertices and edges are quantified by $v = [v_1, v_2, ..., v_n] \in \mathrm{R}^{d_v \times n}$and $e = [e_1, e_2, ..., e_n] \in \mathrm{R}^{d_e \times (n-1)}$, respectively. Here, d_v and d_e are dimensionalities of vertices and edges in a graph g, respectively, and n denotes the number of vertices in a graph.

For the sign language representation, we designed a 3D model by using a set of 57 marker points, as shown in Fig. 3. The 57 markers cover 98% of the movements involved

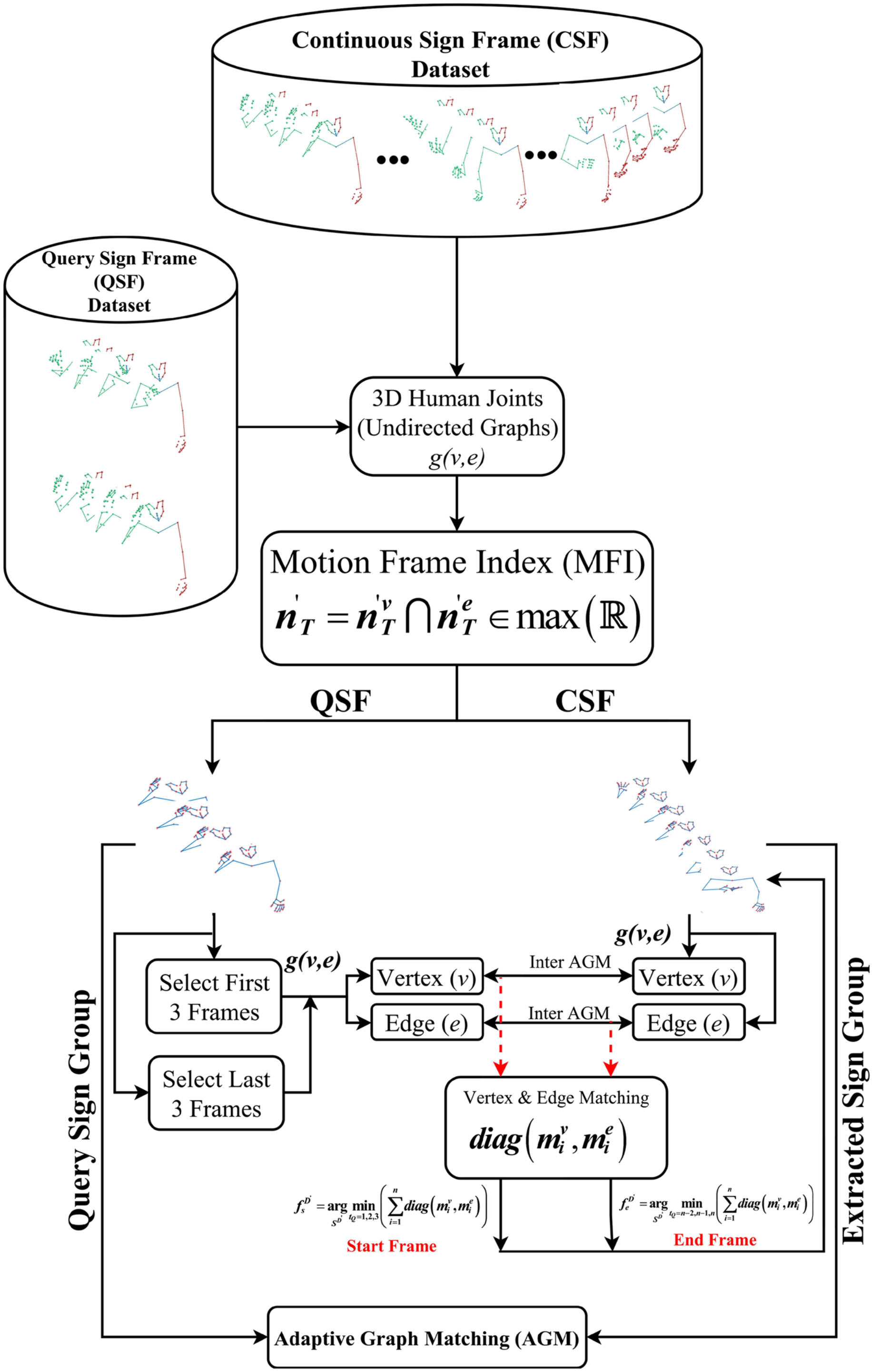


Fig. 2 Flowchart for the proposed 3D SLR process

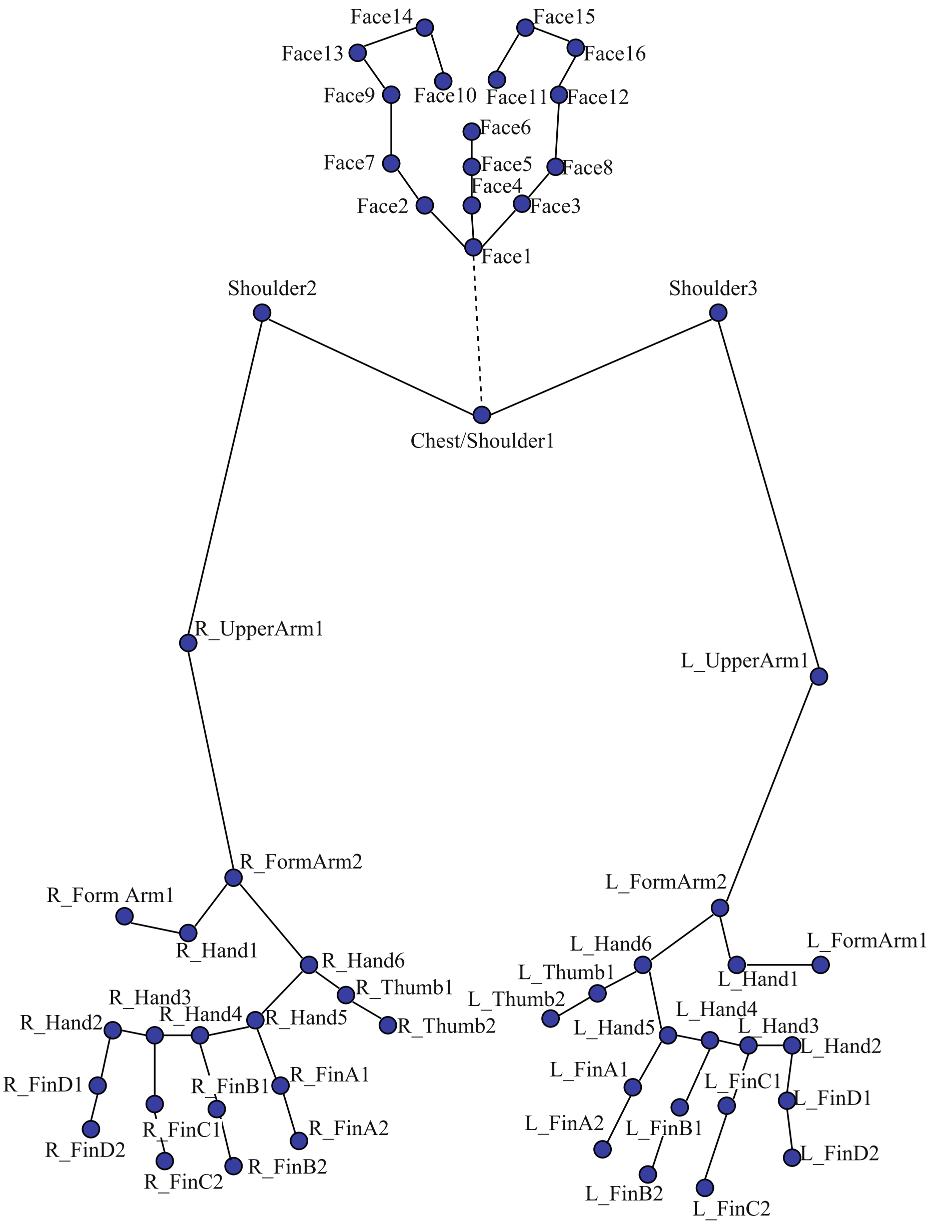


Fig. 3 Signer's representation in 3D motion capture

in sign language. Each marker is labeled and is represented as a vertex or node of a graph. Edges are defined as distance between two adjacent marker points. Thus, graphs for the 3D sign language are represented using 57 vertices and 56 edges. The current study focused on representing 3D points as a symmetrical graph, where edge features are symmetrical with respect to the horizontal axis of the graph. This notation for graphs representing 3D data can be used for both undirected and directed graphs.

Signs in a sign language represent non-static, nonlinear movements of human hands and fingers and sometimes involve the head, face, and torso. 3D trajectories that are represented by $T^t = (x_i, y_i, z_i)_t \rightarrow v^i_t$ form the features of the vertex in t^{th} frame and i^{th} vertex. An edge in the t^{th} frame is a distance feature that is computed between two adjacent i^{th} and j^{th} vertex in a graph.

$$e^{ij}_t = \underset{i,j}{\forall} \left\| v_i - v_j \right\|^2_2 \tag{1}$$

Each frame in the 3D sign video is represented with the same features. Signs are a combination of time-varying spatial relations between the features in a single frame graph. Hence, the 3D graphs for representing signs are a set of adaptive graphs, as shown in Fig. 3. with labeled vertices.

3.2 3D sign AGM

A 3D adaptive graph $g\,(v, e, t)$, where t is the frame rate, for a sign, is an entity that adapts non-linearly in accordance with the movement of the 57 marker points on the signer. A sign is represented as a set of t frames of spatial deformations on an adaptive graph g. Given a pair of adaptive sign graphs S^D and S^Q from the CSF datasets and QSFs of the input query video, respectively, the AGM is defined as a similarity measure between each vertex and edge pair. For the sign graph from the CSF dataset, $S^D = \left\{V^D_d, E^D_d, G^D, H^D\right\}$, and query sign graph from the QSFs, $S^Q = \left\{V^Q_d, E^Q_d, G^Q, H^Q\right\}$, we compute two matching matrices $M^V \in \mathrm{R}^{n_1 \times n_2}$ and $M^E \in \mathrm{R}^{n_1 \times n_2}$ for matching the vertex and edge pairs of the two sign graphs. Here, G^D, H^D, G^Q, H^Q contain the encoded topology information of the two vertex–edge pairs in a graph, and $G^D, H^D, G^Q, H^Q \in \{0, 1\}^{n \times m}$. For the k^{th} edge, e^k starts at the i^{th} vertex and ends at the j^{th} vertex, and thus, $G_{ik} = H_{jk} = 1$. These additional parameters are applicable to the graph if the matching is being initiated for an asymmetric graph. For SLR, we consider symmetric graphs, where $G_{ik} = H_{jk} = 1$. The parameter d gives the dimensionality of the vertex V and edge E features.

The similarity measures between i^{th}_1 and i^{th}_2 nodes of S^D and S^Q is decided by computing the equations $m^v_{i_1,i_2} = \underset{min}{\Delta}\left\{v^D_{i_1}, v^Q_{i_2}\right\}$ during vertex matching and $m^e_{i_1,i_2} = \underset{min}{\Delta}\left\{e^D_{i_1}, e^Q_{i_2}\right\}$ during edge matching. The sign matching between the dataset and input query is defined as the vertex and edge matching score and is formulated as

$$m_S = \begin{cases} m^v_{i,j} & if\;\; i_{v_n} = j_{v_n}, n - \text{no. of vertices} \\ m^e_{i,j} & if\;\; i_{v_n} \neq j_{v_n}, \;\; \text{and} \qquad G_{ik}, H_{jk} = 1 \\ 0 & Otherwise \end{cases} \tag{2}$$

where the diagonal elements provide the vertex and edge matches and the non-diagonal elements provide the non-matches in the matrix m_S. However, few implementation problems are associated with this model for 3D SLR. The first is the number of frames in the 3D sign dataset. The 3D sign dataset is a continuous set of signs that constitute meaningful sentences that are used in daily life. These sentences are concatenated to form a complete CSF dataset having 121 sentences of varying lengths in a sequential format. The total frame count of the of 3D dataset after combining the signs together is 48572. Due to the concatenation, the dataset developed redundancy in the form of non-sign frames, that is, non-motion or less-motion frames. In the next section, we discuss the process of removing the motionless redundant frames in the dataset.

3.3 IntraGM for motion frames extraction

Figure 4 displays a frame view of the sign “Good Morning” in the CSF dataset. The intraGM performed on the entire 3D dataset causes the motion segmentation problem in the 3D spatial domain. This section describes the deformable spatial GM (DSGM) method to extract motion vertices from less-changing or static vertices. This retains the motion intensive sign frames only.

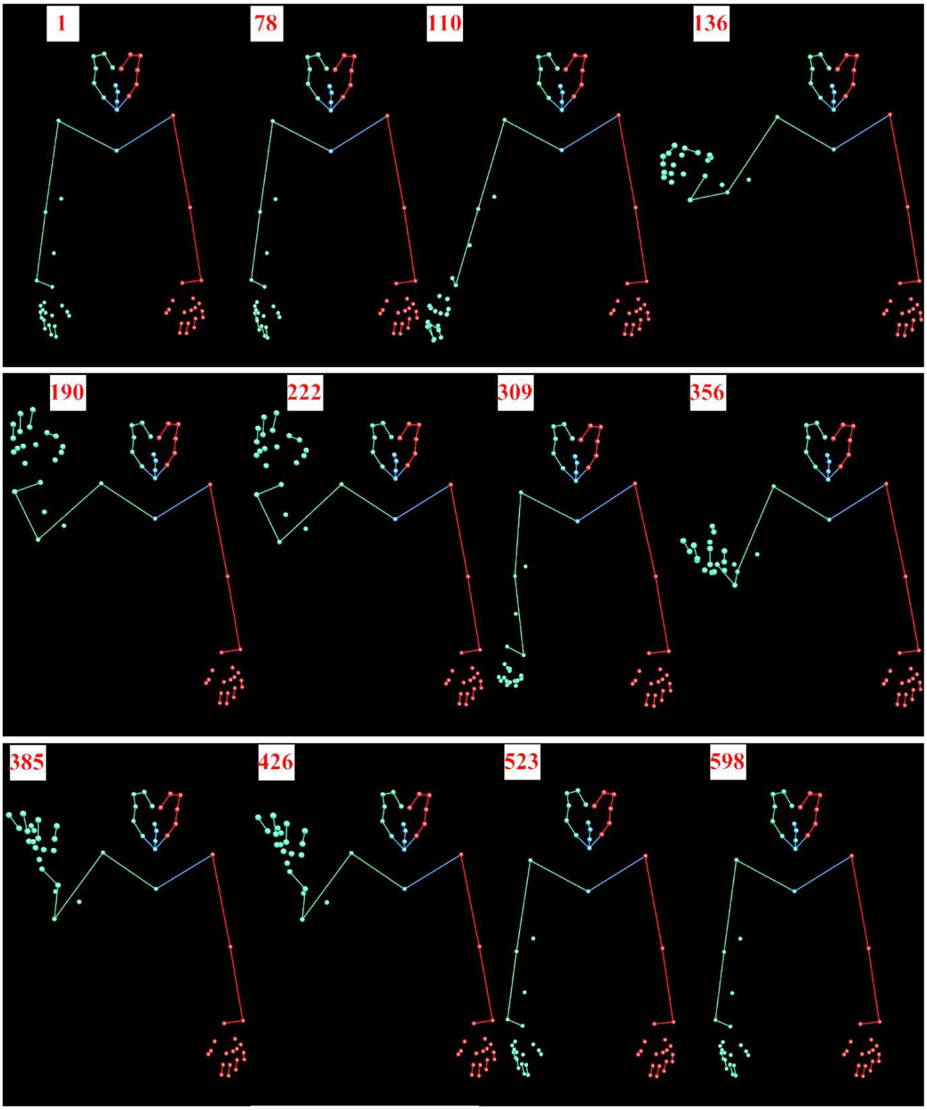


Fig. 4 “Good Morning” sign recorded in 600 frames with the intermediate rigid and changing frames with frame numbers

In 3D motion capture, $\{x, y, z\} \in \mathrm{R}$, and the trajectories of the markers on a signer's body are obtained. Subsequently, vertex features of each graph denoted as $S_t^D = \{V, E, G, H\}$ are assigned as the vertex trajectories, $V = [v_1, v_2 ....., v_n] \in \mathrm{R}^{d\times n_t}$, where d is the feature dimension and n_t is the number of features per frame t. Similarly, the l_2 norm calculation of position vectors for 3D markers presented in (1) represent the edge features $E = \left[e_1, e_2, ..., e_{n-1}\right] \in \mathrm{R}^{d\times n_t}$. Here, $G_{ik} = H_{jk} = 1$ because the graph is fully symmetrical.

For any two consecutive frames in the dataset represented by graphs S^{D_t} and $S^{D_{t-1}}$ and a geometrical trancajectories by $T(\bullet)$, we computed the vertex matching score $M^V(T) \in \mathrm{R}^{n\times n}$ and edge matching score $M^E(T) \in \mathrm{R}^{(n-1)\times(n-1)}$ with the Euclidean distance function defined as

$$m^v_{i_1,i_1'}(T) = \left\| v^t_{i_1} - v^{t-1}_{i_1'} \right\|_2^2 \tag{3}$$

$$m^e_{(i_1,j_1),(i_1',j_1')}(T) = \left\| v^t_{i_1} - v^{t-1}_{i_1'} \right\|_2^2 - \left\| T(v^t_{j_1}) - T(v^{t-1}_{j_1'}) \right\|_2^2 \tag{4}$$

where $\left\{i_1, i_1'\right\} \in \mathrm{R}$ and i_1 and i_2 are the vertices in corresponding frames t and t – 1, respectively. The goal of using DSGM for solving the sign dataset motion segmentation problem is to retain frames that have high-motion content without re-ordering the sign frames. For example, in Fig. 3, we have n_{T} (= 602) frames that represent the "Good Morning" sign. The role of DSGM is to optimize the number of frames from n_{T} to $n_{\mathrm{T}'}$ by removing the rigid frames and retaining the high-motion content frames in the same order for appropriately representing the sign. The motion frame index (MFI) n_T' is defined as follows by using the vertex features:

$$n'^v_T = \mathop{\arg}_{i_1,i_1' \in R^{n\times n}} \quad m^v_{i_1,i_1'} \geq m^v_{tr} \tag{5}$$

where the threshold of the vertices for all the common vertex points is defined as

$$m^v_{tr} = \arg \max_{\substack{i_1,\, i_1' \in \mathrm{R} \forall i_1 = i_1' \\ t \in (1-T)}} \left(m^v\right) \tag{6}$$

Similarly, the MFI between two consecutive frames with edge features is defined as

$$n'^e_T = \mathop{\arg}_{i_1,j_1,i_1',j_1' \in \mathrm{R}^{(n-1)\times(n-1)}} \quad m^e_{i_1,j_1,i_1',j_1'} \geq m^e_{tr} \tag{7}$$

with an edge threshold of the following:

$$m^e_{tr} = \arg \max_{\substack{i_1,\, j_1,\, i_1',\, j_1' \in \mathrm{R} \forall i_1 = i_1',\, j_1 = j_1' \\ t \in (1-T)}} \left(m^e\right) \tag{8}$$

The sign frame is a motion frame; if both the vertex and edge frame indices are high, then a maximum real-value number is obtained. Otherwise, the frame is discarded.

$$n'_T = n^{'v}_T \bigcap n^{'e}_T \in \max(\mathrm{R}) \tag{9}$$

The variable "arg" in (5) and (7) represents the extracted frame from a sequence of frames in the dataset. The motion segmented sign 3D frames in the dataset are represented as S^D. Moreover, S^D forms the dataset for testing the input sign. The input 3D video QSFs also undergo the same process and are represented by a graph through S^Q. AGM between S^Q and S^D will provides 3D sign recognition.

3.4 InterAGM for sign motion extraction

A sign in the sign language contains hand shapes and motions that involve the head, face, or torso. The objective of SLR in this study is to match the motion components from the 3D features obtained using the motion capture setup with a 3D sign database. The matching is between the trajectories of the moving body parts represented as graphs in the 3D sign dataset S^D and query sign input S^Q. For effective matching between S^Q and S^D, a two-step matching is proposed in this study.

In the first step, to avoid matching between all frames in the input query and the CSF dataset, we match the first three and last three frames of S^Q with the entire CSF dataset. This provides a set of multiple instances that have the same starting and ending frames in the CSF dataset as that of the input QSFs that are matched. This process is not sufficient for the sign matching because the number of three frame matchings between CSF and QSF can be more than one.

However, in the second step, we match each vertex and each edge in the query graph $S^{Q'}$ with each vertex and edge in the instantaneous sign dataset $S^{D'}$ graph. All vertex and edge matching will model similarity in hand motion and shape with respect to head, face and torso recognizing sign in the dataset perfectly. Figure 5. displays the GM process used in this study.

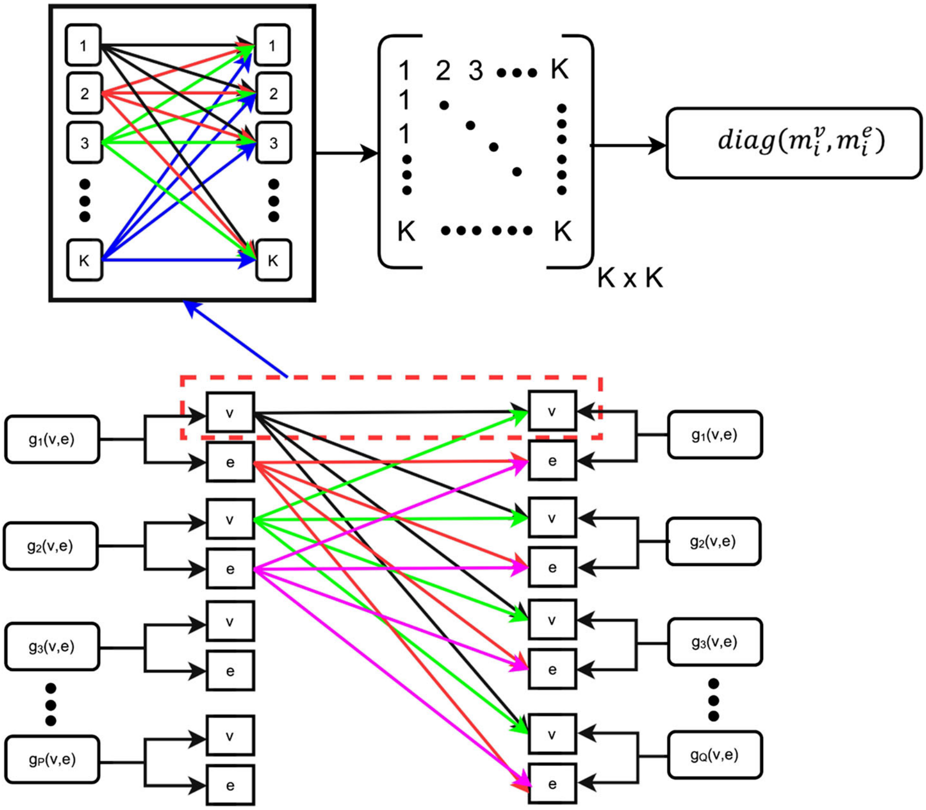


Fig. 5 AGM visualization for sign similarity matching

3.4.1 Segmenting signs in dataset $S^{D'} \to S^{D*}_{\gamma}$

The objective of this process is to identify and extract the first and last three frames in the query dataset $S^{Q'} = \left\{V_d^{Q'}, E_d^{Q'}, G^{Q'}, H^{Q'}\right\}$ by referring to the indices bookmarked in the start and end frames of the CSF dataset $S^{D'} = \left\{V_d^{D'}, E_d^{D'}, G^{D'}, H^{D'}\right\}$. The first three and last three frames in the query sign are represented as $S^{Q\;\;i}\;\;\forall\;\; i = 1, 2, 3, t_Q - 2, t_Q - 1, t_Q$, where t_Q gives the number of frames in the query sign video. Matching similarity between vertex components of the graphs $S^{Q\;\;i}$ and $S^{D'}$ is computed as

$$m^{v}_{i_1,\{i_1,i_2,....,i_n\}}(T) = \left\| v^{t^i_Q}_{i_l} - v^{t_D}_{\{i_1,i_2,....,i_n\}} \right\|_2^2 \in \mathrm{R}^{n\times n} \;\;\forall\;\; l = 1, 2, ...n \tag{10}$$

The matching variable is computed for each frame t and for n vertices in both the graphs. Each vertex in a graph is matched to all n vertices in another graph; thus, a vertex similarity matrix $m^{v}_{i_1,\{i_1,i_2,....,i_n\}}(T)$ of size $n \times n$ is obtained. In some cases, the number of vertices is different due to the loss of markers in motion. This occurs in less than 5% of the cases during capturing the signs. To solve this problem, we use the edge matching similarity when vertices in graphs are different after computation as follows:

$$m^{e}_{(i_1,j_1),(i'_l,j'_l)}(T) = \left\| v^{t^i_Q}_{i_1} - v^{t_D}_{i'_1} \right\|_2^2 - \left\| T(v^{t^i_Q}_{j_1}) - T(v^{t_D}_{j'_1}) \right\|_2^2 \in \mathrm{R}^{(n-1)\times(n-1)} \;\;\forall\;\; l = n - 1 \tag{11}$$

Similar calculations are performed for the last three frames of the input query sign. The start frame and end frame indices are computed using the expression and by extracting the maximum vertex–edge match index.

$$f_s^{D'} = \underset{S^{D'}}{\arg} \underset{t_Q=1,2,3}{\min} \left(\sum_{i=1}^{n} diag\left(m_i^v, m_i^e\right) \right) \tag{12}$$

and the new dataset start frame is

$$S_1^{D'} = S^{D'}\left(f_s^{D'}\right) \tag{13}$$

The end frame in the input video QSF dataset is

$$f_e^{D'} = \underset{S^{D'}}{\arg} \underset{t_Q=n-2,n-1,n}{\min} \left(\sum_{i=1}^{n} diag\left(m_i^v, m_i^e\right) \right) \tag{14}$$

and the end bookmark in the dataset is

$$S_2^{D'} = S^{D'}\left(f_e^{D'}\right) \tag{15}$$

However, there are multiple sentences in the CSF dataset $S^{D'}$ that contain same words that have the same sign, such as "Good Morning, how are you today?; My wife looks good; My life now is in good shape." Now the process detects each of these "good" signs in the dataset and generates multiple start and end frames in the dataset. These are stored as GCFSs and are given by

$$S^{D*}_{\gamma} = \underset{\gamma}{\arg} \bigcup_{f_s^{D'}}^{f_e^{D'}} \left\{S_1^{D'}, S_2^{D'}\right\} \tag{16}$$

where γ is the number of GCFSs pertaining to the start and end frames of the input QSF dataset. The group may or may not contain the sign in the query. By matching the first few frames in the QSF dataset, we cannot decide whether the same sign being clustered in the sign group. Hence, in the next process, we match each frame of the QSF dataset $S^{Q'}$ and the GCFSs $S_{\gamma}^{D^*}$.

Due to false matching or the same movement between signs, there can be two different signs grouped into same the sign group. This can be solved by using the algorithm in the next section. The proposed matching model is visualized in Fig. 5.

3.4.2 Sign recognition with AGM

The AGM is used to recognize the query sign and convert it into text or voice. The motion of hands with respect to the head or torso is defined by the Euclidian distance between the vertices on the signer's hand and the other parts of the body. Similarly, the hand shape is defined by distances within the hand group. The Euclidian distance is minimum when the vertices on the tips of all fingers come close to each other; this indicates that the hand is in the closed shape. If all the fingers are away, the distances between the vertices on the tips of fingers are large; this implies an open palm. Hence, we estimate the vertex and edge match for the input signed query $S^{Q'}$ and extracted sign group $S_{\gamma}^{D^*}$ from the dataset.

The vertex matching similarity index is calculated using the following expression:

$$m_S^v(T) = \bigcup_{i=1}^{t_Q} \left\| v_{i_l}^{Q_i} - v_{\{i_1,i_2,....,i_n\}}^{D_i^*} \right\|_2^2 \quad \forall\ l = 1:n \quad \in \mathrm{R}^{n\times n\times t_Q\times t_{D^*}} \tag{17}$$

A 4D matrix of size $n \times n \times t_Q \times t_{D^*}$ is obtained, which indicates that all nodes in one frame are matched with the nodes in all the other video frames and the process continuous for all the frames in the first set. This 4D matrix is a matching matrix between the signs. The formula to calculate the sign similarity match index for identifying the sign in the frames can be given as follows:

$$m_g^e = \bigcup_{\forall\gamma} \frac{\sum_{i=1}^{n}\sum_{j=1}^{n} m_S^v(i,j)}{n^2} \in \mathrm{R}^{t_Q\times t_{D^*}} \tag{18}$$

where t_{D^*} is the number of frames in the dataset sign group γ. The matrix m_s^{γ} is a non-diagonal matrix that indicates the matches between frames in the query sign and dataset group. To find the best match or matches for the query sign in the dataset sign group, we must compute the edge similarity index between the query sign graph and dataset sign graph. Because 3D hand shapes are accurately represented using edges, the edge similarity index is calculated as

$$m_S^e(T) = \bigcup_{ii=1}^{t_Q} \left\| v_{i_l,j_l}^{Q_{ii}} - v_{i_1'}^{D_{ii}^*} \right\|_2^2 - \left\| T(v_{j_l}^{Q_{ii}}) - T(v_{j_l'}^{D_{ii}^*}) \right\|_2^2 \in \mathrm{R}^{(n-1)\times(n-1)\times t_Q\times t_{D^*}} \quad \forall\ l = n-1 \tag{19}$$

For each dataset group,

$$m_{\gamma}^e = \bigcup_{\forall\gamma} \frac{\sum_{i=1}^{n}\sum_{j=1}^{n} m_S^e(i,j)}{n^2} \in \mathrm{R}^{t_Q\times t_{D^*}} \tag{20}$$

Finally, to identify the correct sign group, we conduct sign similarity matching on each group vertex and edge similarity matrices as follows:

$$S_s^{\gamma} = \prod_{jj=1}^{g} m_{\gamma}^{v}(jj)\, m_{\gamma}^{e}(jj) \tag{21}$$

S_s^{γ} provides the diagonal similarity matrix that displays matching between the query sign and sign groups. The maximum matching coefficient obtained after matches the query sign index and the dataset index is used to decode the text and speech component for the input query sign.

The following algorithm was experimentally tested on the Indian sign language dataset that comprises 200 signs in the form of continuous sentences.

4 Experimental results with discussion

In this section, the experimental results obtained after testing the proposed AGM method on the Indian sign language 3D datasets are proposed. Moreover, the results are compared with other state-of-art GM techniques that involve the use of temporal pyramids, spectral matching, and AGK. We test the performance of our AGM method by using the word recognition frequency (WRF) given by the following expression:

$$R_{wmf} = \prod_{i=1}^{g} \frac{N\left(S_s^i\right)}{N\left(S_g\right)} \times 100 \tag{22}$$

where $N\left(S_s^i\right)$ is number of times a sign in the QSF matches a sign in the CSF dataset in the group and $N\left(S_{\gamma}\right)$ represents the number of signs in the group.

4.1 Indian sign language 3D dataset

The experiments were performed on a 3D Indian sign language dataset created at the Biomechanics and Vision Computing research lab, K. L. E. F (Deemed-to-be-University), India. The optical 3D motion setup comprised eight infrared cameras and one RGB video camera to capture the signs. Figure 6 displays the setup with the camera and marker positions designed for the capturing the signs.

The cameras were adjusted in height, focus, and viewing angle for minimizing data loss during movements. Each optical camera captures the movement of the markers at 120 fps. The major problem encountered during 3D motion capture of ISL is in the marker design. Markers on hands are difficult to capture when the hands are moving in all directions. The 3D template consists of 57 markers that are categorized as 18 left hand, 18 right hand, two shoulder, one chest, two arm, 12 face, and four head markers. The template in Fig. 3 is obtained after testing different marker positions. The selected model template produced the best capture information for the 200 signs [22, 23].

A continuous meaningful sentence is captured to represent the 3D dataset in ISL as CSF. The paragraph has 200 signs, and 24 signs are repeated more than once in the dataset. Few examples of the sentences are presented as follows: “Hi, **Good**, I am **good**. Hope you all are doing well. Drink tea and eat biscuits. Women are beautiful, and men are handsome. *I* welcome you all. My name is A N I L K U M A R. My father’s name is R A M A R A O. My mother’s name is R A M Y A. *I* am the only child born in my family. How are you. *I* am

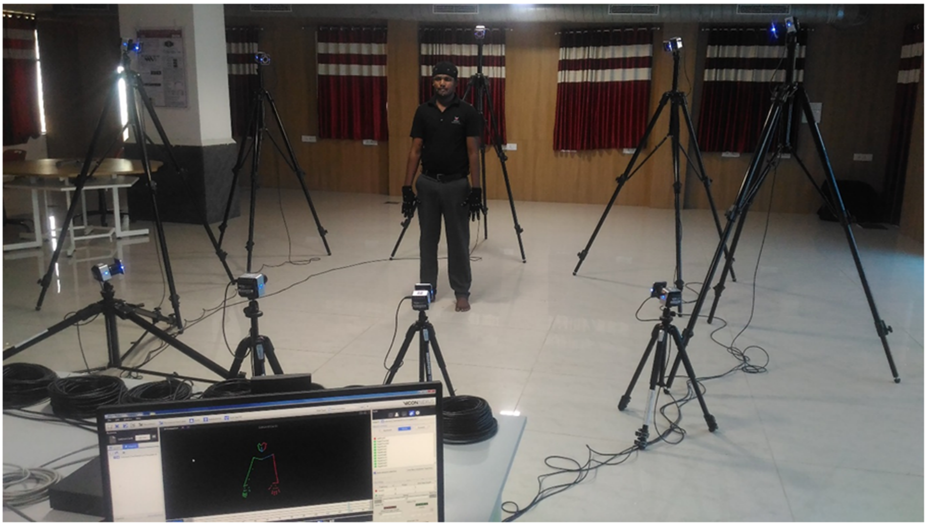


Fig. 6 3D motion capture setup with the camera and marker positions

good. My wife is beautiful. Her sister is a **good** woman. *I* have a son and a daughter. They go to **good** school. What is that product? *I* am the hardest working person in my office, and *I* enjoy work. Mornings are happy, afternoons are good, and evenings are sad. I am the head of the product design group at K. L. University. Currently, design is an important phase of engineering process development. In this short communication, we will learn how to design sign language modules for automatic SLR. We focus on a video-based sign language model. This sign language project was stationed in the year 2016 as part of the technology for the disabled and elderly program by the department of science and technology, government of India. The principle investigator is Dr. P. V. V. Kishore."

The aforementioned sentences are formed such that a particular word is repeated multiple times, as it happens during verbal conversations. The words "good," "I," and "my" appears four, five, and six times, respectively, in the dataset.

For testing, we a database with different sentences by using the repeated words "good," "I," and "my" as follows: "Good afternoon, I am A N I L. I am the only child to father and mother. I and my wife enjoy work. We buy products in the evening." The repeated words are the same in the training and testing sentences conveying a different meaning. This model of training and testing provides faster outputs compared with the word to word matching of the signs. In this study, we propose sentence to sentence matching on the basis of four cases: (i) same training and query datasets with signs stored in the same order, (ii) same training and query datasets with signs stored in a different order, (iii) different testing dataset with the same words, and (iv) different testing dataset by involving words from the testing dataset mixed with new words whose sign movements or shapes are similar to the old words.

Performance of the AGM algorithm was tested on 200 words by using two parameters. True WRF (TWRF) refers to the signs that exactly match the signs in the dataset. However, false WRF (FWRF) is calculated for the signs that match signs of other words in the dataset. Both performance measures are calculated by using (22).

The transformation in the vertex and edge (T) is set to one. This is due to minute transformations in the adjacent graph topologies while performing a sign. The graphs generated from the CSF dataset and QSFs are fully un-connected non-directional graphs. The vertices and edges of the graph are reconstructed effectively to represent the 57 marker points by using the nexus software, thus eliminating the scope for noisy measurements.

4.2 AGM conducted by using the same dataset for training and testing in sequence

The first experiment tests the performance of the AGM algorithm on a continuous sign sequence that involves 200 words to form meaningful sentences, as presented in the previous section. Consider the sentences *Hi, Good Morning, I am good. Hope you all are doing well. Drink tea and eat biscuits. Women are beautiful, and men are handsome.* The sentences have multiple repeated words, which have the same sign and is a common feature during communication.

This experiment tests the AGM model by using the same training and testing data in the same sequence. We use the same 200-word sentences in the same order for testing and training. The results for a few aforementioned sentences are shown below. The confusion matrix obtained after the vertex matching by using (18) is shown in Fig. 7.

By using (22), the TWRF value is estimated to be approximately 100% and the FWRF is 45%. On the basis of the standards of SLR, the obtained FWRF value is high. The high FWRF is due to the similar hand movements in the signer's space for every sign. Hand movements in a continuous video sequence encounter a similar point space for closely matched signs.

To overcome this difficulty, edge matching is performed using (20) on the dataset that produces a TWRF of 100% and FWRF of 85%. Edges are links between two joints, and the joint noise while performing the signs is higher compared with the vertex noise.

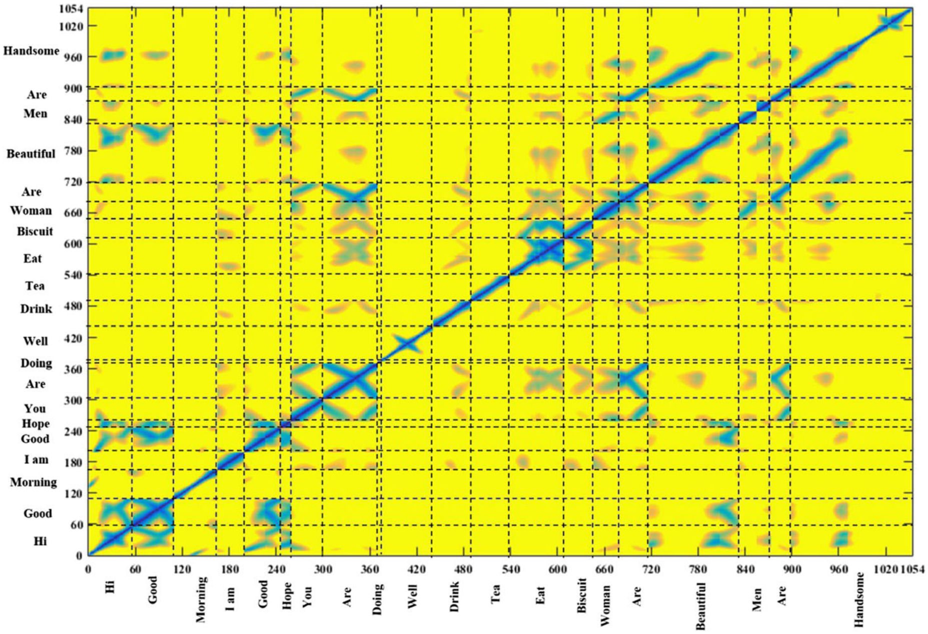


Fig. 7 Vertex GM confusion matrix

The noise term is used to indicate similar joints for two or more signs. This fact can be observed in the edge confusion matrix for the same testing sentences, as displayed in Fig. 8.

Figure 8 displays optimal matching with edges with almost 100 percentage of true matching but false matchings of the signs also increase proportionally. It is crucial to make the system independent of the 45% FWRF in the vertex matching and 85% FWRF in the edge matching. Therefore, (21) is employed, which is the product of the vertex and edge matching matrices. The confusion matrix generated using (21) is shown in Fig. 9. The small, but effective, matching that occurred for each sign with the adjacent signs was eliminated because the error value increased due to multiplication.

By analyzing the confusion matrices displayed in Figs. 7, 8, and 9, we observe that the TWRF value is 100%, whereas the FWRF value is considerably decreased to 12.55%. The matrices in these figures are obtained using (22). For example, the query sign “Hi” matches perfectly with the sign for “Hi” in the stored database. The “Hi” sign has a video dynamic range of 1–56 3D frames. There is a strong adjacency effect as the algorithm matches all vertices with all other vertices in the same frame.

The presence of the adjacency effect can be seen in all figures. However, this does not show any effect on the recognition rate. The sign “Good” required 57 to 111 frames, and the matching of this sign is perfect with the same sign in the testing sequence. However, because we have considered marker trajectories in this study, the hand location for the sign “Hi” and “Good” are similar. This can be observed in Fig. 10 that displays a set of five frames that are randomly picked from the sequence.

The dark regions in the confusion matrix have an approximate value of zero, thus implying a perfect match. The yellow region represents large numerical error values. The signs “Hi” and “Good” show some resemblance with the sign “beautiful.” The sign beautiful is

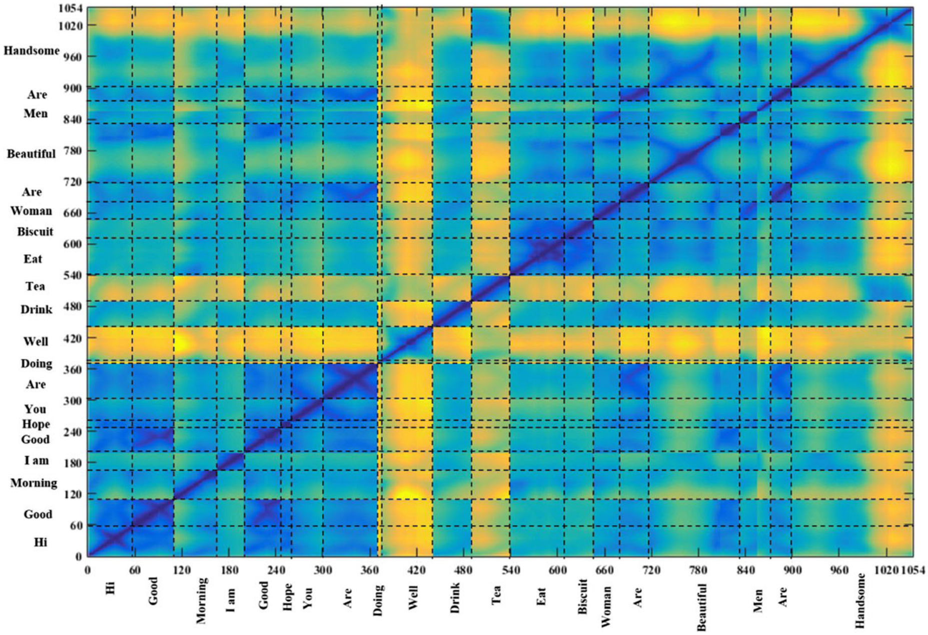


Fig. 8 Edge GM confusion matrix

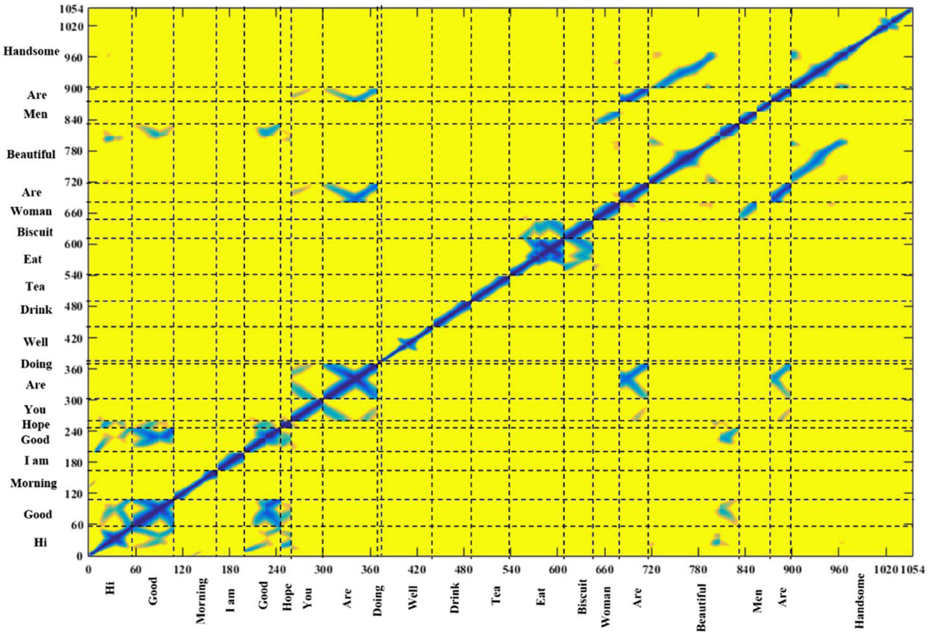


Fig. 9 Confusion matrix obtained using (21) for the testing sentences in ISL

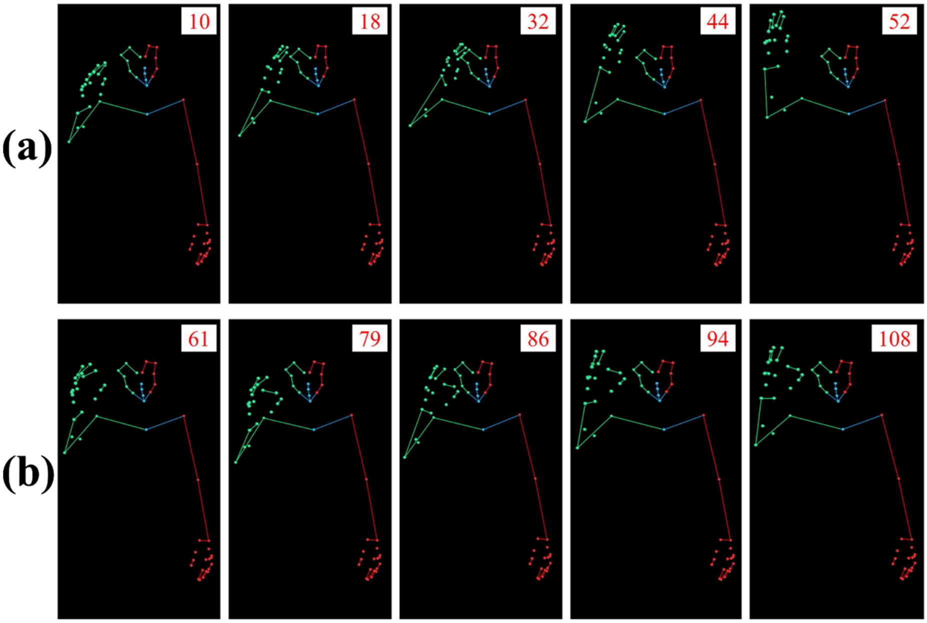


Fig. 10 3D motion captured sign frames for the words **a** "hi" and **b** "good"

timed between frames 719–833. Figure 11 shows the 3D skeleton of the signs “hi,” “good,” and “beautiful” in the four frames. The trajectories of the markers in all the three signs are closely related; hence, a small false recognition appears for these signs. However, the error values are not exactly zero; hence, these false matching do not hinder the performance of the algorithm. These problems are common in 2D-data-based SLR, otherwise are fully rectified in 3D.

For the word sign groups in which similar words appear multiple times in a conversation, AGM algorithm is the best algorithm for identifying the word sign groups. This conclusion can be arrived at by analyzing Fig. 9. The following model is used to the find p word sign groups of similar signs in a continuous sentence:

$$p = \frac{d\left(S_s^{\gamma}\right)}{dS^t} \tag{23}$$

These p groups are clustered into a word sign group such as “are” in the test sentence and can be seen in Fig. 9.

The word sign appears thrice, and these “are” signs are grouped together into one set. This process will speed up the system during testing by using the query sequence, where the algorithm identifies that there are three “are” sign groups in the test sequence. The query “are” matches at least three locations, which eliminates the multiple indexing problems for multiple instances of the sign words.

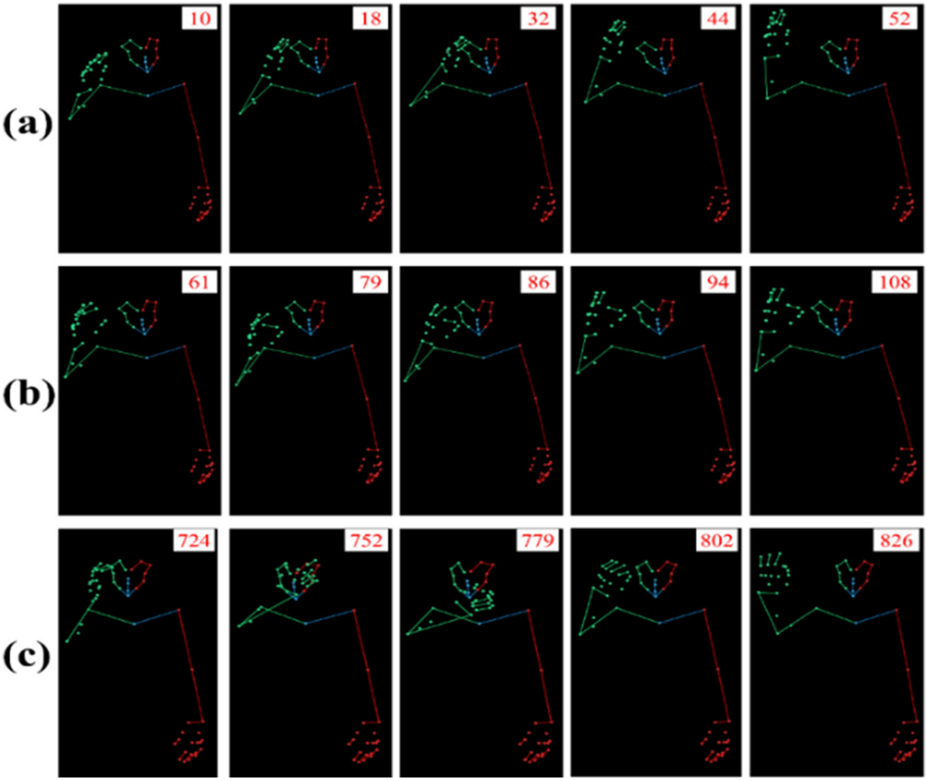


Fig. 11 Hand position similarity in the signs **a** “hi,” **b** “good,” and **c** “beautiful”

4.3 AGM by using the same dataset for training and testing with words not in sequence

In experiment 2, we supply a query set that consists of the same words but are arranged in a different sequence. Consider the query sentences “Hello, I am good. Hope that the morning biscuits you eat are good. Beautiful women are doing well. Handsome men drink tea.” The resulting confusion matrix for the QSF is shown in Fig. 12.

The TWRF value is 100%, as shown in Fig. 12. The FWRF value is very small and is in the range of 6%–8%. Multiple changes in sequences are tested, and the AGM algorithm classified each sign with 100% accuracy. When grouping is initiated after the first testing is completed, all the similar signs were grouped under two labels. One label is the sign and other is the starting and ending frames of the signs in the group. During the second testing round, the same word appears multiple times such as the sign for “are” excludes the motion extraction phase and extracts the labeled sign from the CSF dataset. In the experiment 3, we propose using the same CSF and QSF dataset by using a different signer.

4.4 AGM conducted by using different dataset for training and testing in sequence

To check the robustness of the proposed AGM algorithm for sign matching on different human signers, the experiment 3 was designed. Here, the QSF dataset is captured by using a different signer. The body dynamics change when the signer changes. Therefore, handling the variations in the trajectories or the locations of markers in a frame is a challenging task.

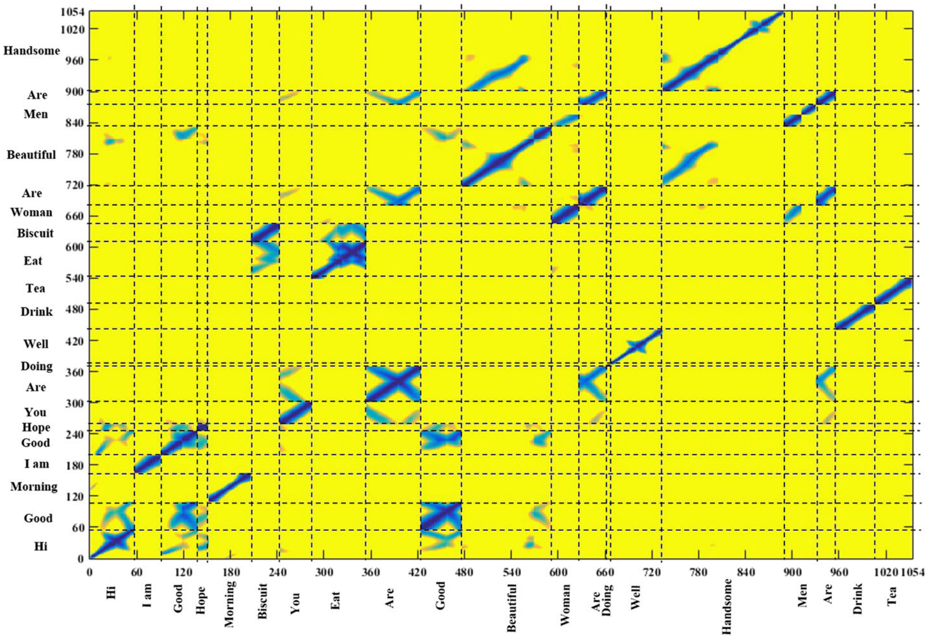


Fig. 12 Confusion matrix for experiment 2 by using AGM out-of-sequence QSFs

Fig. 13. displays the confusion matrix obtained by using different signers to generate the CSF and QSF databases.

The matching pixels in the confusion matrix are not too dark. This indicates that the Euclidian distance function does not produce a zero value for similar signs. The normalized distance error is in the range of 0.098–0.155 for the tested signs. However, the TWRF value calculated from (22) is 100%. There are no signs that the system failed to reproduce during the recognition phase. However, there are instances when false matching occurred due to the reasons discussed in the previous sections. In this case, the FWRF value was 10.25% due to the overlapping in the sign trajectories. The final experiment pertains to matching signs that are unknown or are similar to the known signs. This experiment was conducted by using different signers.

4.5 AGM conducted by using different datasets for training and testing words that are not in sequence

The experiment 4 tests the system's recall rate when the QSFs is packed with different words that are closely related to the words in the CSF database. Some of the new words in the QSFs are "Water," "Evening," "My," "Due to," "From," and "in." Some interesting facts were identified during this matching process. Figure 14. displays the resulting confusion matrix.

By close observation, we can see that the "morning" and "evening" sign match has an FWRF value of 93.45%. However, it can be interpreted as negative FWRF, as shown in Fig. 14. The signs of both words contain the same hand shapes; however, the hand

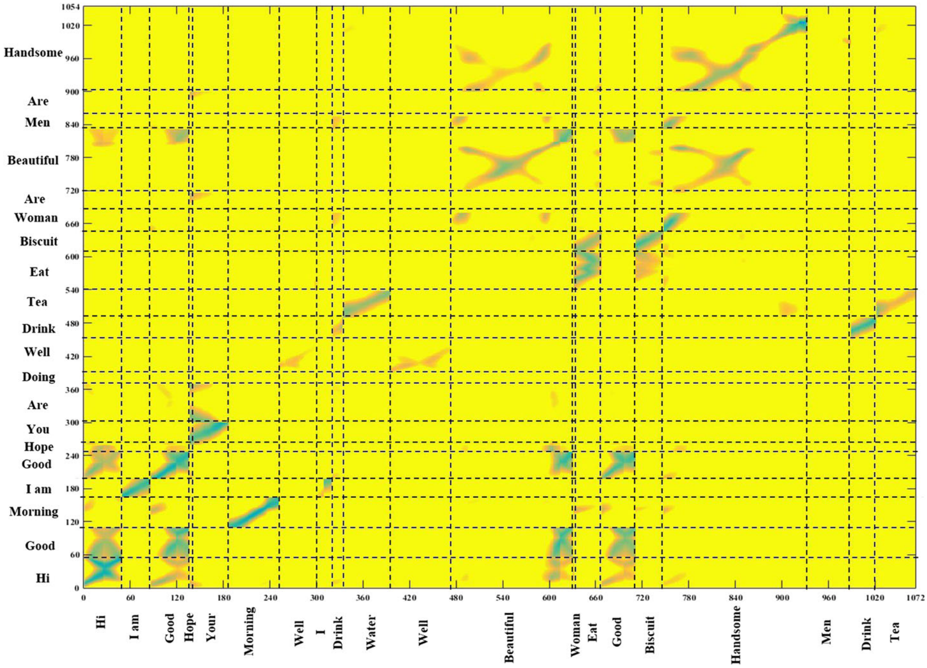


Fig. 13 Confusion matrix of experiment 3 obtained by using different signers to generate the CSF and QSF databases for an out-of-sequence 3D sign video

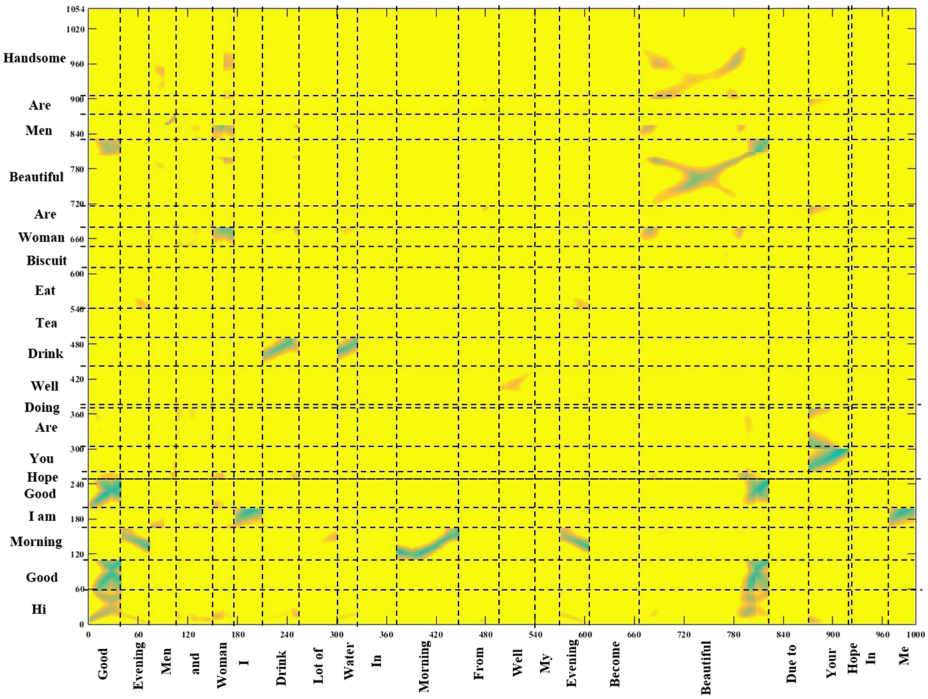


Fig. 14 Confusion matrix of experiment 4 obtained when the CSF database and mixed QSFs comprise known and unknown signs that are used in the CSF database

trajectories for the words are in the reverse direction. Therefore, the matching numbers are in the reverse order. Similarly, the words “beautiful” and “handsome,” “drink” and “water,” “you” and “your,” and “me” and “I” or “I am” have similar signs and show a 100% TWRF. The word “well” has the least TWRF value of 66.4%.

4.6 Performance testing

The confusion matrices presented above are designed on a frame to frame basis. Interestingly, the experimental results are obtained by using two techniques of GM that are presented in [29] and [18] for the sample 3D sentence. In [29], GM is conducted on 3D data by using a temporal pyramid structure. In [18], spectral GM is used for 3D data classification. The objective of the sign language recognizer is to retrieve similar signs that belong to a class of QSFs. Given a QSF sentence, the retrieval accuracy of the proposed method is estimated by computing the TWRF ratio. Moreover, the accuracy of the proposed method is compared with those of [29] and [18]. Figure 15 shows the TWRF for an individual set of words by using the proposed AGM, GM with temporal pyramids, spectral GM, and the proposed AGM on 2D data. The average TWRF over the entire range of signs for the proposed method, [18, 29], and 2D data are 98.32%, 94%, 86%, and 59.36%.

The ability of the algorithm in recalling QSFs with high accuracy decides the real-time recognition capabilities of the algorithm. Here, accuracy is the ratio of retrieving a sign

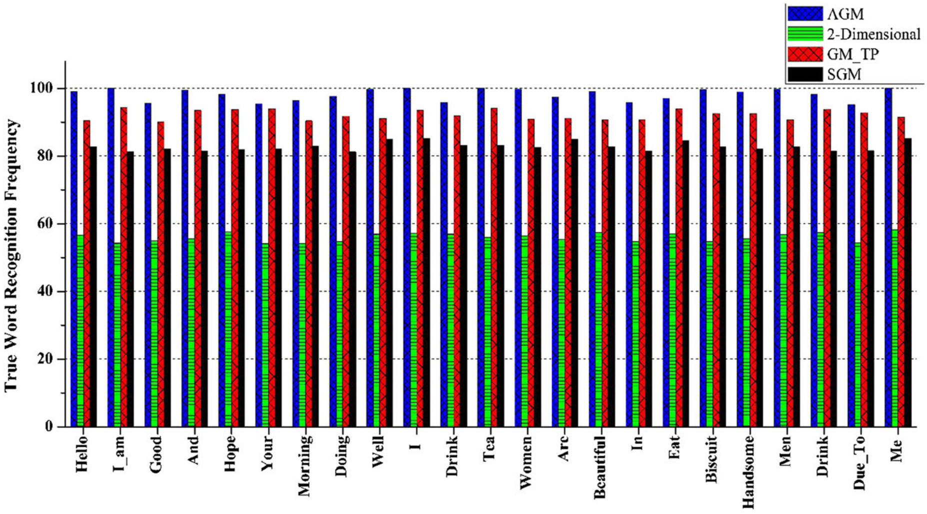


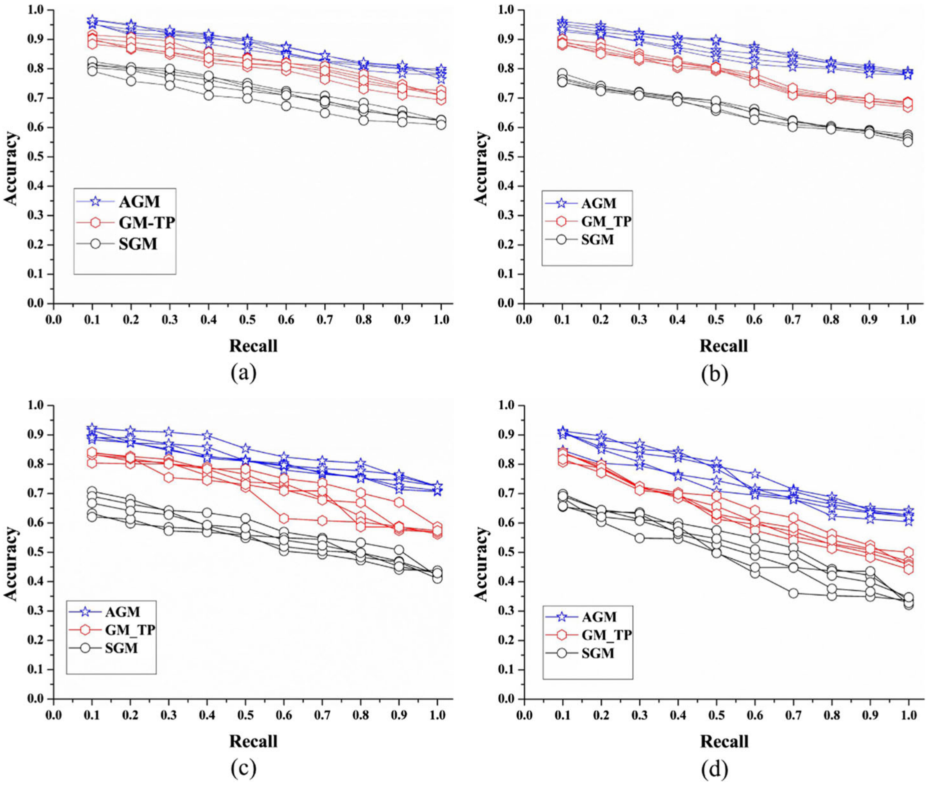
Fig. 15 TWRF of the proposed method and other methods in [18, 29], and that using 2D data

correctly to the total number of signs retrieved. Recall is the ratio of correctly retrieved sign and total number of retrieved signs. Figure 16 displays the comparison of the three GM models for five different signers with the 200-word dataset by using the four aforementioned cases. The proposed model displayed an optimal matching accuracy and recall compared with those provided by the methods in [29] and [18].

The proposed method is faster and independent of number of frames in the query video and database video signs. This gives an undue advantage to the human signer to perform the action sign at his or her own pace. When compared to similar graph matching algorithms on our sign language data, we found them as slower and frame dependent. The slowness in the algorithms is caused due to frame to frame similarity checking in query and database videos. The uneven frames in query and database videos is handled with pyramid model, which is based on manual considerations of pyramid sizes. Unfortunately, our proposed method is computationally intense during the sign identification phase from the continuous database.

The proposed AGM model is validated on benchmark action datasets HDM05 and CMU, which are captured with 3D motion captured technology. We used 20 actions with 10 subjects from both datasets spanning over 200 videos. Table 1, records the recognition rates for our sign dataset and two benchmark datasets HDM05 and CMU for same subject and cross subject matching. The recognition rates are averaged with respect to number of samples in the datasets. The success of the proposed method is attributed to one to many matchings between the graph vertices and edges as opposed to one to one matching or global matching proposed by other algorithms in the table. Our algorithm uses product of edge and vertex matching for recognition, whereas any one matching is used for recognition in all other methods. Moreover, the cross subject performance of the proposed algorithm is superior to other algorithms, due to one to many adaptive graph matching.

In future, the 3D sign language models will be used to build a augmented reality based sign language recognition system for translating 2D video signs into voice or text.



(a) (b) (c) (d)

Fig. 16 Accuracy–recall comparison plots for **a** experiment 1: same testing and training data in sequence, **b** experiment 2: same testing and training data that is not in sequence, **c** experiment 3: different testing and training data in sequence, and **d** experiment 4: different testing and training data with closely matched new words

Table 1 Performance comparison for the proposed and the state-of-the-art methods on the HDM05, CMU and 3D Sign Datasets

Method	HDM05		CMU		3D Sign	
	Same Subject	Cross Subject	Same Subject	Cross Subject	Same Subject	Cross Subject
Graph Matching_Temporal Pyramid [29]	94.42	91.14	91.61	87.66	94.63	90.49
Spectral Graph Matching [18]	92.63	90.61	90.32	88.03	93.65	89.22
Histogram [5]	89.35	87.34	84.49	81.15	88.63	84.37
Weighted Graph Matching [46]	91.37	87.09	89.39	87.28	89.72	86.59
Dynamic Time Wraping [27]	88.01	82.01	85.06	78.69	86.42	83.91
Eigenvalues [20]	90.42	88.46	92.09	87.72	91.59	90.45
Our Proposed	98.99	97.78	97.68	96.96	99.35	98.21

5 Conclusion

In this study, we proposed sign language automation, which is a challenging task. A database involving 3D signs from the Indian sign language is created. 3D position trajectories are used as features for creating and representing signs in the form of undirected graphs. The process was initiated by developing a 3D template design for the signer's that can fully capture all the signs in the Indian sign language. Motion segmentation and AGM models are utilized on the sign data to recognize the signs and convert them to text. The proposed AGM for the sign language has two improvements over the previously proposed GM models that pertain to shape extraction and relative frame extraction with high precision. This significantly improves the accuracy of matching in continuous videos for simultaneous sign identification and recognition. The TWRF is approximately 100% for the signs that are tested by us using AGM on 3D motion capture data. However, the computations required by the proposed model must be decreased to enable a real-time recognition. The 3D motion capture based sign language recognition forms a basis for producing a 3D model based mobile sign language recognizer.

Acknowledgements This work was supported in part by the research project scheme titled "Visual – Verbal Machine Interpreter Fostering Hearing Impaired and Elderly", by the "Technology Interventions for Disabled and Elderly" programme of the Department of Science and Technology, SEED Division, Govt. of India, Ministry of Science and Technology under Grant SEED/TIDE/013/2014(G).

References

1. Agarwal A, Thakur MK (2013) Sign language recognition using microsoft kinect. In: 2013 Sixth international conference on contemporary computing (IC3), IEEE. https://doi.org/10.1109/ic3.2013.6612186
2. Aggarwal J, Xia L (2014) Human activity recognition from 3d data: a review. Pattern Recogn Lett 48:70–80. https://doi.org/10.1016/j.patrec.2014.04.011
3. Almeida SGM, Guimarães FG, Ramírez JA (2014) Feature extraction in brazilian sign language recognition based on phonological structure and using RGB-d sensors. Expert Syst Appl 41(16):7259–7271. https://doi.org/10.1016/j.eswa.2014.05.024
4. Ansari ZA, Harit G (2016) Nearest neighbour classification of indian sign language gestures using kinect camera. Sadhana 41(2):161–182. https://doi.org/10.1007/s12046-015-0405-3
5. Barnachon M, Bouakaz S, Boufama B, Guillou E (2014) Ongoing human action recognition with motion capture. Pattern Recogn 47(1):238–247
6. Belgacem S, Chatelain C, Paquet T (2017) Gesture sequence recognition with one shot learned CRF/HMM hybrid model. Image Vis Comput 61:12–21. https://doi.org/10.1016/j.imavis.2017.02.003
7. Borzeshi EZ, Piccardi M, Xu RYD (2011) A discriminative prototype selection approach for graph embedding in human action recognition. In: 2011 IEEE International conference on computer vision workshops (ICCV workshops), IEEE. https://doi.org/10.1109/iccvw.2011.6130401
8. Cahill-Rowley K, Rose J (2017) Temporal–spatial reach parameters derived from inertial sensors: Comparison to 3d marker-based motion capture. J Biomech 52:11–16. https://doi.org/10.1016/j.jbiomech.10.031
9. Çeliktutan O, Wolf C, Sankur B, Lombardi E (2014) Fast exact hyper-graph matching with dynamic programming for spatio-temporal data. J Math Imaging Vision 51(1):1–21. https://doi.org/10.1007/s10851-014-0503-6
10. Chai X, Li G, Chen X, Zhou M, Wu G, Li H (2013) Visualcomm: a tool to support communication between deaf and hearing persons with the kinect. In: Proceedings of the 15th international ACM SIGACCESS conference on computers and accessibility, ACM, p 76
11. Cui J, Liu Y, Xu Y, Zhao H, Zha H (2013) Tracking generic human motion via fusion of low-and high-dimensional approaches. IEEE Trans Syst Man Cybern Syst 43(4):996–1002
12. Duan J, Zhou S, Wan J, Guo X, Li SZ (2016) Multi-modality fusion based on consensus-voting and 3d convolution for isolated gesture recognition. arXiv:1611.06689

13. Geng L, Ma X, Wang H, Gu J, Li Y (2014) Chinese sign language recognition with 3d hand motion trajectories and depth images. In: Proceeding of the 11th world congress on intelligent control and automation, IEEE. https://doi.org/10.1109/wcica.2014.7052933
14. Grest D, Krüger V Gradient-enhanced particle filter for vision-based motion capture. In: Human motion–understanding, modeling, capture and animation. Springer, Berlin, pp 28–41
15. Gärtner T, Flach P, Wrobel S (2003) On graph kernels: hardness results and efficient alternatives. In: Learning theory and kernel machines. Springer, Berlin, pp 129–143
16. Guess TM, Razu S, Jahandar A, Skubic M, Huo Z (2017) Comparison of 3d joint angles measured with the kinect 2.0 skeletal tracker versus a marker-based motion capture system. J Appl Biomech 33(2):176–181. https://doi.org/10.1123/jab.2016-0107
17. Han F, Reily B, Hoff W, Zhang H (2017) Space-time representation of people based on 3d skeletal data: a review. Comput Vis Image Underst 158:85–105. https://doi.org/10.1016/j.cviu.2017.01.011
18. Huang P, Hilton A, Starck J (2010) Shape similarity for 3d video sequences of people. Int J Comput Vis 89(2-3):362–381
19. Jeong YS, Jeong MK, Omitaomu OA (2011) Weighted dynamic time warping for time series classification. Pattern Recogn 44(9):2231–2240
20. Junejo IN, Aghbari ZA (2012) Using SAX representation for human action recognition. J Vis Commun Image Represent 23(6):853–861
21. Kakadiaris I, Barrón C Model-based human motion capture. In: Handbook of mathematical models in computer vision, Springer, pp 325–340
22. Kishore PVV, Kumar DA, Sastry ASCS, Kumar EK (2018) Motionlets matching with adaptive kernels for 3-d indian sign language recognition. IEEE Sensors J 18(8):3327–3337. https://doi.org/10.1109/jsen.2018.2810449
23. Kumar Eepuri K, Kishore PSSASC, Maddala TKK, Kumar Dande A (2018) Training CNNs, for 3d sign language recognition with color texture coded joint angular displacement maps. IEEE Signal Processing Letters :1–1. https://doi.org/10.1109/lsp.2018.2817179
24. Kumar P, Gauba H, Roy PP, Dogra DP (2017) Coupled hmm-based multi-sensor data fusion for sign language recognition. Pattern Recogn Lett 86:1–8
25. Kumar P, Gauba H, Roy PP, Dogra DP (2017) A multimodal framework for sensor based sign language recognition. Neurocomputing 259:21–38
26. Kushwah MS, Sharma M, Jain K, Chopra A (2016) Sign language interpretation using pseudo glove. In: Proceeding of international conference on intelligent communication, control and devices, Springer, Singapore, pp 9–18
27. Leightley D, Li B, McPhee JS, Yap MH, Darby J (2014) Exemplar-based human action recognition with template matching from a stream of motion capture. In: Lecture notes in computer science, Springer international publishing, pp 12–20
28. Li K, Zhou Z, Lee CH (2016) Sign transition modeling and a scalable solution to continuous sign language recognition for real-world applications. ACM Transactions on Accessible Computing 8(2):1–23. https://doi.org/10.1145/2850421
29. Li M, Leung H, Liu Z, Zhou L (2016) 3d human motion retrieval using graph kernels based on adaptive graph construction. Comput Graph 54:104–112
30. Li SZ, Yu B, Wu W, Su SZ, Ji RR (2015) Feature learning based on SAE–PCA network for human gesture recognition in RGBD images. Neurocomputing 151:565–573. https://doi.org/10.1016/j.neucom.2014.06.086
31. Liu L, Cheng L, Liu Y, Jia Y, Rosenblum DS (2016) Recognizing complex activities by a probabilistic interval-based model. In: AAAI, vol 30, pp 1266–1272
32. Liu Y, Nie L, Han L, Zhang L, Rosenblum DS (2015) Action2activity: recognizing complex activities from sensor data. In: IJCAI, vol 2015, pp 1617–1623
33. Liu Y, Nie L, Liu L, Rosenblum DS (2016) From action to activity: sensor-based activity recognition. Neurocomputing 181:108–115
34. Lu Y, Wei Y, Liu L, Zhong J, Sun L, Liu Y (2017) Towards unsupervised physical activity recognition using smartphone accelerometers. Multimedia Tools and Applications 76(8):10701–10719
35. Mapari RB, Kharat G (2016) American static signs recognition using leap motion sensor. In: Proceedings of the Second international conference on information and communication technology for competitive strategies, ACM, p 67
36. Moeslund TB, Granum E (2001) A survey of computer vision-based human motion capture. Comput Vis Image Underst 81(3):231–268

37. Nai W, Liu Y, Rempel D, Wang Y (2017) Fast hand posture classification using depth features extracted from random line segments. Pattern Recogn 65:1–10. https://doi.org/10.1016/j.patcog.2016.11.022
38. Park S, Park H, Kim J, Adeli H (2015) 3d displacement measurement model for health monitoring of structures using a motion capture system. Measurement 59:352–362. https://doi.org/10.1016/j.measurement.2014.09.063
39. Rao GA, Kishore P (2017) Selfie video based continuous indian sign language recognition system. Ain Shams Engineering Journal. https://doi.org/10.1016/j.asej.2016.10.013
40. Rucco R, Agosti V, Jacini F, Sorrentino P, Varriale P, Stefano MD, Milan G, Montella P, Sorrentino G (2017) Spatio-temporal and kinematic gait analysis in patients with frontotemporal dementia and alzheimer's disease through 3d motion capture. Gait Posture 52:312–317. https://doi.org/10.1016/j.gaitpost.2016.12.021
41. Sandler W (2017) The challenge of sign language phonology. Annual Review of Linguistics 3:43–63
42. Sun C, Zhang T, Bao BK, Xu C, Mei T (2013) Discriminative exemplar coding for sign language recognition with kinect. IEEE Transactions on Cybernetics 43(5):1418–1428. https://doi.org/10.1109/tcyb.2013.2265337
43. Sun S, Luo C, Chen J (2017) A review of natural language processing techniques for opinion mining systems. Information Fusion 36:10–25. https://doi.org/10.1016/j.inffus.2016.10.004
44. Sun Y, Bray M, Thayananthan A, Yuan B, Torr P (2006) Regression-based human motion capture from voxel data. In: Procedings of the british machine vision conference 2006. British machine vision association
45. Ta AP, Wolf C, Lavoue G, Baskurt A (2010) Recognizing and localizing individual activities through graph matching. In: 2010 7Th IEEE international conference on advanced video and signal based surveillance, IEEE. https://doi.org/10.1109/avss.2010.81
46. Xiao Q, Wang Y, Wang H (2014) Motion retrieval using weighted graph matching. Soft Comput 19(1):133–144. https://doi.org/10.1007/s00500-014-1237-5
47. Yang C, Cheung G, Stankovic V (2017) Estimating heart rate and rhythm via 3d motion tracking in depth video. IEEE Trans Multimedia 19(7):1625–1636. https://doi.org/10.1109/tmm.2017.2672198
48. Zhang W, Liu Z, Zhou L, Leung H, Chan AB (2017) Martial arts, dancing and sports dataset: a challenging stereo and multi-view dataset for 3d human pose estimation. Image Vis Comput 61:22–39. https://doi.org/10.1016/j.imavis.2017.02.002
49. Zhang Z, Kurakin AV Dynamic hand gesture recognition using depth data (2017). US Patent 9,536, 135

D. Anil Kumar received the B.Tech degree from the University of JNTUK, India in 2014, and the M.Tech degree from the university of Koneru Lakshmaiah Education Foundation, in 2016. He is currently pursuing the Ph.D. degree with the department of Electronics and Communications Engineering from the same university. His research interests are in the area of video processing, computer vision and sign language machine translation. His work is mainly focused on the development of 3-D processing algorithms for computer vision applications.

A. S. C. S. Sastry received the M.Tech. and Ph.D. degrees in Electronics and Communications Engineering from JNTU college of engineering, Kakinada, India. Currently, he is a Professor with the Department of Electronics and Communication Engineering and he is the associate dean academics in the University of Koneru Lakshmaiah Education Foundation. His research includes adaptive signal processing, biomedical signal processing, medical image processing and human-computer interaction. In these areas, he has authored more than 65 publications in journals and conferences.

P. V. V. Kishore is a professor of Image & Video Processing with the department of Electronics and Communications Engineering, where he manages the Image, Speech and Signal processing Research Group. He went on to study M.Tech at Cochin University of science and technology and Ph.D. from Andhra University College of engineering in 2013. He is the chair of the Biomechics and vision computing research center.

His works focus on mechine learing, biomechanics, artificial intelligence, human motion analysis and sign language machine translation. His research explores how motion capture data models can effectively model low end video objects in real time for better recogntion and analysis. He is particularly intersted in developing new innovations in the areas of computer vision and mechine learing. He has authored several publications in these fields.

E. Kiran Kumar received the B.Tech degree in Electronics and Communication Engineering from the JNT University, Kakinada, India, in 2009, M.Tech degree in Systems and Signal Processing from the same University, in 2013, specializing in evolving optimized object segmentation and recognition. Currently, he is working as Junior Research Fellow in the project “Visual–Verbal Machine Interpreter Fostering Hearing Impaired and Elderly” and pursuing the Ph.D. degree from University of Koneru Lakshmaiah Education Foundation, India. His research interests include the analysis of musculoskeletal movements of hand and movement strategies of the wrist and fingers in Indian sign language recognition.