Signed English Recognition with Spatio-Temporal Graphs and Language Modelling

Shashwath Kumar Santhosh
Dept. of CSE
PES University
Bengaluru, India
shashwath457@gmail.com

Sushanth R Kayshap
Dept. of CSE
PES University
Bengaluru, India
sushanthkashyap12@gmail.com

Shloka Reddy Lakka
Dept. of CSE
PES University
Bengaluru, India
lakkashloka@gmail.com

Arti Arya
Dept. of CSE
PES University
Bengaluru, India
artiarya@pes.edu

***Abstract*—Sign Languages are a unique family of languages that use visual cues rather than verbal means for conveying meaning and a mode for communication. They are extensively used by the deaf and dumb community. Sign language interpretation is a skill with a steep learning curve and is quite essential to understand for accessibility concerns. But with the current advancements in computer vision, recognizing human action and translating sign language into text is possible. This paper proposes a system that uses a combination of MediaPipe holistic (for human pose estimation), graph neural networks, and the Electra transformer to interpret and classify the 3D skeletal data of the signed word. The proposed model predicts the word signed using Graph Convolutional Network (GCN) with Natural Language Processing (NLP) to utilize the context for the word being gestured. Using this approach the model achieves around 97% accuracy on words signed based on its context which is comparable to State-of-the-art models.**

***Index Terms*—GCN, sign language, Computer Vision, NLP score**

I. Introduction

Sign Language remains to be the primary means of communication in the deaf and dumb communities despite the accessibility and ease of use of alternative methods such as texting. Being visual, quick, and accessible, sign language forms a bedrock for a whole class of gesture dialects.

But when it comes to automated Sign Language Recognition(SLR), the technology is yet to take off on consumer devices. Akin to how voice to text is ubiquitous now. Several factors such as poor cameras, low processing power, high compute techniques, and reliance on specialized capture hardware had prevented consumer adaptation. But there have been several attempts at such systems being made with the current explosion in consumer hardware specifications that ameliorate a part of the negative components.

Sign Language Recognition in and of itself is a specialized case of human action recognition. It involves being able to recognize and differentiate sequences of different orientations of the human skeletal structure while also being invariant to external factors such as camera angle, body proportions, lighting, clothing, subject environment, etc., and there have

The main parameters to be considered in an SLR task are the hand signs, palm orientation, body pose concerning the hands, and facial expression. And in the last couple of decades, various techniques such as Support Vector Machines (SVM) [3], Hidden Markov Models (HMM) [14], Convolutional Neural Networks (CNN) [13], Long Short-Term Memory (LSTM) [5] and Graph Convolutional Networks (GCN) [9] have been applied to this problem. And these techniques can be broadly grouped into 2 categories. One being the methods that perform end-to-end deep learning recognition on the video feed itself. And the other being those that capture essential body orientation/landmark data upon which they let their models run inference on.

In this paper, the latter approach is undertaken. Recent advances in computer vision and deep learning allow for the availability of open-source pose estimation models that provide approximate landmarks from RGB cameras in real-time. Despite their ease of use and setup, data from these streams are generally riddled with noise from spurious recognition. The model constructed must be robust to these fluctuations.

In spite of the fact that recognition accuracy of individual words has increased over the years, their Top-5 accuracy (the 5 most probable words predicted by the model) is usually dramatically higher than their Top-1 accuracy. In order to exploit this, there needs to be a way to pick the right word from the Top-5 predictions.

As all sign languages have a basic grammar structure, one should be able to pick out the right word by utilizing the context of the preceding sentence. This research adds the dimension of context in a language to this problem.

II. Related Works

The task of automated sign language recognition has benefited from decades of research and the solutions reflect the times in which they were proposed. The methods range from statistical inference over motion capture data, all the way to end-to-end deep learning solutions from video feed. The end-to-end method offers a single model solution but the complexity of the task required of the model is tremendous. It needs to understand pixel data and extract human features and orientation across time. Along with being able to recognize patterns of movement within those features. The motion capture approach mentioned previously is less taxing on the recognition model since the essential body features are already extracted, so it should have an easier time tracking this data on account of dimensionality reduction.

Liu et al. [6] have proposed an LSTM model using a Microsoft Kinect 2.0 to extract skeletal features. They extract 4 points: two hand joints, and two elbow joints. They build two datasets based on the Chinese sign language: dataset 1 contains 50 subjects, 100 signs with 5 repetitions each, and dataset 2 contains 50 subjects, 500 signs with 5 repetitions each. They get an accuracy of 86% on dataset 1 and 63% on dataset 2.

Anshul Mittal et al. [7] have used a leap motion sensor to extract hand features. They extracted 12 points from the hands and represented them in 3D space. The points included were ten fingertip points and two points from the middle of each palm. Their proposed model is a modified LSTM which has a reset gate and gets the input from a CNN. Their dataset consists of 6 subjects, 35 words signed with 15 repetitions each. They get an accuracy of around 89% for isolated signs and 72% for sign sentences.

Koller et al. [8] propose a model architecture where they combine CNN with HMM for the task of Sign Language Recognition. CNNs have strong discriminative properties but are inadequate for sequential modelling. However, combining them with HMMs, the authors were able to exploit the sequential nature of continuous SLR tasks while also being an end-to-end model for added simplicity. They achieved 13% improvement over the state-of-the-art (SOTA) at the time of its publication.

De Amorim et al. [9] apply a recent technique from Graph Convolution Networks, i.e, multiscale spatial-temporal graph convolution operator (MS-G3D) on a skeletal graph for the task of Isolated Sign Language Recognition. They compare the results with those achieved by using 3D-CNNs and find their proposed approach to perform much better. They also find only a small increment in the global accuracy when combined with 3D-CNN in an ensemble. Hence, they conclude that GCNs with MS-G3D perform well enough to be used in isolation. The benchmarks were obtained over the AUTSL dataset [24].

Cheng et al. [10] propose a novel shift graph convolutional network (Shift-GCN) to get around two main drawbacks that plague previous GCN approaches. They tackle the problem of high-cost computation as well as attempt to rectify the inflexible nature of the spatio-temporal graph. They test their results using the NTU-120 RGB+D dataset [28] and the Northwestern-UCLA dataset [29] and note that they get better accuracy than previous state of the art models on both datasets while incurring a computation cost that is 10 times lower.

Jiang et al. [11] propose a Skeletal aware multi-modal SLR framework (SAM-SLR) where they combine and assemble a varied set of models. The skeletal aware section of these are the GCNs made of Sign Language Graph Convolution Network (SL-GCN) and Spatio-Temporal Convolution Network (SSTCN). These features are combined with an ensemble of CNNs, one per each of the RGB and Depth modality channels. Their purpose is to complement the features extracted by the GCNs. With this approach, the authors were able to achieve state-of-the-art results on the AUTSL dataset [24] that is based on the Turkish Sign Language.

Most of the SLR research discussed here has been on improving isolated sign recognition accuracy. Yet, sentence formation, the basis of every language, has not been researched extensively in SLR. This is primarily because the reputed datasets that exist for SLR testing are for word-level sign recognition. The proposed work enhances previous works by including the context of the sentence to improve recognition of the gestured word.

III. Proposed Approach

A brief overview of the proposed approach is presented in this section. To begin, body orientation/landmark is extracted from the video feed and this data is treated as a temporal-variant graph. Thereby using a version of Graph Attention - Convolution Network, the Top-N predictions are inferred from the model and are cross-validated with an NLP model to pick the right prediction within the context of the sentence.

For the NLP model to be effective, it needs a representative corpus that follows the same grammar as the signed language. Signed English follows the same grammatical structure as written English. As English has comprehensive literature for training the NLP model, Signed English is chosen as a proof-of-concept language but the same technique can be applied to any other Signed language.

A. System Design

The system for capturing signs from the user and getting a prediction is given in Fig. 1

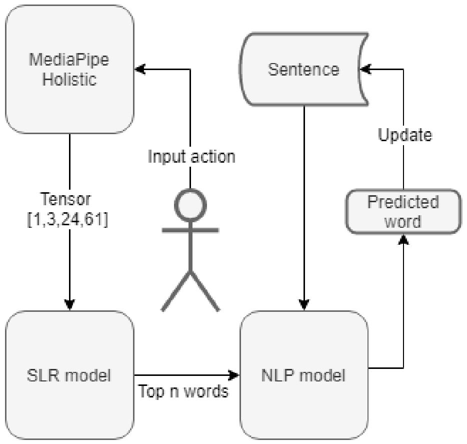

Fig. 1. Sign Language Recognition Pipeline

The pipeline is as follows:

- The person performs a sign which is within the vocabulary set, captured by any video capture device.
- The frames are then passed through MediaPipe holistic for feature extraction. The extracted features are the skeletal data points of the body and each hand.
- The data is pre-processed before it is fed into the GCN model.
- The Top-N words predicted from the model are then fed to an NLP next-word predictor model to check which of the N-words fit best in the context of the sentence.

- The best word is chosen and the sentence is updated for the next sign-word.

B. Dataset

The pipeline has two main sections: GCN and NLP. In order to train them, a custom dataset for each is constructed. The first detail on the dataset is done for the GCN model.

To train the model, a database of signs from Signed English Extract (SEE) is required. This database needs to contain each sign repeated multiple times, over a fixed set of frames, from a varied set of subjects. The dataset was hand-crafted and it consists of 4 subjects, 120 words(action classes), 24 frames per word, 30 repetitions per word per subject. Video feed frames are captured through Open-CV [23], and skeletal features are extracted using MediaPipe holistic. While the dataset is recorded, the actions are processed to exclude extraneous joints in the graph. Each word is stored as a tensor of shape: [Repetition_no, Channel_no, Frame_no, Landmark_no]. During training all the tensors of each word are combined into one tensor and the data is shuffled. The shuffled data is then divided equally into 32 batches with 70-15-15 train-validation-test split.

Algorithm 1 Data collection of one hand sign

```
Using mp_holistic.Holistic as holistic
$result \leftarrow []$
$frame\_num \leftarrow 0$
while $frame\_num < 24$ do
    $frame \leftarrow readImageFromWebcam()$
    $image \leftarrow mediapipe\_detection(frame, holistic)$
    $result \leftarrow result + image$
end while
```

The NLP model is taken from a pre-trained model from HuggingFace [25]. For testing the significance of considering context for SLR, a dataset that contains sentences with masked words is necessary. A labelled dataset of partial sentences for the task of next-word prediction is created. Each entry in the partial sentence dataset has two sections: the preface phrase and the expected word at the end of the phrase. 94 partial sentences were collected where the expected word is drawn from the sign vocabulary set. These partial sentences provide context to the NLP model.

C. Pre-processing and Feature Extraction

The video feed is captured using Open-CV at an approximate frame rate of about 20 Frames Per Second. At every frame, landmark feature extraction is done through an off-the-shelf pose-net model, which is Google's MediaPipe [17] holistic pose-net solution. The holistic solution from MediaPipe outputs four graphs being: Face mesh, 2 Hand graph-net [18] and body pose-net [19]. The face mesh is discarded since the body pose mesh provides enough landmarks to locate the face. Fig 2 shows the above-mentioned graphs in detail.

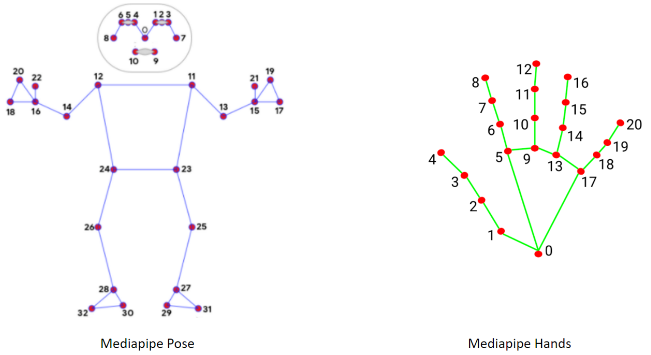

Fig. 2. The features extracted from MediaPipe [21]

Some of the landmarks within these graphs are extraneous or redundant for SLR such as points below the hip and palm markers from pose-net. These points are removed from the graph. The hand graphs and pose-net are then stitched together with an extra edge across the wrist joints to make one seamless graph representing all necessary data points required for SLR. Fig 3 shows the pre-processed data points.

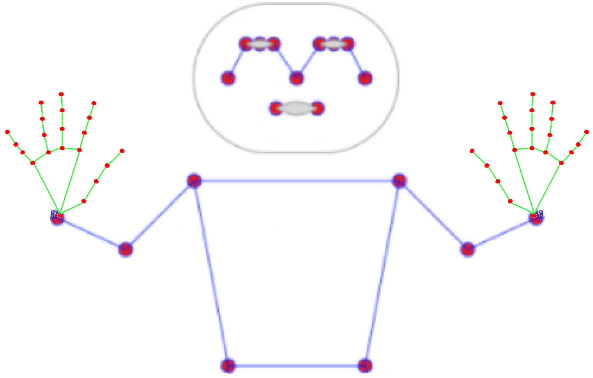

Fig. 3. Essential points extracted

In total 61 nodes/points are captured, 19 from pose-net and 21 from each hand (42 in total). The output of MediaPipe is pre-processed to a tensor of shape [61,3] (61 nodes, 3 channels each that represent x,y,z hip-centered coordinates of the landmarks). In case any of the output graphs are not detected, i.e the body or hand is not seen, the values of those nodes are set to be zero. Size normalization of the coordinates is done by MediaPipe.

Most signs take about 1.5 seconds to complete and 24 frames provide just enough time to perform the gesture. 24 also factorizes well into 2,2,2,3 which will be useful later during striding.

24 frames per action are captured to get a data tensor of shape [24,61,3]. In order to feed this data into the GCN model, the data of shape [Batch size, channels, Frames, Nodes] is required. The data which is in the form [Frames, Nodes, channels] is reshuffled to the required format - [1,3,24,61], to be fed in for inference.

D. Graph Convolutional Network (GCN)

Just like CNNs [16], GCNs [1] use convolution to perform feature extraction. But unlike CNNs, they are more general in that they can operate on all graph data through a technique called message passing. The graphs can be static or dynamic. In this case, it is a Spatio-Temporal graph, which is static over node-edge structure but dynamic over values of the nodes across time. A specific version of Spatio-Temporal GCN [2] which is specialized for human action recognition called 2-stream Adaptive GCNs (AAGCN) is used [20].

AAGCN layers implemented by the library PyTorch-geometric-temporal [12] are used. The model has 10 AAGCN layers strategically stridden to provide 1 output per 24 frames which is fed into a linear layer in order to obtain logits for each action class in the sign vocabulary. The Top-N of the predicted logits is then fed into the NLP model for disambiguation.

E. Next-word Predictor NLP model

A next-word predictor model is used for predicting the next word in a sentence. It takes an input of previous words as tokens to predict the probability of occurrence of a word in the next position. This probability is assigned as a score to each word of the vocabulary. In conjunction with the GCN model, it is used to predict the most probable word that the user is performing. The preceding sentence as well as the Top-N words predicted by the GCN model are taken as input for the NLP model to perform disambiguation. The word with the highest score among the Top-N words, as assigned by the NLP model, is selected. This word is then appended to the sentence to continue the inference loop. The NLP model chosen for this task is the pre-trained 'Electra Transformer' from HuggingFace [22].

IV. Training

A. Model

The proposed GCN model contains 10 AAGCN strided layers, followed by 1 linear layer with a dropout of 0.4. It is built using PyTorch-Geometric-Temporal extension which is specially designed to process Spatio-temporal signals. Fig 4 shows the representative model and Fig 5 shows the working of each AAGCN layer.

The model is trained on cross-entropy loss function, a batch size of 32 with the Adam optimizer of learning rate 0.01. The network was trained for 50 epochs as training it any higher would overfit the training data.

B. Integrating NLP

The NLP model that is used for language modelling is Google's Electra small generator model. There are many other models such as GPT-2 [30], BERT [31], ROBERTA [32], etc. This model is chosen because it is considerably more efficient than the alternatives and provides good accuracy. Fig 6 shows some of the comparisons between the models using The General Language Understanding Evaluation (GLUE) benchmark scores against Floating point Operations per Second (FLOPs).

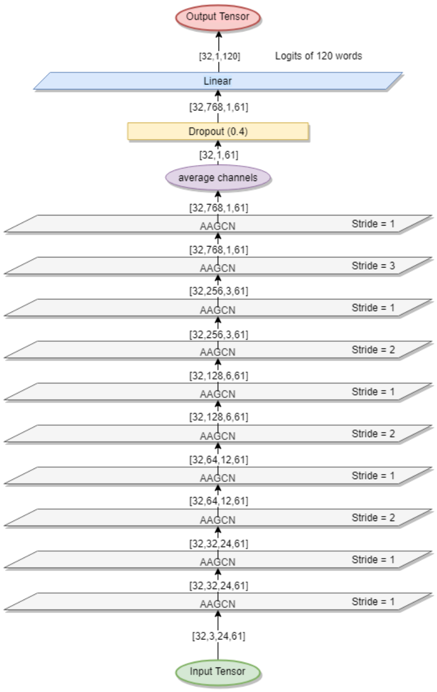

Fig. 4. GCN model with training input

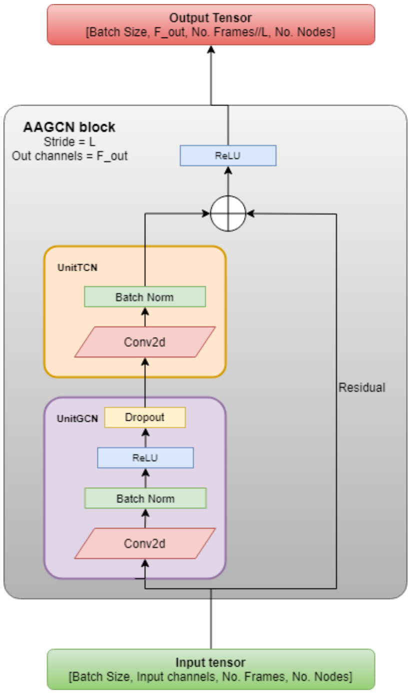

Fig. 5. AAGCN block

Electra has a high GLUE score while having a low FLOPs value, making it highly cost-effective.

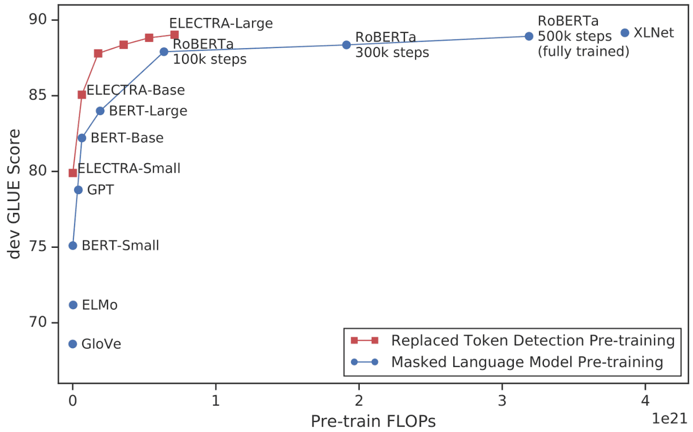

Fig. 6. Comparing Electra to other State-of-the-Art NLP models [26]

To get a prediction, the sentence is first encoded using the model's tokenizer. A prediction is made for all the words that follow the sentence, with values assigned to each word based on how well it follows the sentence preceding it. Language modelling is used to choose the best of the Top-N words that fit the antecedent, the word which has the highest value assigned to it by the NLP model from the Top-5 words is chosen.

Algorithm 2 Prediction from NLP

```
**Require:** Hand sign from user
  $pred \leftarrow getGCNPrediction(hand\_sign\_array)$
  $top5words = topk(pred, 5)$
  $sentence \leftarrow$ ”$Preceding\ sentence$ ”
  $nlp\_scores \leftarrow []$
  $i \leftarrow 0$
  **for each** *word in top5words* **do**
    $score \leftarrow ElectraPred(word, sentence)$
    $nlp\_scores \leftarrow nlp\_scores + score$
  **end for**
  $top\_word\_index \leftarrow Argmax(nlp\_scores)$
  $predicted\_word \leftarrow top5words[top\_word\_index)]$
  $updated\_sentence \leftarrow sentence + predicted\_word$
```

V. Results and Discussions

The results of training and testing are presented in this section. The Top-1 accuracy of the proposed GCN model is observed to be 84% and the Top-5 accuracy is observed to be 97%.

The GCN model was also tested by varying the number of AAGCN layers, with and without strides. The results of the Top-1 accuracy of models are presented in Table I.

The proposed approach also helps solve ambiguity between very similar signs in most sign languages. For example, Fig 7 shows the similarity between signs in American Sign Language [27]. The context in which the words ”loud” and ”surprise” are used differ vastly. The GCN model can get the sign wrong occasionally as the sign gestures are almost identical but the context of the sentence can help identify the correct word. Fig 8 shows indistinguishable signs that represent a similar case of lexical ambiguity that prevails spoken languages.

TABLE I
Accuracy of varying GCN models

AAGCN Layers	With strides	Accuracy
4	no	68
4	yes	70
10	no	83
10	yes	84
14	no	75
14	yes	76

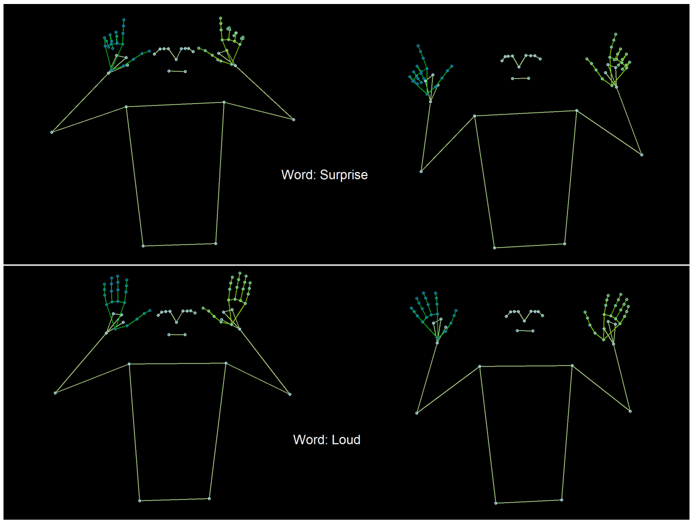

Fig. 7. Comparison of similarity between signs in American Sign Language

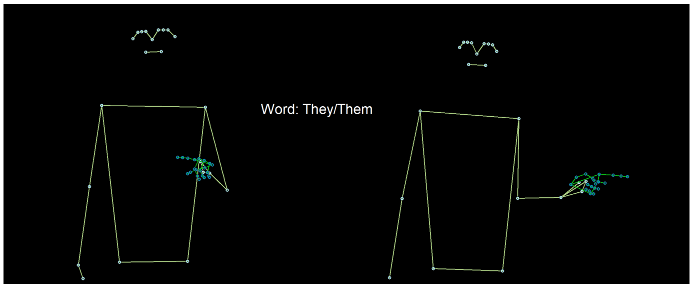

Fig. 8. Identical signs in American Sign Language

A disambiguation example is shown in Table II. The sign ”Them” is performed in the context of the preceding sentence ”After lunch, all four of —– ”. The Top-5 words predicted from the GCN model are assigned scores using the NLP model. Although the Top-1 word predicted from the GCN is ”They”, the final word is then chosen to be ”Them” which has the highest NLP score.

TABLE II
NLP SCORES FOR TOP-5 GCN PREDICTIONS

Preceding sentence: ”After lunch, all four of ”	
Top-5 GCN words	**NLP Scores**
They	tensor(11.6432)
Them	tensor(16.8974)
All	tensor(8.3795)
Their	tensor(9.3995)
Where	tensor(4.6770)
Final predicted sign	Them

The partial sentences from the hand-crafted labelled dataset are used for comparing the two approaches:

1) GCN-only: The Top-1 word from the GCN model is chosen for word prediction.
2) Proposed Context-based solution: The Top-N words are chosen along with language modelling. In the proposed approach, N is chosen to be 5 since Top-5 accuracy of most current State-Of-The-Art models are available for comparison.

TABLE III
COMPARISON BETWEEN GCN-ONLY AND THE PROPOSED APPROACH

Model	**Accuracy**
GCN-only	78.72%
Proposed	97.87%

The results of the comparison between the two approaches are listed in Table III. The NLP cross verification at the end of GCN prediction provides ample boost in accuracy. The proposed NLP approach can also be used at the end of any sign gesture recognition model and find a little boost in the accuracy for sentence formation, provided the NLP model is trained on a representative corpus.

VI. CONCLUSION

This approach offers a novel way to combine the language aspect of Sign Language with the current approaches and make a consumer-ready SLR system feasible. Such an application easily transfers into translation use cases, educational and training purposes. It increases user accessibility for the deaf and dumb.

According to the hypothesis presented in the paper, the context of a sentence can serve as a method to select the best prediction made by a traditional approach. The proposed system uses a dataset of 120 isolated sign words for training the sign recognition model and a dataset consisting of 94 sentences to test the hypothesis. As per the results obtained, it is seen that the proposed approach does have a significant increase in the accuracy of word prediction and the hypothesis can be accepted to be true.

Hopefully, this contribution sparks further progress in this domain. In the future, expanding this work to the Indian Sign Language (ISL) domain will be considered.

REFERENCES

[1] Wu F, Souza A, Zhang T, Fifty C, Yu T, Weinberger K. Simplifying graph convolutional networks. InInternational conference on machine learning 2019 May 24 (pp. 6861-6871). PMLR.

[2] Yan S, Xiong Y, Lin D. Spatial temporal graph convolutional networks for skeleton-based action recognition. InThirty-second AAAI conference on artificial intelligence 2018 Apr 27.

[3] Ekbote J, Joshi M. Indian sign language recognition using ANN and SVM classifiers. In2017 International conference on innovations in information, embedded and communication systems (ICIIECS) 2017 Mar 17 (pp. 1-5). IEEE.

[4] Kishore PV, Kumar DA, Sastry AC, Kumar EK. Motionlets matching with adaptive kernels for 3-d indian sign language recognition. IEEE Sensors Journal. 2018 Feb 28;18(8):3327-37.

[5] Bantupalli K, Xie Y. American sign language recognition using deep learning and computer vision. In2018 IEEE International Conference on Big Data (Big Data) 2018 Dec 10 (pp. 4896-4899). IEEE.

[6] Liu T, Zhou W, Li H. Sign language recognition with long short-term memory. In2016 IEEE international conference on image processing (ICIP) 2016 Sep 25 (pp. 2871-2875). IEEE.

[7] Mittal A, Kumar P, Roy PP, Balasubramanian R, Chaudhuri BB. A modified LSTM model for continuous sign language recognition using leap motion. IEEE Sensors Journal. 2019 Apr 9;19(16):7056-63.

[8] Koller O, Zargaran O, Ney H, Bowden R. Deep sign: Hybrid CNN-HMM for continuous sign language recognition. InProceedings of the British Machine Vision Conference 2016 2016.

[9] de Amorim CC, Macêdo D, Zanchettin C. Spatial-temporal graph convolutional networks for sign language recognition. InInternational Conference on Artificial Neural Networks 2019 Sep 17 (pp. 646-657). Springer, Cham.

[10] Cheng K, Zhang Y, He X, Chen W, Cheng J, Lu H. Skeleton-based action recognition with shift graph convolutional network. InProceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition 2020 (pp. 183-192).

[11] Jiang S, Sun B, Wang L, Bai Y, Li K, Fu Y. Skeleton aware multi-modal sign language recognition. InProceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition 2021 (pp. 3413-3423).

[12] Wang J, Nie X, Xia Y, Wu Y, Zhu SC. Cross-view action modeling, learning and recognition. InProceedings of the IEEE conference on computer vision and pattern recognition 2014 (pp. 2649-2656).

[13] Rao GA, Syamala K, Kishore PV, Sastry AS. Deep convolutional neural networks for sign language recognition. In2018 Conference on Signal Processing And Communication Engineering Systems (SPACES) 2018 Jan 4 (pp. 194-197). IEEE.

[14] Starner T, Pentland A. Real-time american sign language recognition from video using hidden markov models. InMotion-based recognition 1997 (pp. 227-243). Springer, Dordrecht.

[15] Fujimori Y, Ohmura Y, Harada T, Kuniyoshi Y. Wearable motion capture suit with full-body tactile sensors. In2009 IEEE International Conference on Robotics and Automation 2009 May 12 (pp. 3186-3193). IEEE.

[16] Albawi S, Mohammed TA, Al-Zawi S. Understanding of a convolutional neural network. In2017 International Conference on Engineering and Technology (ICET) 2017 Aug 21 (pp. 1-6). Ieee.

[17] Lugaresi C, Tang J, Nash H, McClanahan C, Uboweja E, Hays M, Zhang F, Chang CL, Yong MG, Lee J, Chang WT. Mediapipe: A framework for building perception pipelines. arXiv preprint arXiv:1906.08172. 2019 Jun 14.

[18] Zhang F, Bazarevsky V, Vakunov A, Tkachenka A, Sung G, Chang CL, Grundmann M. Mediapipe hands: On-device real-time hand tracking. arXiv preprint arXiv:2006.10214. 2020 Jun 18.

[19] Bazarevsky V, Grishchenko I, Raveendran K, Zhu T, Zhang F, Grundmann M. BlazePose: On-device Real-time Body Pose tracking. arXiv preprint arXiv:2006.10204. 2020 Jun 17.

[20] Shi L, Zhang Y, Cheng J, Lu H. Two-stream adaptive graph convolutional networks for skeleton-based action recognition. InProceedings of the IEEE/CVF conference on computer vision and pattern recognition 2019 (pp. 12026-12035).

[21] https://ai.googleblog.com/2020/12/mediapipe-holistic-simultaneous-face.html

[22] Clark K, Luong MT, Le QV, Manning CD. Electra: Pre-training text encoders as discriminators rather than generators. arXiv preprint arXiv:2003.10555. 2020 Mar 23.

[23] Bradski, Gary. (2000). The openCV library. Doctor Dobbs Journal. 25. 120-126.

[24] Sincan OM, Keles HY. Autsl: A large scale multi-modal turkish sign language dataset and baseline methods. IEEE Access. 2020 Oct 1;8:181340-55.

[25] Wolf T, Debut L, Sanh V, Chaumond J, Delangue C, Moi A, Cistac P, Rault T, Louf R, Funtowicz M, Davison J. Huggingface's transformers: State-of-the-art natural language processing. arXiv preprint arXiv:1910.03771. 2019 Oct 9.

[26] https://ai.googleblog.com/2020/03/more-efficient-nlp-model-pre-training.html

[27] Liddell SK, Johnson RE. American sign language: The phonological base. Sign language studies. 1989;64(1):195-277.

[28] Liu J, Shahroudy A, Perez M, Wang G, Duan LY, Kot AC. Ntu rgb+ d 120: A large-scale benchmark for 3d human activity understanding. IEEE transactions on pattern analysis and machine intelligence. 2019 May 14;42(10):2684-701.

[29] Wang J, Nie X, Xia Y, Wu Y, Zhu SC. Cross-view action modeling, learning and recognition. InProceedings of the IEEE conference on computer vision and pattern recognition 2014 (pp. 2649-2656).

[30] Radford A, Wu J, Child R, Luan D, Amodei D, Sutskever I. Language models are unsupervised multitask learners. OpenAI blog. 2019 Feb 24;1(8):9.

[31] Devlin J, Chang MW, Lee K, Toutanova K. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805. 2018 Oct 11.

[32] Liu Y, Ott M, Goyal N, Du J, Joshi M, Chen D, Levy O, Lewis M, Zettlemoyer L, Stoyanov V. Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692. 2019 Jul 26.