Expressive ASL Recognition using Millimeter-wave Wireless Signals

Panneer Selvam Santhalingam *, Yuanqi Du*, Riley Wilkerson*, Al Amin Hosain *,
Ding Zhang*, Parth Pathak*, Huzefa Rangwala* and Raja Kushalnagar †
*Department of Computer Science, George Mason University, Fairfax, VA, USA
Email: {psanthal, ydu6, rwilker2, ahosain, dzhang13, hrangwal, phpathak}@gmu.edu
†Department of Science, Technology and Mathematics, Gallaudet University, Washington, DC
Email: raja.kushalnagar@gallaudet.edu

***Abstract*—Over half a million people in the United States use American Sign Language (ASL) as their primary mode of communication. Automatic ASL recognition would enable Deaf and Hard of Hearing (DHH) users to interact with others who are not familiar with ASL as well as voice-controlled digital assistants (e.g., Alexa, Siri, etc.). While ASL recognition has been extensively studied, there is a little attention given to recognition of ASL non-manual body markers. The non-manual markers are typically expressed through head, torso and shoulder movements, and add essential meaning and context to the signed sentences. In this work, we present ExASL, a sentence-level ASL recognition system using millimeter-wave radars. ExASL can recognize manual markers (hand gestures) and non-manual markers (head and torso movements). It utilizes multi-distance clustering to recognize body parts and cluster mmWave point clouds. We then present a multi-view deep learning algorithm that can learn from clustered body part representation for an expressive sentence-level recognition. Our evaluation shows that ExASL can recognize ASL sentences with a word error rate of 0.79%, sentence error rate of 1.25%, and non-manual markers with an accuracy of 83.5%.**

I. Introduction

About 30 million people in the United States have bilateral hearing loss, and about 1 million are functionally deaf [1]. Speech production quality is correlated with hearing loss, which can lead to difficulty in spoken communication. Around half a million people in the United States communicate visually through American Sign Language (ASL), and use it as their primary means of communication. Computer-based ASL recognition can enable the Deaf and Hard-of-Hearing (DHH) users to seamlessly interact with others who are unfamiliar with ASL. It can also enable the DHH users to communicate with personal or home digital assistant devices (such as "Siri" on iPhones, "Google Now" on Android smartphones, and "Alexa" on Amazon Echo smart-speakers) that are primarily voice-controlled.

ASL recognition has been primarily studied as a form of gesture recognition problem in previous work. Recognizing ASL hand gestures using RGB video has been investigated extensively [2]–[4] in prior research. Recently, researchers have explored the use of other sensing modalities such as IMU in wearables [5], depth sensors (Kinect and Leap motion) [6], and Radio Frequency (RF) based systems [7]. An important limitation of these existing works is that they only focus on recognizing manual signs (i.e., the hand gestures) of ASL. However, signed languages including ASL have structural complexities that are similar to any other spoken language. ASL includes both manual signs as well as non-manual markers. The manual signs typically correspond to words signed through hand gestures. On the other hand, the non-manual markers add meaning, context and emphasis to the sentences, making them critically important components of ASL. These non-manual markers are expressed through head, torso and shoulder movement along with facial expressions. For example, a signer signing the following three independent manual signs `I, LIKE` and `APPLES` translates to `I LIKE APPLES` in English. But if the signer simultaneously shakes her head (non-manual marker for negation), the translation becomes `I DON'T LIKE APPLES`. Body movements like torso shift add significant expressive power to the language. Signing for `"Fire that"` would imply `"That is fire."`, but when accompanied with a torso shift towards the addressee it turns into the question `"Is that fire?"`. Despite their importance, there exists very little research [8], [9] on recognition of non-manual markers. Additionally, their integration with manual signs to perform contextual, sentence-level ASL recognition remains an outstanding problem.

In this work, we present ExASL which can perform manual as well non-manual marker recognition and can provide an Expressive, sentence-level ASL recognition. ExASL is based on millimeter-wave (mmWave) RF signals, a novel and emerging sensing modality that provides many advantages over existing sensing modalities. Due to their higher operating frequency and large available bandwidth, mmWave signals can provide a very high range resolution for sensing. Compared to other low frequency RF radars (operating at sub 6 GHz), mmWave signals provide better accuracy (less clutter) due to directional communication. Due to these advantages, many mmWave radars have become commercially available [10], [11] with applications in automotive and industrial sensing. mmWave sensing can help in addressing some of the challenges posed by camera and vision based techniques. Vision based ASL recognition requires that user is continuously recorded through an RGB camera, leading to serious privacy concerns. Also, such solutions perform poorly in dark or

978-1-7281-1062-2/20/$31.00 2020 © IEEE

low lighting conditions. With emerging mmWave WLAN networking standards such as 802.11ad/ay, it is also possible to utilize the networking devices for the purpose of sensing. mmWave radars have smaller physical footprint, and can be easily integrated in today's IoT devices (such as Amazon Echo speaker and smartwatch).

mmWave sensing radars estimate range (distance), angle and velocity of an object through reflection of mmWave signals from the object. In case of a multi-point scatterer where multiple points of the object reflect the signal, such estimation results in a 3-dimensional point cloud. The obtained point clouds can be tracked over time to identify the spatial changes of objects surrounding the mmWave radar. In terms of ASL recognition, the point cloud represents body parts (hands, torso, etc.) that move over time when the user perform manual signs and non-manual markers. In this work, we are interested in using the point cloud representation of mmWave radars to perform sentence-level ASL recognition. Recognizing ASL signs and non-manual markers using the obtained point clouds impose multiple challenges. We list the challenges, our proposed solutions and contributions below.

(1) Body part separation for mmWave point clouds: The point cloud representation provided by mmWave radar is not only sparse but also have non-trivial noise due to second order reflections (produced by reflections from objects other than user). For a reliable identification of manual and non-manual markers, it is necessary that the points are associated with specific body parts after noise removal.

We propose a multi-distant clustering algorithm, which enables ExASL to separate and identify the human body parts (Left hand, Right hand, and Torso) from the obtained point cloud data. The separation of body parts is key to non-manual marker recognition, as it allows ExASL to model the underlying parts in isolation, which leads to a reliable recognition. The body part separation, also enables ExASL to model the interaction between different parts, which improves ExASL's recognition performance. The proposed algorithm includes outlier removal as a component, making ExASL resistant to the impact of second order reflections and other objects in the environment.

(2) Sentence-level ASL recognition: Point cloud data generated by ExASL is sparse by nature, without much visual resemblance to the objects (body parts) they represent. The model that ExASL utilizes for ASL recognition should be able to capture the body part interactions and accurately recognize ASL sentences from these sparse representations. Additionally, a model that can jointly recognize and integrate manual and non-manual markers is needed for sentence level recognition.

ExASL utilizes a multi-view deep learning algorithm which explicitly models the interaction between different body parts in recognizing ASL signs. The proposed algorithm, extends existing models for point cloud representation with time sequence modeling (Long Short-Term Memory (LSTM) units), enabling ExASL to recognize ASL signs from a sequence of point cloud data. The presence of hierarchical convolutional layers enables ExASL to learn feature representations, that compensate for the sparse point cloud representations. The algorithm also utilizes Connectionist Temporal Classification (CTC), which allows ExASL to perform ASL sentence recognition without segmentation or frame level labeling.

We extensively evaluate ExASL's performance on ASL recognition with, 23 ASL signs, 29 ASL sentences, and 6 non-manual markers. The evaluation dataset is built on data collected from 5 participants. By taking advantage of ExASL's ability to separate and identify different body parts, we propose a simple data augmentation technique. We quantify the significance of the multi-distant clustering algorithm and multi-view deep learning model by comparing it with a model which does not take advantage of the body part separation. On ASL sign recognition the proposed multi-view deep learning model achieves an accuracy of 92.5% for manual signs (word-level recognition). For ASL sentence recognition, the model achieves a Word Error Rate (WER) of 0.79% and Sentence Error Rate (SER) of 1.25% . We observe that our multi-distant clustering based body part separation substantially improves the accuracy, and reduces WER and SER. For non-manual marker recognition, separation of body parts results in 7% increase in accuracy (83.5%), in comparison with data without body part separation.

The remaining paper is organized as follows. In Section II we give a brief background and system overview, followed by multi-distant clustering in Section III, multi-view deep learning in Section IV, and evaluation in Section V. Section VI provides the related work. Finally, we conclude in Section VII.

II. Background and System Overview

A. mmWave Radars

Radars utilize the radio frequency waves to determine various spatio-temporal characteristics of an object such as its distance, angle and velocity. mmWave based radars offer better range resolution (due to large available bandwidth), less clutter (high directionality) and high flexibility (can be electronically steered in different directions). With the demand for mmWave based radars increasing, several commercial low cost mmWave based sensors have become commercially available [10], [11]. We now provide a brief primer on how the mmWave radars use FMCW and generate point cloud output.

1) FMCW Signal Processing: FMCW radars (refer Fig. 2) operate by modulating the frequency of the carrier wave linearly within a given range. The signal generated is referred as a "chirp". The operating range of the chirp's bandwidth (B), slope of the chirp (S) gives the rate of modulation, and the total time of modulation between start frequency (f_c) and end frequency (f_e) gives the chirp duration (T_c).

Range and Range Resolution: When the chirp signal is reflected from an object, the range (distance) of the object is given by $R = \frac{c_0 \Delta t}{2}$ where c_0 is the speed of light, Δt is the time delay of the received signal. The range can also be represented by the frequency of the IF signal obtained by calculating an FFT (Range-FFT). With 7 GHz bandwidth (allowed bandwidth in 60 GHz band by FCC), the range resolution of a mmWave radar is 2 cm [12].

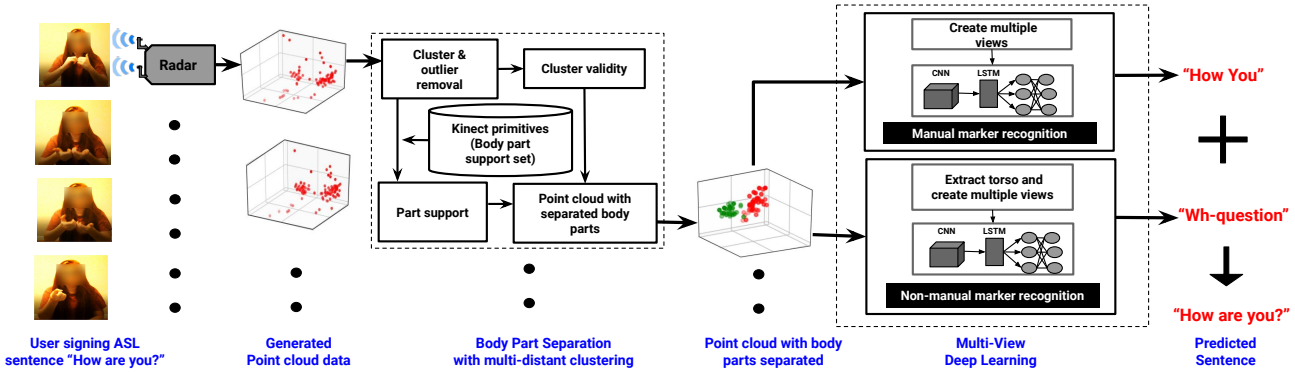


Fig. 1: Overview of ExASL

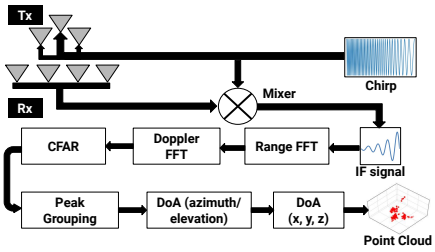


Fig. 2: FMCW radar block diagram

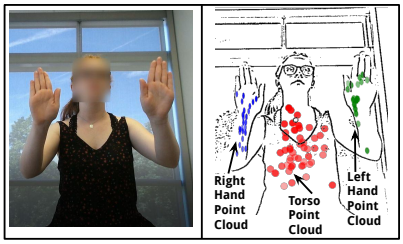


Fig. 3: Figure comparing RGB camera data (left), to ExASL's point cloud data representation (right).

Velocity Estimation: FMCW radars send multiple chirp signals and utilize the phase difference between the received signals to estimate the velocity of a moving object as $v = \frac{\lambda\omega}{4\pi T_c}$. Where λ and ω are the wavelength and the phase of the IF signal, respectively. The phase difference is obtained by computing another FFT (Doppler-FFT) on the previously obtained Range-FFT.

Angle of arrival: The presence of multiple antennas on the receiver is used to estimate the angle of arrival of the received signal in FMCW radars. The difference in phase ($\Delta\phi$) between two antennas is given by $\frac{2\pi\Delta d}{\lambda}$ where Δd is the additional distance traveled because of the distance between the antennas. Hence, the angle of arrival is given by $\theta = \sin^{-1}(\frac{\lambda\Delta\phi}{2\pi l})$ where l is the distance between the antennas.

2) Point Cloud Estimation: As shown in Fig. 2, the Tx sends a frame with N chirps. Upon receiving the frames, the Rx computes the range using Range-FFT and performs Doppler estimation using Doppler-FFT. A Constant False Alarm Rate (CFAR) threshold is applied on the output of Doppler-FFT for object detection. Finally, direction of arrival is estimated. This results in the azimuth and elevation values, and combining the angle with the range, we can calculate the x, y, and z coordinates for each detected object. When ExASL transmits, the transmitted signals are incident upon multiple locations of an object (e.g., different body parts). and each of these reflected signals will be detected as a distinct object (with x, y, z coordinates). All detected objects together result in a "point cloud" as shown in Fig. 3.

B. Non-manual markers in ASL

The non-manual markers are expressed through head movements (shake, tilt, nod, etc.), shoulder movements, torso shifts and even facial expressions [13]. These markers are essential in terms of automatic sign recognition due to their ability to change the underlying meaning of a sentence. In the following, we describe six important non-manual markers that are considered in this paper.

1) Yes-No questions: These are questions which are answered with either Yes or No. This non-manual marker is indicated through shift of the torso towards the addressee [13]. As an example, the statement `"Father became angry"` becomes a question `"Did Father become angry?"` when accompanied with the Yes-No marker.
2) Wh questions: These include question sentences with wh-words such as `Where, Who, When, How, etc.` This non-manual marker is expressed through shift of torso forward towards the addressee along with a head tilt [13]. As an example, the ASL signs `"How you"` with this marker becomes the question `"How are you?"`.
3) Negation: Negation is indicated using side-to-side head shake and frowning. Side-to-side head shake is a common non-verbal indication to imply "NO" [13]. Negation changes the meaning of a sentence with its presence. For example, the signs `"Me feel good"` followed by negation in the end means `"No, I don't feel good"`.
4) Assertion: The assertion marker is expressed through head nodding. Several types of head nods have been identified in ASL including rapid slight head nods, fast head nods, and a larger, deeper, slower head movement [14]. The ASL signs `"Me worried"` with a head nod would translate to `"I am definitely worried"`.
5) Verb inflection: In ASL, same verb can take different meanings when the sign is made with spatio/temporal variations. For example, the sign for verb `want` is to pull the left and right hands towards torso with a grabbing gesture. The same sign when performed with a torso shift

away from the addressee means `really wish`.

6) Spatial agreement: Spatial agreements are used to identify multiple subjects or objects when a single sentence has more than one subject or object. In the sentence `"My father has two brothers and one sister"`, the sign for two is performed with a torso shift towards the left and sign for one is performed with a torso shift towards the right.

C. System Overview

Figure 1 shows the overview of ExASL, which is comprised of three components, mmWave radar for point cloud estimation, multi-distant clustering for body part separation, and multi-view deep learning for ASL sentence recognition.

a) mmWave radar point cloud generation: ExASL utilizes COTS radar from Texas Instruments [10]. The radar has 3 Tx antennas and 4 Rx antennas, with 4 GHz (76 to 81 GHz) continuous bandwidth. The Tx antenna configuration enables the radar to detect range in 3D (x,y, and z axis range values). The 3 dB beamwidth in the azimuth plane and the elevation plane are $\pm 28°$ and $\pm 14°$ respectively. The signals received by the radar is processed on board through the point cloud estimation pipeline discussed in Section II-A1. We use a sliding window (window size of $150ms$ and time step of $50ms$) to buffer and output a continuous stream of frames.

b) Multi-distant clustering: The generated point cloud data is input to the multi-distant clustering algorithm, which separates and associates different point clouds to three considered body parts (Left hand, Right hand, and Torso). The algorithm utilizes a body part support set built using Kinect, which is comprised of primitive motions (collected only once offline), that model the probable locations for different body parts (when ASL signs are performed). The algorithm starts by clustering the obtained point cloud data. For each resulting cluster (with outliers removed), cluster validity and part support are computed in the next step. Cluster validity is computed by establishing cluster similarity with other clusters. Finally, the cluster validity and part support are used in the association of the point clouds to their corresponding body parts.

c) Multi-view deep learning: The multi-view deep learning algorithm is comprised of two components, one for ASL manual marker recognition and the other for ASL non-manual marker recognition. The manual marker recognition takes the body part separated data as input and creates 3 views (xy, yz, and xz axes of 3D euclidean space) for each body part. Convolutional Neural Networks (CNN) are used for learning feature representations from the created views, and Long Short-term Memory (LSTM) units are used for modeling the signs over time. The non-manual marker recognition, operates in a similar fashion with just the torso body part views. ExASL combines the output of the two components in recognizing the signed ASL sentence (Figure 1).

III. Body part separation using Multi-distant clustering

The raw point cloud data generated by our mmWave radar (after pre-processing) for a single frame is shown in Figure 4a. As we can observe, the data provides little indication on which point cloud corresponds to which body part. Also, as pointed in the figure, the data contains second order reflections. These reflections are inherent to RF systems and they are introduced when the signals reflecting of an object goes through another reflection (from a wall or another object). Hence, before we pass the data to the deep learning models for recognizing ASL sentences, we have two challenges to address: (i) separate and associate different point clouds to the body parts, and (ii) remove the second order reflections. To address these challenges we develop a multi-distant clustering algorithm which performs clustering with multiple distances and uses the support of pre-collected Kinect templates, to associate the obtained clusters to their corresponding body parts.

A. Primitive Motions for Body Part Association

In order to associate point clouds to body parts, we calculate the likelihood of each point belonging to a specific body part cluster. This likelihood can be based on distance from the point clouds to the "probable" locations of different body parts. The obtained distance can be strongly correlated to the likelihood of association (i.e., lower the distance, higher the likelihood of association). The probable locations can be estimated based on the fact that when a user performs ASL signs, there are a few distinct locations within which the hands can move (termed as major and minor locations of ASL signs). This is because most ASL signs are some combinations of primitive hand movements (such as push, pull, hands up, etc.). Based on this, we identify a 6 primitive motions: (i) extending both hands forward, (ii) extending both hands vertically up, (iii) extending both hands to the left, (iv) extending both hands to the right, (v) extending left hand to left and right to the right simultaneously, and (vi) lift both the hands from bottom to face. These primitive motions once modeled, can be used as points of reference for body part locations during ASL signing. These reference points can be compared to the obtained point clouds in determining there association to a body part.

We use Kinect to create the templates for the primitive motions. The templates are essentially the locations of the body parts as a user performs the primitive motions. There are various advantages of using Kinect for this purpose. Kinect can provide an accurate estimate of the body part locations with less noise. In addition, the data provided from Kinect is the 3D coordinate (x, y, and z) of each joint, matching the point cloud data representation. This enables a direct comparison between the point cloud data and Kinect data. Kinect provides values for 24 joint locations, of which we use the joint locations of left hand, left wrist and left hand tip (the tip of index finger) to model the probable locations of left hand. Joint locations of right hand, right wrist and right hand tip are used to model the probable locations of right hand and joint locations of head, spine and neck are used to model the probable locations for the torso. Fig. 4d shows a snapshot of the primitive motion extending both the hands vertically up for Kinect. *We note that this collection of primitive motion is done only once using a single user. Once collected these primitive motions are used*

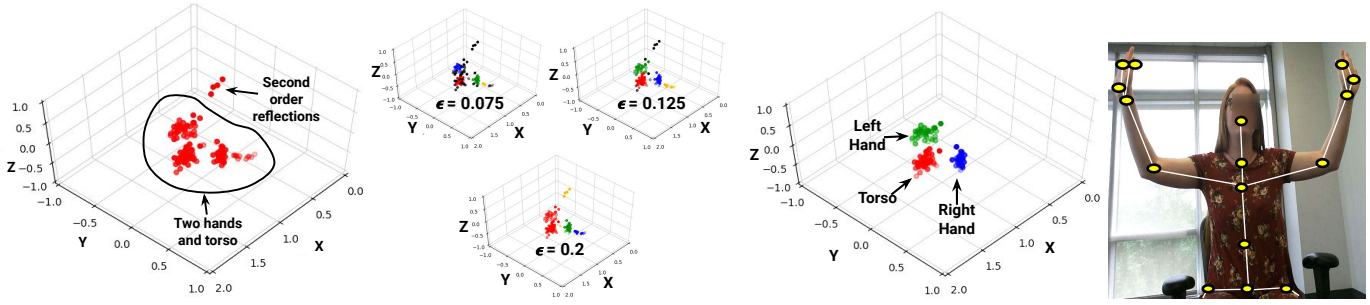


Fig. 4: (a) A single point cloud frame without body part separation (b) body part separation using clustering with fixed distance ϵ (c) body part separation using Kinect support set and multi-distant clustering, and (d) joint information provided by Kinect while person performing a primitive motion

as a template in the proposed algorithm for determining point cloud association.

B. Algorithm for Point Cloud Association

Algorithm 1 Multi-distant clustering

Input: i) Point cloud data X generated by ExASL at time t_i. ii) list of distances D iii) Threshold γ to determine cluster overlap iv) Threshold Δ to determine cluster validation v) Set of primitive motions P_m vi) List of body parts B
Output: Body part clusters (BPC) for the three body parts
Procedure: $clusters \leftarrow getClusters(X, D)$
▷ Compute V and PS for each cluster
for $d \in D$ **do**
  $C \leftarrow$ clusters of distance d
  $C' \leftarrow$ clusters with distance other than d
  for $c \in C, c' \in C'$ **do**
    if *getClusterOverlap(c,c')* $\leq \gamma$ **then**
      $V \leftarrow V + 1$
    end if
  end for
  for $b \in B, p \in Pm$ **do**
    PS[b] $\leftarrow$ *getPartSupport(b,p)*
  end for
end for
▷ Utilize V and PS to determine body part association
for $c \in clusters$ **do**
  if V for c $> \Delta$ **then**
    $bodyPart \leftarrow Min(\{PS[b]\ \forall b \in B\})$
    if $BPC[bodyPart] = \emptyset$ **then**
      $BPC[bodyPart] \leftarrow c$
    else
      $P \leftarrow getNonOverlapping(c, BPC[bodyPart]$
      $BPS[bodyPart]$+=P
    end if
  end if
end for

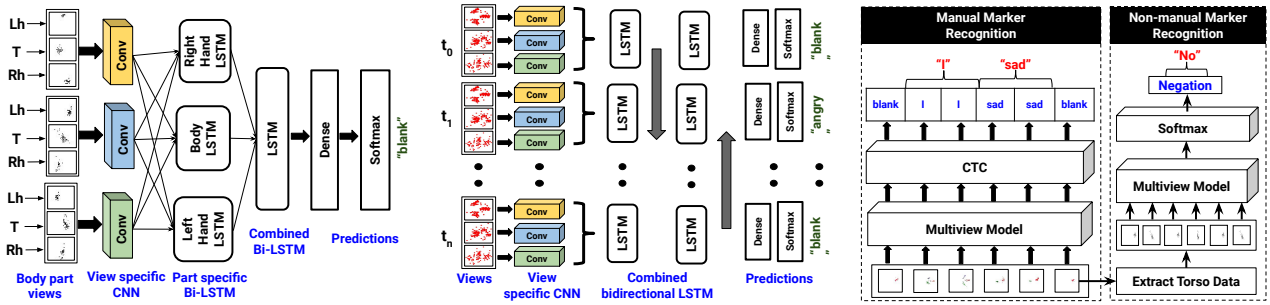
Algorithm 1 gives the pseudo code for the proposed multi-distant clustering algorithm. The algorithm operates in two phases, for each frame, first it computes clusters using DB-SCAN clustering algorithm for a list of possible distances (we choose ϵ values between 0.075 and 0.2), and calculates the *validity (V)* and *part support (PS)* for the obtained clusters. We use multiple distances as different distance values result in different set of clusters each having some advantage over the other (depicted in Fig. 4b). In the second phase, the computed *validity* and *part support* values are used to associate the points to different body parts. We define *validity (V)* of a cluster as the number of clusters that are similar to the cluster for which validity is calculated. Similarity is defined by the overlap between the clusters. We define the overlap for cluster c_1 with respect to cluster c_2 as

$$\text{Overlap}(c_1, c_2) = \frac{|\{\, c \mid \forall c \in c_1\ \exists c' \in c_2\ \wedge \text{Dist}(c,c') < \beta\}|}{|\,c1\,|} \tag{1}$$

where here c_1 and c_2 are set of points, Dist represents the distance and $\beta = 0.00001$. Let P_m be the set of primitive motions, d be the cluster centroid of the cluster c_1, then the *part support (PS)* for the cluster c_1 with respect to body part b is defined as

$$PS[b] = \sum_{p \in P_m} \text{getNearestDistance}(p, d, b, N) \tag{2}$$

where N defines the number of nearest points to the cluster centroid which we set as 30 based on empirical observations. Once we obtain the *validity* and *part support* for all clusters, we associate the points to different body parts in the next phase. We only choose a cluster if it has a *validity* greater than Δ (where $\Delta = 2$), and the body part associated with the cluster is the one with the minimum *part support*. If the corresponding body part already has points in it, we add the non-overlapping points from the current cluster c_1 with the existing points for that body part. We define the set of non-overlapping points P in c_1 with respect to c_2 as $P = \{\, point \mid point \in c_1 \wedge point \notin S\}$ where S is the bounding sphere that encloses all the points in c_2. We compute the sphere using Ritter's bounding sphere algorithm. Figure 4c shows the output of multi-distant clustering on a single frame when a user is performing an ASL sign (Figure 4a). In Figure 4a, the second order reflections are visible, while in Figure 4c, the second order reflections have been removed and each body has been distinctly identified. Compared to Fig. 4b which uses a fixed distance to clustering, multi-distant clustering performs much better in correctly separating body parts. As we show in the



(a) `Clust-MV` (b) `Unclust-MV` (c) ASL sentence `"I am not sad"` recognition

Fig. 5: (a-b) Proposed variants of the multi-view deep learning models, and (c) ASL sentence recognition in ExASL.

evaluation section, body part separation through multi-distant clustering is effective in improving the ASL sentence and sign recognition.

IV. Multi-view Deep Learning

With the separation of body parts, ExASL produces a 3D point cloud representation of the three body parts: left hand, right hand, and torso. When the user performs a gesture, a sequence of point cloud frames are generated. Given this sequence of frames, ExASL has to recognize the corresponding ASL manual signs as well as the non-manual markers. Signing of an ASL sentence involves variation of the separated body parts over time, and the ability track these variations is key to accurate recognition. Also, the point cloud data generated by ExASL is sparse in nature (refer Fig. 4c), where a visual recognition of the underlying body part is difficult.

To address these challenges, we employ a multi-view CNN model and extend it with Bidirectional Long Short-Term Memory (LSTM) units to model the body part interactions over time from sparse point cloud representations. In Multi-view CNNs, multiple views of the 3D point cloud data are generated, with each view being a 2D snapshot of the point cloud data from a particular view point (e.g., xy axis). LSTM's possess multiple gates (termed as input, output, forget and update gates) along with a cell state to keep track of information from the previous time steps and to gather information (required) from the input at current time step. Because of their power to capture long term and short term dependencies over time, LSTM's have become the state of the art models for time sequence modeling. Bidirectional LSTM is a variant of the LSTM which processes the input in two directions, one from start to end (from time t_0 to t_n) and another from end to start (from time t_n to t_0).

A. Multi-view Deep Learning Models

ExASL utilizes 3 views (xy, yz, and xz) to represent the point cloud data. We propose three different models, each take the images of these 3 views to recognize the ASL sentence.

1) **Clustered Multi-view (Clust-MV):** At each time step (shown in Fig. 5a), this model takes 9 images as input where each image represents a view of a body part (3 body parts $\times$ 3 views = 9 Images). The model utilizes separte 2D CNNs for each view for learning feature representations. The learned features for each body part is separated and modeled over time using separate bidirectional LSTMs. The output of these LSTMs are then concatenated and passed through another bidirectional LSTM to model the interaction between the parts. This is followed by dense and softmax layers for prediction.
2) **Clustered Multi-view Swapped (Clust-MV-SWP):** The model improves over the previous model `Clust-MV` with a simple data augmentation technique. For every input data sample, it takes another copy of the sample with right and left hands swapped, along with the original sample. The intuition behind the swapping is based on the symmetry in the human skeletal structure. This data augmentation doubles the training sample size and also leads to better performance (as quantified in evaluation section).
3) **Unclustered Multi-view (Unclust-MV):** Fig. 5b shows the model for multiple time steps. The model takes 3 images at every time step, which are the three views of the unprocessed point cloud. Similar to `Clust-MV`, it uses separate 2D CNNs for learning feature representation of each view. The learned features are concatenated and input to a bidirectional LSTM for modeling the time variations. Finally, the output of the bidirectional LSTMs are passed through dense and softmax layers for prediction.

For all the discussed models, each convolutional layer is made of convolutional kernels of size 5×5, followed by max pooling layer (for down-sampling) with kernel size 2×2 and rectified linear units (for non-linearity). View specific CNNs are employed with 4 such convolutional layers, each with 16, 32, 64, and 128 filters respectively. All the bidirectional LSTMs contain two layers of LSTM cells each with 2048 hidden units. The dense layer consists of three linear layers each with 2048, 1024, and 512 hidden units, respectively. All the linear layers use rectified linear units for activation. A dropout layer is present between the last two linear layers with a drop out rate of 0.65 for regularization. The same models are used for both word level and sentence level classification.

B. Sentence and Non-manual Marker Recognition

The network output at each time step is used to compute the Connectionist Temporal Classification (CTC) loss [15] with respect to the target sequence (i.e., sentence). CTC loss is used for training the network. CTC enables direct modeling of the alignment between the input sequence (frames) and target sequence (sentences) without the need for segmentation or frame level labeling. To use CTC, `"blank"` is added as

a class to the existing classes. The reason for this is shown in Figure 5c. Here, the model is trying to align a sequence of 6 frames to a sentence of two words "I Sad". As pointed out earlier, CTC predicts on every frame, but there could be frames which need not be part of any label (first and last frame in Figure 5c). CTC overcomes this issue, by introducing the blank class to the set of classes. After generating per frame label, a decoding algorithm is used to get the actual labels from the given sequence. Algorithms such as the best path and beam search are commonly used decoding algorithms. ExASL utilizes the best path algorithm, which given a sequence of outputs as seen in Figure 5c will remove the blanks and replace the multiple continuous occurrence of a class with a single occurrence. So, the final output for the example would be "I sad", meaning "I am sad".

Since the non-manual markers are recognized through head and torso movements (jointly referred as torso herein), only Unclust-MV model is used, both for the unprocessed data as well as the body part separated data. In the former case, the input is the 3 unprocessed views. For the body part separated data, the 3 views corresponding to the torso body part are input to the model. A None class is included to label the sentences without any non-manual markers. Figure 5c shows the non-manual marker recognition process. For the same set of frames, the model takes only the torso body part as input and predicts Negation as the non-manual marker. Combining this with the prediction of sentence recognition model would change the meaning to "I am not sad". Lastly, we also develop a model for performing only word-level recognition for comparison in evaluation. For the word level, the output from the final time step (for all the models) is used for prediction.

V. Evaluation

A. Dataset Collection and Implementation

1) Participants and ASL Data Collection: We pick 29 ASL sentences for evaluation, of which 16 sentences have non-manual markers and 13 do not have non-manual marker. Of the 13 sentences without non-manual markers, 9 of them have counterparts with non-manual marker. For example, the sentence "Fire that" with a torso forward shift (Yes/No-question) indicates the question "Is that fire?". The counter part of this sentence without the torso forward shift, is the statement "That is fire." is also included as part of the dataset. Without these counterparts, the deep learning model could learn the difference in sentences to label the non-manual markers, instead of learning the underlying non-manual markers (torso/head movements). Table I gives the list of the 23 words and 6 non-manual markers for the chosen sentences. We include the class blank to the set of words for performing sentence level recognition using CTC as explained in Section IV-B. None is used as the label for sentences without non-manual markers, both while training and testing for non-manual marker recognition.

For evaluating the performance of ExASL in sentence level ASL recognition, we collect samples from 5 participants (IRB approved). For each participant, we collect 20 instances per sentence resulting in 2900 samples in total. We also collect samples on the list of words (refer Table I), to evaluate the performance of ExASL in isolated ASL sign (word level) recognition. As the major focus is to showcase the performance of ExASL in adding context to the ASL sentences with non-manual markers, we only collect 10 instances per word from 4 participants. The data was collected with the participant seated in front of the sensor (for both words and sentences). While the words were 2 to 3 seconds long, sentences were 5 to 7 seconds long, and the total collection time for all participants was between two to three weeks. We recruited participants for the study with varying experience in ASL usage (beginner to advanced).

Words	I, Want, Piano, Wake up, Me, Weather, Teach, Angry, Worried, Never, That, Mine, Books, One, How, Two, They, Visiting, Students, You, Fire, Time, and, There.
Non-manual markers	Wh/Yes or No-question (Torso forward shift), Verb inflection (Torso backward shift), Spatial agreement (Torso side shift) , Assertion (Head nod), Negation (Head shake), and None.
Word level data	4 Participants, 23 words, 10 instances 4 x 23 x 10 = 920 Samples
Sentence Level data	5 Participants, 29 sentences, 20 instances 5 x 29 x 20 = 2900 Samples

TABLE I: List of words, non-manual markers, and data collected for word , sentence, and non-manual marker recognition evaluation of ExASL.

2) Implementation: The data collected is processed for body part separation and the required views are created on a separate Linux desktop (offline), before being input to deep learning algorithms for training/testing. We implement the deep learning models on computing clusters with NVIDIA Tesla K480 GPU's. The models were implemented using Pytorch. We use 80% of data for training and the remaining 20% data for testing. The word level models took approximately 14 hours to train, while the sentence level models took approximately 2 days to train.

3) Evaluation metrics: We use accuracy as the metric of evaluation for word recognition and non-manual marker recognition. For sentence recognition, we use Word Error Rate (WER) and Sentence Error Rate (SER) as the metrics of evalution. WER is a standard metric used in automatic speech recognition systems, and machine translations. WER is defined in comparison between the target sequence (sentence – sequence of words in our case) and the predicted sequence as $WER = \frac{S+D+I}{N}$ where S is the number of substitutions, D is the number of deletions, I is the number of insertions and N is the number of words in the target sequence. WER (derived from Levenshtein distance), accounts for the number of substitutions, insertions, and deletions required in the target sequence to recover the predicted sequence. For example, if the target sequence (ground truth) is "I want apple" and the predicted sequence is "I apple", then one deletion is required in the target sequence to recover the predicted sequence. Thus, the WER would be $\frac{1}{3} = 33.33\%$, with 1

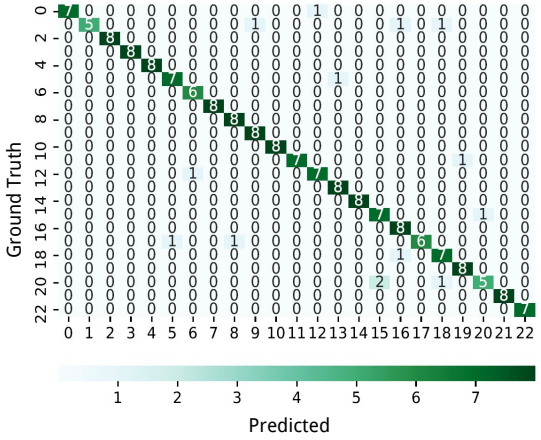


Fig. 6: Confusion matrix for word level recognition using `Clust-MV-SWP`

deletion (D) and 3 words (N) in the target sequence. When computed for a collection of sentences, WER is calculated as the ratio of total number required insertions, deletions, and substitutions to the total number of words in all the sentences. SER is defined as the ratio between the number of incorrect sentence predictions, to the total number of predictions.

B. Numerical Results

1) Word level: For evaluating the word level recognition of ExASL, we compare the proposed multi-view deep learning models (refer Section IV-A) by training with 80% of user data and testing with remaining 20%. From the results

Model	Accuracy
`Unclust-MV`	89.5%
`Clust-MV`	91.4%
`Clust-MV-SWP`	**92.5%**

TABLE II: World level recognition accuracy

shown in Table II, it is evident that, all the three models offer comparable performance. `Clust-MV-SWP` model performs the best with a 3% increase in accuracy, compared to `Unclust-MV` model. Figure 6 shows the confusion matrix for `Clust-MV-SWP` model. We observe that, a common source of confusion is between the words that involve similar motion. For example, `You` involves moving a single hand towards the addressee, while `How` involves moving both the hands jointly towards the addressee with a minor downward movement in the end. Because of the sparse nature of the obtained point cloud, the difference between moving the hands jointly (for `How`)) and moving a single hand (for `You`) is not substantial.

2) Sentence Level: We evaluate ExASL's performance in sentence level recognition, in a set up similar to word level evaluation (80% train and 20% test) with the 3 multi-view deep learning models. Figure 7 shows the performance of the three models. `Clust-MV-SWP` offers the lowest WER (Figure 7a) of 0.79% which is 1.6% lower than `Unclust-MV` model. Even without data augmentation, `Clust-MV` results in 1.37% decrease in WER which shows the significance of multi-distant clustering and explicit modeling of body part interactions. When evaluated on SER (Figure 7b), `Clust-MV-SWP` reduces the error rate by 2.23% (1.25%), compared to the `Unclust-MV` (3.48%) model which does not take advantage of the body part separation.

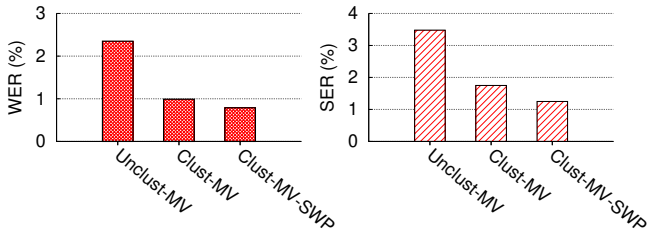


Fig. 7: Results for ASL sentence recognition (a) Word Error Rate (WER) and (b) Sentence Error Rate (SER) for the three Multi-view deep learning models

a) Cross subject evaluation: We train the models with 4 participants data, and test on the remaining one participant, to evaluate ExASL's ability to adapt to new users. The resulting WER's are 33.68%, 28.35%, and 24.11% for `Unclust-MV`, `Clust-MV` , and `Clust-MV-SWP` respectively. For SER, the results are 46.15%, 42.24%, and 39.14% for `Unclust-MV`, `Clust-MV` , and `Clust-MV-SWP` respectively. `Clust-MV-SWP` model taking advantage of the body part separation, achieves 9.57% and 7.01% decrease in WER and SER respectively. While the results reestablish the significance of the proposed models, we see an increase in WER and SER in cross subject evaluation in comparison to training and testing on same participants. We believe the problem to be the lack of participant diversity in our current dataset. As statistical learning (based on which deep learning models are built) is heavily based on data availability, adding data from more participants (leading to more diversity) can alleviate this problem. We also observe that, words that were confused in word level recognition (because of similar movement), are also confused during sentence level recognition.

Model	Accuracy
`Unclust-MV` (Trained and tested on data without body part separation)	76.5%
`Unclust-MV` (Trained and tested on Torso body part data alone)	**83.5%**

TABLE III: Non-manual recognition accuracy with and without body part separation

3) Non-manual markers: Table III shows the results for non-manual marker recognition when trained on 80% participant data and tested on remaining data. As explained in Section IV-B, we only use `Unclust-MV` model for non-manual marker recognition, as the only body part required for training is the torso. `Unclust-MV` performs better when input with just the torso data (obtained with multi-distant clustering), compared to input without body part separation. There is a 7% increase in accuracy in the former case. Figure 8 shows the confusion matrix for non-manual marker recognition. The major confusion happens between `Assertion` (Head nod), `Negation` (Head shake) and `None`. Compared to torso, head has smaller cross section, resulting in even sparser point clouds. This makes it more difficult for ExASL to differentiate

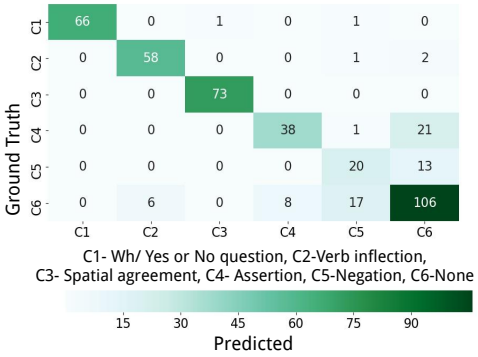


Fig. 8: Confusion matrix for non-manual markers using `Unclust-MV` with torso body part data

between these non-manual markers and `None` (participant not performing any non-manual marker). Even in the absence of non-manual marker, there is some head movement when a participant does any ASL sentence, adding to the confusion.

VI. Related Work

Initial works that utilized RGB camera based systems for ASL recognition where based on Hidden Markov Models (HMM) [2], [3] with hand crafted features. With the advent of Deep learning algorithms, recent works have adopted them for ASL recognition from continous videos. In [4] authors propose a hierarchical attention network with latent space, to do continuous sign language recognition without segmentation. A new optimization technique, that combines Dynamic Time Warping (DTW) alignment constraint with maximum likelihood constraint was proposed in [16], for ASL recognition from videos. In contrast to ASL manual-marker (hand signs) recognition, non-manual marker recognition, which is the focus of our work, has received little attention. In [8], authors propose a framework for face tracking and position estimation, which they use in classifying two non-manual markers (Wh-question and negation). In [9], authors utilize a 2-level Conditional Random Fields (CRF) to track the eyebrow and head gestures over time, to recognize five non-manual markers. Authors in [17] propose a adaptive face tracking methodology to recognize facial expression from continuous video, which they use for non-manual marker recognition. Unlike the existing works that focus on eye brow and head gesture non-manual markers, our work studies non-manual markers that involve movements in torso and head.

Because of the direct availability of human joint data, Kinect and Leap Motion (infrared based) has been extensively studied for ASL sign recognition [6], [18]. Wearable IMUs [5] have also been used for ASL sign recognition. In [7] authors have utilized WiFi channel state information (CSI) for ASL sign recognition. While Kinect has the same limitations as camera based systems, wearables require on body presence, while ExASL offers a device free solution. No existing work in other modalities have studied non-manual marker recognition in ASL, which is the focus of our work. The presence of higher bandwidth and directional antennas has led to the recent wave of mmWave sensing. In [19], authors exploit the directional nature of mmWave systems for human identification and human vital sign monitoring. Authors in [12] propose a new system for short range gesture recognition using mmWave. Compared to [12], our focus is on ASL hand and body gesture recognition at larger distances where body part separation is necessary for accurate recognition.

VII. Conclusion

In this work, we presented a 60GHz mmWave based ASL recognition system, for ASL sentence recognition with manual and non-manual markers. We proposed a multidistant clustering algorithm, which utilized Kinect primitives to separate different body parts from the generated point cloud data. We developed a multi-view deep learning algorithm, that took advantage of the separated body parts in both manual and non-manual marker recognition. We extensively evaluated ExASL's performance in sentence-level ASL recognition.

References

[1] B. B. Blanchfield, J. J. Feldman, J. L. Dunbar, and E. N. Gardner, "The severely to profoundly hearing-impaired population in the United States: prevalence estimates and demographics." *Journal of the American Academy of Audiology*, vol. 12, no. 4, pp. 183–9, 2001.

[2] T. Starner, J. Weaver, and A. Pentland, "Real-time asl recognition using desk and wearable computer based video," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1998.

[3] K. Grobel and M. Assan, "Isolated sign language recognition using hidden markov models," in *IEEE International Conference on Systems, Man, and Cybernetics*, 1997.

[4] J. Huang, W. Zhou, Q. Zhang, H. Li, and W. Li, "Video-based sign language recognition without temporal segmentation."

[5] J. Hou, X.-Y. Li, P. Zhu, Z. Wang, Y. Wang, J. Qian, and P. Yang, "Signspeaker: A real-time, high-precision smartwatch-based sign language translator," in *Mobicom 2019*.

[6] B. Fang, J. Co, and M. Zhang, "Deepasl: Ubiquitous and non-intrusive word & sentence-level sign language translation," in *SenSys '17*.

[7] Y. Ma, G. Zhou, S. Wang, H. Zhao, and W. Jung, "Signfi: Sign language recognition using wifi," *ACM IMWUT 2018*.

[8] N. Michael, D. Metaxas, and C. Neidle, "Spatial and temporal pyramids for grammatical expression recognition of asl," in *ACM Assets '09*.

[9] J. Liu, B. Liu, S. Zhang, F. Yang, P. Yang, D. N. Metaxas, and C. Neidle, "Recognizing eyebrow and periodic head gestures using crfs for non-manual grammatical marker detection in asl," in *IEEE FG 2013*.

[10] [Online]. Available: http://www.ti.com/sensors/mmwave/overview.html

[11] [Online]. Available: https://www.siversima.com/products/radar-sensors/

[12] J. Lien, N. Gillian, M. E. Karagozler, P. Amihood, C. Schwesig, E. Olson, H. Raja, and I. Poupyrev, "Soli: Ubiquitous gesture sensing with millimeter wave radar," *ACM Trans. Graph.*, 2016.

[13] C. Baker and C. Padden, "Focusing on the nonmanual components of american sign language," *Understanding language through sign language research*, pp. 27–57, 1978.

[14] R. B. Wilbur, "Nonmanuals in american sign language," *The Signs of Language Revisited: An Anthology To Honor Ursula Bellugi and Edward Klima*, p. 190, 2013.

[15] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks," in *ACM ICML '06*.

[16] J. Pu, W. Zhou, and H. Li, "Iterative alignment network for continuous sign language recognition," in *IEEE CVPR 2019*.

[17] D. Metaxas, B. Liu, F. Yang, P. Yang, N. Michael, and C. Neidle, "Recognition of nonmanual markers in American sign language (ASL) using non-parametric adaptive 2D-3D face tracking," in *LREC 2012*.

[18] A. Agarwal and M. K. Thakur, "Sign language recognition using microsoft kinect," in *2013 Sixth International Conference on Contemporary Computing (IC3)*, Aug 2013, pp. 181–185.

[19] Z. Yang, P. H. Pathak, Y. Zeng, X. Liran, and P. Mohapatra, "Monitoring vital signs using millimeter wave," in *ACM MobiHoc '16*.