Pose-based Sign Language Recognition using GCN and BERT

Anirudh Tunga*
Purdue University
atunga@purdue.edu

Sai Vidyaranya Nuthalapati*
vidyaranya.ns@gmail.com

Juan Wachs
Purdue University
jpwachs@purdue.edu

Abstract

Sign language recognition (SLR) plays a crucial role in bridging the communication gap between the hearing and vocally impaired community and the rest of the society. Word-level sign language recognition (WSLR) is the first important step towards understanding and interpreting sign language. However, recognizing signs from videos is a challenging task as the meaning of a word depends on a combination of subtle body motions, hand configurations and other movements. Recent pose-based architectures for WSLR either model both the spatial and temporal dependencies among the poses in different frames simultaneously or only model the temporal information without fully utilizing the spatial information.

We tackle the problem of WSLR using a novel pose-based approach, which captures spatial and temporal information separately and performs late fusion. Our proposed architecture explicitly captures the spatial interactions in the video using a Graph Convolutional Network (GCN). The temporal dependencies between the frames are captured using Bidirectional Encoder Representations from Transformers (BERT). Experimental results on WLASL, a standard word-level sign language recognition dataset show that our model significantly outperforms the state-of-the-art on pose-based methods by achieving an improvement in the prediction accuracy by up to 5%.

1. Introduction

Hearing and vocally impaired people use sign language instead of spoken language for communication. Just like any other language, sign language has an underlying structure, inter alia, grammar and intricacies to allow users (signers or interpreters) to fully express themselves. To comprehend sign language, one must consider and understand multiple aspects such as hand movements, shape and orientation of the hand, shoulder orientation, head movements and facial expressions. The study of accurately recognizing and understanding sign language technique falls under the ambit of sign language recognition.

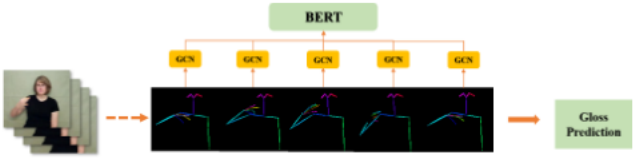

Figure 1. We train a model based on GCN and BERT to predict the glosses from the poses extracted from video frames.

According to [22], there are approximately 500,000 users of American Sign Language in the US itself. While on one hand, hearing and vocally impaired communities are completely dependent on sign language for communication, on the other hand, the rest of the world does not understand sign language, creating a communication barrier between the two groups. It is also unlikely that people without such impairments will learn an additional language which is not seen as a necessity for them. This gap between the rest of the world and the hearing and vocally impaired community can be reduced by developing Automatic Sign Language Recognition (ASLR).

Sign language recognition can be broadly classified into two parts: word-level sign language recognition (WSLR) and sentence-level sign language recognition. WSLR is the fundamental building block for interpreting sign language sentences. As shown in Figure 1, signalling a sign language word requires very subtle body movements that makes WSLR a particularly challenging problem. In this paper, we focus on WSLR by exploiting the information from human skeletal motion. In WSLR, given a sign language video, the goal is to predict the word that is being signalled in the video. 'Gloss' is another term for representing the word that is being shown. Recently, deep learning techniques have shown a huge promise in the field of WSLR [38, 18, 26, 23]. The techniques that are employed for ASLR can be divided into two categories: 1. Methods based on 2D pose estimation, and 2. Architectures utilizing the holistic image features. We believe that human skeletal motion plays a significant role in conveying what word the person is signalling. Hence, this work focuses on

*equal contribution

a pose-based model to tackle the problem of WSLR. The existing pose-based methods either model both the spatial and temporal dependencies between the poses in different frames simultaneously or only model the temporal information without fully utilizing the spatial information [26]. Inspired by [20], where the authors have used late temporal fusion to achieve a performance boost in action recognition, we propose a novel pose-based architecture, GCN-BERT, which first captures the spatial interactions in every frame comprehensively before explicitly utilizing the temporal dependencies between various frames in the video. We validate our architecture on the recently released large-scale WLASL dataset [26] and experimental results show that the proposed model significantly outperforms the state-of-the-art pose-based models by achieving an accuracy improvement of up to 5%.

2. Related Work

Sign language recognition mainly involves three phases - feature extraction phase, temporal modelling phase, and prediction phase. Historically, spatial representation was generated using hand crafted features like HOG-based features [4, 12], SIFT-based features [52, 45], and frequency domain features [1, 3]. Temporal modelling was done using Hidden Markov Models (HMM) [42, 16, 57], and hidden conditional random fields [50]. Some works utilized Dynamic Time Wrapping [40, 29] to handle varying frame rates. The prediction phase was treated as a classification problem, and models like Support Vector Machine (SVM) [34] were used to predict the words from the signs. The vast majority of traditional sign language recognition models are evaluated on small scale datasets, which had less than one hundred words [56, 30, 25].

With the advent of deep neural networks, there was a significant boost in the performance for many video-based tasks like action recognition [15, 17], and gesture recognition [5]. Both, action recognition and sign language recognition share a similar problem structure. Inspired from the network architectures for action recognition, new architectures for sign language recognition were proposed. For example, a CNN-based architecture was used for sign language recognition in [37], and a frame-based CNN-HMM model for sign language recognition was proposed in [24]. These two papers are representative of a more general trend of deep neural-based architectures for sign language recognition. It was learnt that these works can be partitioned into two categories: image appearance based methods, and pose-based methods, which are presented in more detail below.

2.1. Image appearance based methods

Word level sign language recognition focuses mainly on intricate hand and arm movements, while the background is not very useful in recognition. In this section, we discuss some relevant image based methods for action recognition and sign language recognition.

Utilizing the feature extraction capability of deep neural networks, Simonyan et al., [41], uses a 2D CNN to create a holistic representation of each input frame of the video and then uses those representations for recognition. Temporal dynamics of a video can be modelled by sequence modelling using recurrent neural networks. The works [55, 15], use Long Short-Term Memory (LSTMs) to model the temporal dynamics of the features extracted through CNNs. In [13], a 2D CNN-LSTM architecture, where, in parallel with the LSTMs, it also uses a weakly supervised gloss-detection regularization network, consisting of stacked temporal 1D convolutions. A simpler variant of LSTMs, Gated-recurrent Units (GRU) [11], which consist of only two gates (update and reset gates), and have the internal state (output state) fully exposed, have also been used for temporal modelling [54].

While the above works used RNNs to model the gesture temporal behaviour, a few works have used CNNs to achieve this. For instance, 3D CNNs [44] can not only learn the holistic representation of each input frame, but also the spatio-temporal features. The C3D [48] model was the first model to use 3D CNNs for action recognition. In [19], the I3D [8] architecture has been trained and adopted for sign language recognition. In [58], the authors extended the I3D architecture by adding a RNN. Recent work [20], has used the Bidirectional Encoder Representations from Transformers (BERT) [14] at the end of a 3D CNN.

2.2. Pose-based methods

2.2.1 Pose estimation

Human pose estimation involves localizing keypoints of human joints from a single image or a video. Historically, pictorial structures [39] and probabilistic graphical models [53] were used to estimate the human pose. Recent advances in deep learning have greatly boosted the performance of human pose estimation. Two main methods exist in localizing the keypoints: directly regressing the x, y coordinates of joints, and estimating keypoint heatmaps followed by a non-maximal suppression technique. In [47], Toshev et al. introduced 'Deep Pose', where they directly regress the keypoints from the frame. In [46], Tompson et al. used a ConvNet and a graphical model to estimate the keypoint heatmaps. Recent works [35, 6] have improved the performance of human pose estimation significantly using heatmap estimation. Though pose estimation succeeds at estimating positions of human joints, it does not explore the spatial dependencies among these estimated keypoints or joints.

2.2.2 Adapting pose estimation for activity recognition

Human poses and their dependencies can be used to recognize actions. In [51], the authors built a graphical model using poses to recognize actions. In [9], the authors used pose-based feature descriptors for activity recognition. RNNs have been used to model the temporal sequential information of the pose movements, and the representation output by RNN has been used for the sign recognition. In [33], the authors analyze human motions by using RNNs to model the pose sequences. A few works, for instance, [28], have used graph network based architectures to model the spatial and temporal dependencies of the pose sequence.

Recently, pose-based methods have been used for sign language recognition. In [26], the authors used the pose sequence extracted from the video followed by a graph convolution network to model the spatio-temporal dependency for sign language recognition. However, the existing works on sign language recognition, either model the spatial and temporal dependencies together or ignore the spatial dependencies, and only model the temporal dependencies. In order to overcome these limitations, we propose a model where we extract the spatial and temporal encoding separately and comprehensively use these two different components in our architecture.

3. Proposed Architecture

3.1. Notation

We denote the WSLR dataset containing N labelled training examples by $\mathcal{D} = \{X_i, Y_i\}_{i=1}^{N}$ where $X_i \in \mathbb{R}^{l*w*h*3}$ is the input RGB video; l, w, h represent the length, width and height of the video respectively; and $Y_i \in [1, 2, ..., G]$ is the sign corresponding to the input video, and G is the number of output classes.

3.2. Overview

Sign languages are based on a sequence of body part movements to convey a message. Deep learning methods that are trained on skeletal information have shown a great promise in detecting and analyzing such body movements [2]. This is primarily because using poses helps the model to focus on the most important parts of the image rather than focusing on the unimportant components (for e.g. lighting conditions, and background of the image). Pose-based approaches have been widely used in the literature to extract such skeletal information. They mainly use RNNs to capture the temporal information of the changes in the pose. However, such pose-based approaches fail to capture the spatial interactions between various keypoints in the pose which is extremely important for sign language recognition. Deviating from the existing pose-based architectures which model the temporal and spatial interactions together, we delegate the modelling of the two interactions to two individual components in our model. Inspired by the success of graph neural network in capturing the spatial information for sign language recognition, we use graph convolutional networks (GCN) [26, 7, 21] to encode the spatial relationship among various body key points. Once the spatial information is collected across all the frames, we deploy BERT [14], a transformer-based architecture [49] to collate the temporal information. Figure 2 depicts the GCN-BERT architecture in detail.

We represent a pose using K keypoints representing the upper body and both the hands. Each keypoint is a 2-dimensional vector showing the location of the corresponding keypoint in the frame. These keypoints serve as inputs to our model. Further, as videos vary in length, we randomly select 50 consecutive frames from the input video to maintain a constant input size. This information is passed as input to our model for predicting the gloss.

3.2.1 Pose-Based Graph Convolution Network

We extract the spatial information from every video frame using a graph neural network. Inspired by [26], we use GCN to extract this information. The input to the graph is represented using $x_t \in X$ and $t \in [1, 2, ..., T]$, where T is the number of frames in the video and $x_t \in \mathbb{R}^{K\times 2}$ (we multiply it with 2 as we use 2D coordinates for locating keypoints). We represent the human body as a fully connected graph [10] which allow us to express the relative positions of various body keypoints, as essential parts in order to accurately determine the gloss. While the keypoints form the nodes in the graph, the edges are weighted and learnt during the training process. We represent the weigthed adjacency matrix as $A \in \mathbb{R}^{K\times K}$. Initially, the nodes are represented using the 2D coordinates for the corresponding keypoint. During the training process, the node representations are updated using the update rules below:

$$H^{(l+1)} = f(H^{(l)}, A), \tag{1}$$

$$f(H^{(l)}, A) = \sigma\left(AH^{(l)}W^{(l)}\right) \tag{2}$$

where (i) $H^{(l)} \in \mathbb{R}^{K\times F}$ stands for the representation of nodes in l-th neural network layer, (ii) F is the output feature from the previous layer. Initially, H is set to X and so we take F to be equal to 2, (iii) $W^{(l)} \in \mathbb{R}^{(F\times F')}$ stands for the weight matrix in l-th neural network layer, and (iv) σ represents a non-linear activation function which is *tanh* in our case. We can stack L such layers to create accurate representations of each node in the graph i.e. $l \in [1, 2, ..., L]$. L such stacked graph convolutional layers constitute one GCN network. The updated node representations encode spatial information of all the keypoints and interactions between them.

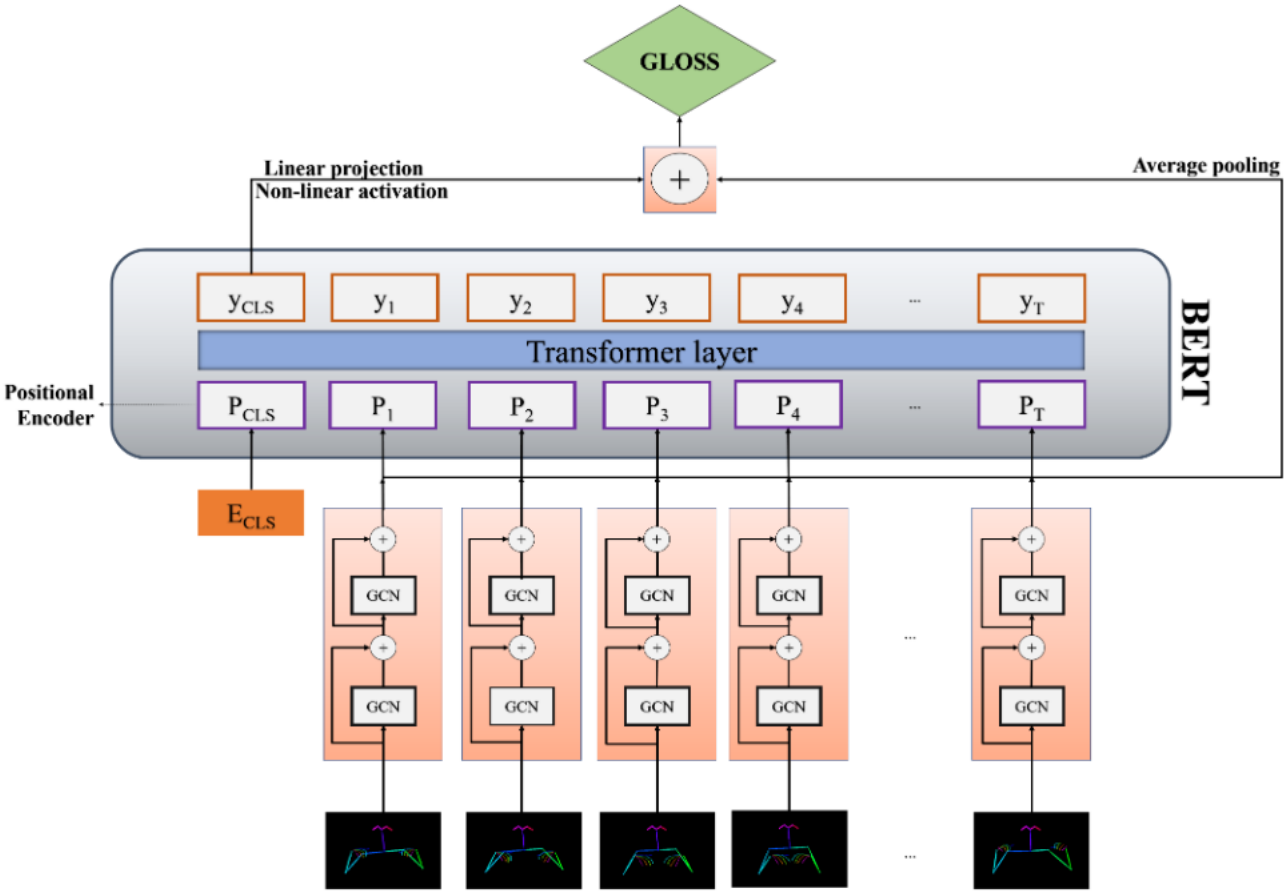

Figure 2. Illustration of the proposed GCN-BERT architecture. The poses extracted from the video are fed to the GCN to model spatial dependencies in the frames. This is followed by BERT to model the temporal dependencies between various frames in the video.

Similar to [26], we stack multiple GCN networks on top of each other and provide residual connections between the stacked GCNs. We represent the input to a single GCN network as I and the output as $\tilde{O}$. With residual connections, the actual output of a single GCN network is given as follows:

$$O = \tilde{O} + I \tag{3}$$

This allows the network to learn to bypass a GCN network if required. We stack B such networks to get the final output representations of keypoints. In Figure 2, we show the case where B equals 2. While the process discussed above is for a single frame in the video, the same process can be repeated for all the frames in the video. In the end, we have the encoded spatial information for all the frames in the video denoted by $S \in \mathbb{R}^{T \times K \times F}$ where $S = [H_1, H_2, \ldots, H_T]$. We also calculate the mean of all the spatial encodings along the temporal direction, denoted by $\hat{S} \in \mathbb{R}^{K \times F}$. This is followed by a fully connected layer and a non-linear activation to project into a G-dimensional space, where G, is the number of output classes. Let us denote the resultant encoding by $\hat{U}$. This will later be used to provide a skip connection from the output of GCN to the output of BERT.

3.2.2 Temporal modelling using BERT

Recently, architectures based solely on multi-head self-attention have achieved state-of-the-art results on sequence modelling tasks [49, 31, 32]. One such architecture - Bidirectional Encoder Representations from Transformers (BERT) [14] - has shown a dramatic success in many downstream Natural Language Processing tasks. It has been designed to learn bidirectional representations by considering both the left and right contexts in all its layers. While it was initially introduced for NLP tasks, it is recently being used to model other sequential tasks such as action classification and video captioning [43]. Inspired by the success of BERT in problems related to activity recognition [20], we use BERT to learn bidirectional representations over sequence of encoded spatial information S generated from GCN. This enables the model to learn contextual informa-

Table 1. Top-1, top-5, top-10 accuracy (%) achieved by pose-based models on WLASL dataset.

	WLASL100			WLASL300		
Methods	Top-1	Top-5	Top-10	Top-1	Top-5	Top-10
Pose-GRU [26]	46.51	76.74	85.66	33.68	64.37	76.05
Pose-TGCN [26]	55.43	78.68	87.60	38.32	67.51	79.64
GCN-BERT(ours)	**60.15**	**83.98**	**88.67**	**42.18**	**71.71**	**80.93**

tion from both left and right directions. Similar to [14], the input S is concatenated with learned position embeddings (denoted by P_i for *i*-th input position.) to capture the positional information. Then, we add a classification token s_{cls} to the start of the input. The corresponding output from the last layer in BERT, y_{cls} is passed through a fully connected layer and is eventually used for predicting the gloss.

Single head self-attention in a BERT layer computes the output as follows [14, 20]:

$$M\left(s_i\right)=\left(\frac{1}{N(s)}\sum_{\forall j}V(s_j)f(s_i,s_j)\right) \tag{4}$$

where $s_i \in S$ represents the spatial information corresponding to *i*-th pose extracted from GCN. $N(s)$ is the normalization factor and is used to produce a softer attention distribution and to avoid extremely small gradients [49]. $f(s_i, s_j)$ is used to measure the similarity between s_i, s_j and is defined as $softmax_j(Q(s_i)^T K(s_j))$, where the functions Q and K are learned linear projections. Combined with V, which is also a learned linear projection, the functions Q and K project the inputs to a common space before applying the similarity measure.

The single head self-attention sub-layer computation above predominantly consists of linear projects. To add non-linearity to the model, we use Position-wise Feed-Forward Network (PFFN) to the outputs of the self-attention sub-layer identically and separately at each position.

$$\begin{aligned} PFFN(x) &= \mathbf{W_2}GELU(\mathbf{W_1}x+\mathbf{b_1})+\mathbf{b_2} \\ GELU(x) &= x\phi(x) \end{aligned} \tag{5}$$

where $\phi(x)$ represents the cumulative distribution function of the standard Gaussian distribution and $\mathbf{W_1}, \mathbf{W_2}, \mathbf{b_1}, \mathbf{b_2}$ are learnable parameters. Combining Equations 4 and 5, we calculate y_i as follows:

$$y_i = PFFN(M(x_i)). \tag{6}$$

While the Equations 4, 5, 6 show attention calculation for single head, we can calculate attention using multiple heads and average the outputs. This constitutes a transformer layer.

Using the equations above we calculate y_{cls}, the output from the transformer layer corresponding to x_{cls}, which is passed through a fully connected layer projecting it into a G-dimensional space followed by *tanh* activation . Let us denote the resulting spatial-temporal encoding by $\hat{V}$.

We provide a skip connection from the output of GCN to the output of BERT as follows:

$$\hat{y} = \hat{U} + \hat{V}, \tag{7}$$

which is followed by a softmax layer to predict the output label. We use the standard cross-entropy loss to train the neural network.

4. Experiments and Analysis

In this section we describe the experimental setup, and provide quantitative and qualitative results.

4.1. Dataset

Table 2. Dataset statistics

	Classes	Train	Validation	Test
WLASL100	100	1442	338	258
WLASL300	300	3548	901	668

The dataset used to validate the results in the paper is the Word Level American Sign Language (WLASL) dataset [26]. This dataset has been recently introduced and supports large scale WSLR. The videos contain native American Sign Language (ASL) signers or interpreters, showing signs of a specific English word in ASL. We show the dataset split of WLASL in Table 2 [27].The number of classes represents the number of different glosses in the dataset. For our experiments, we use the public dataset split released by the dataset authors.

4.2. Implementation details

The proposed GCN-BERT model has been implemented using PyTorch [36]. In sign language, different meanings have very similar sign gestures and the difference can only be made out using the contextual information. Hence, following [26], we use top-*K* accuracy to evaluate the performance of the model. We provide an evaluation of the proposed method using three different values of *K*, specifically 1, 5, 10. We train the model for 100 epochs using using Adam optimizer with an initial learning rate of 10^{-3},

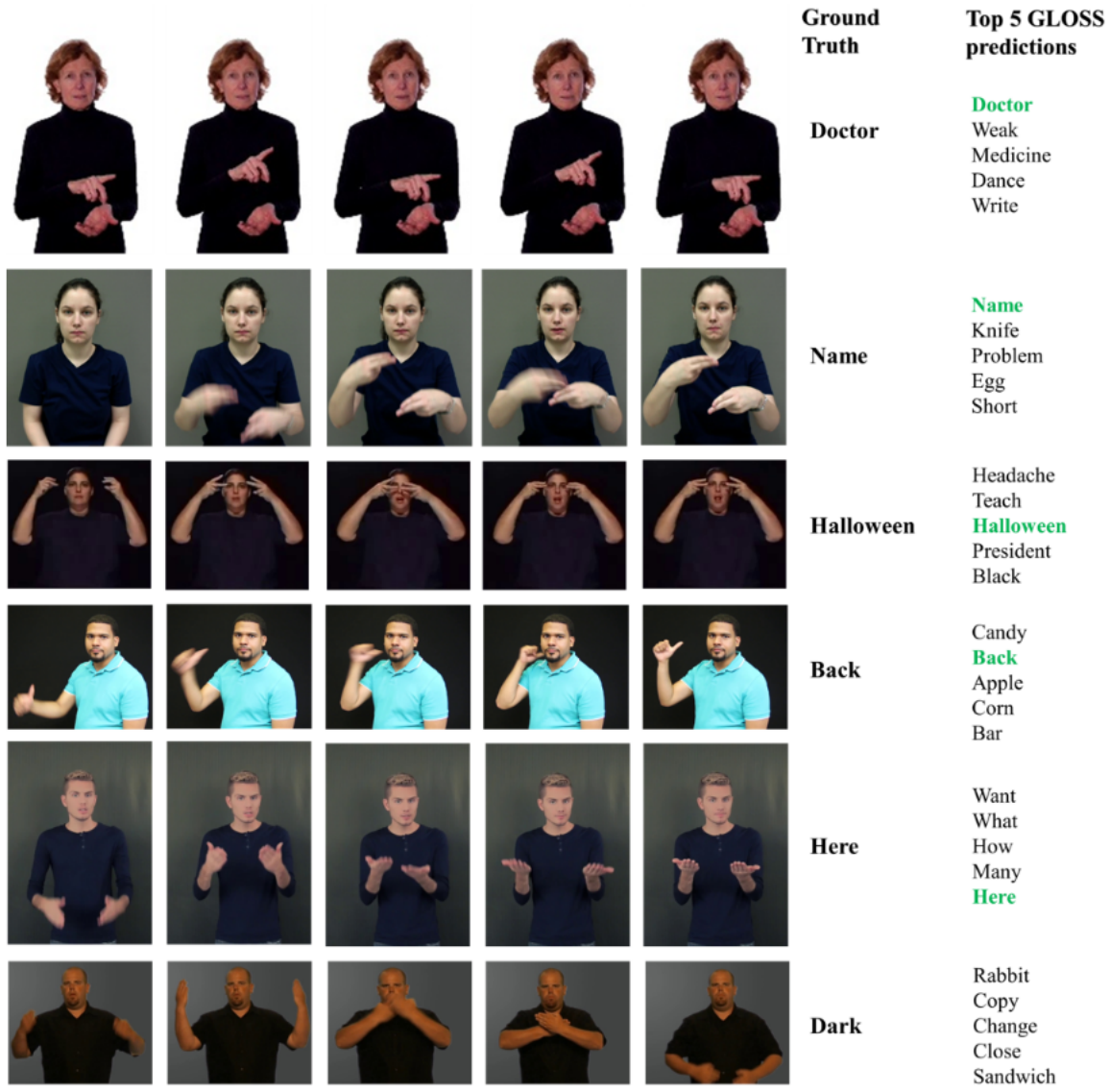

Figure 3. Left to right: video frames, ground truth, and predicted glosses for various videos.

a weight decay of 10^{-8}. Following [26], we extract 13 upper body keypoints and 21 keypoints corresponding to each hand from each frame of the video using OpenPose [7].

4.3. Results and Analysis

Table 1 shows a comparison of our model with the existing pose-based architectures. The results show that the proposed GCN-BERT model improves the existing state-of-the-art pose-based sign language recognition by a significant margin. This indicates that modelling spatial and temporal relationships separately and explicitly with GCN and BERT respectively improves the prediction accuracy.

4.4. Qualitative Analysis

In Figure 3, we show various videos along with their ground truths and predicted glosses. Though our architecture is pose-based, we show the RGB frames of the video in Figures 3 & 4 for better understanding of the reader. In Figure 3, we observe that for predicting the word 'Doctor', the model is able to capture the temporal dependencies accurately and grasp even the slightest movement in the hand. As we can see from Figures 3 and 4, the signs for the words 'halloween' and 'headache' are very similar and can be confused even by a human, so it is natural to expect the model to confuse between them (can be seen by observing the top-5 predictions for 'halloween' in Figure 3). We see similar

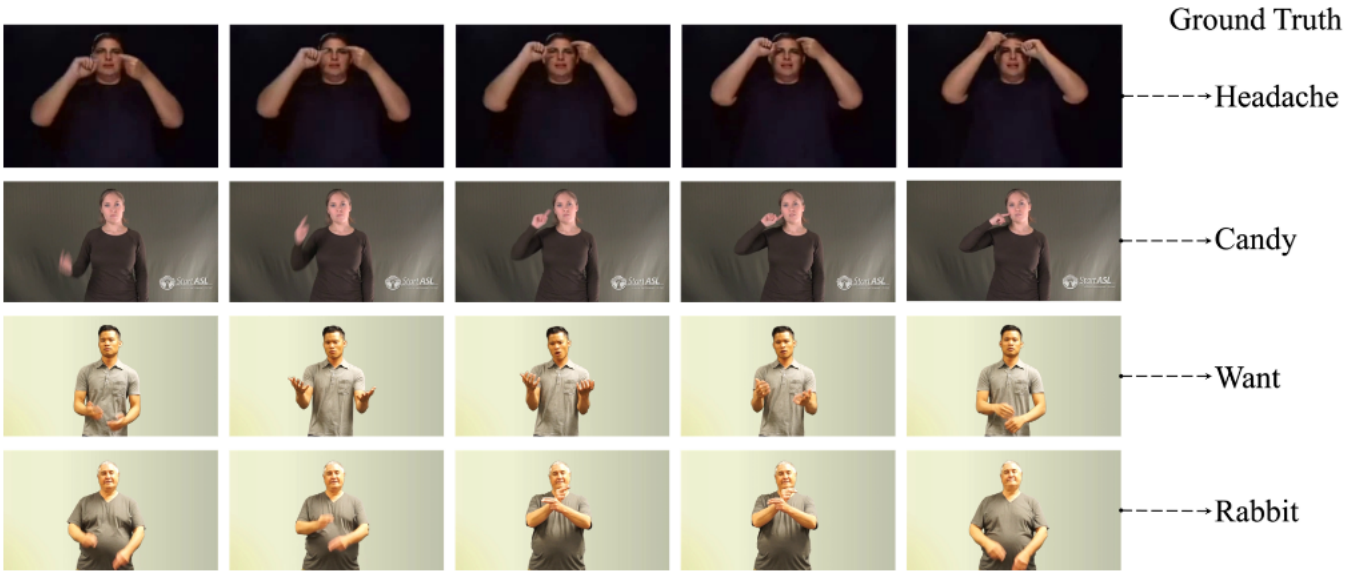

Figure 4. Videos and corresponding ground truth, showing similarities to the predicted glosses in Fig. 3.

trend for other videos corresponding to 'back', 'here' and 'dark'. For the word 'back', we can observe that there is a very subtle difference with the top prediction - 'candy'. Given that the signer is also slightly rotated in the frame leads to very similar poses for the words 'black' and 'candy' making it hard to differentiate. Also, in-plane and out-of-plane movements are not being differentiated by the model due to the fact that we are only utilizing the 2D spatial information. Figure 3 shows a few more signs for which the topmost prediction is not the ground truth. Figure 4 contains the videos for the predicted words for comparison of the videos.

In Table 1, we see the effect of increasing the vocabulary size (number of classes) on the performance of the model. Increasing the vocabulary size contributes to a fall in the accuracy. This happens because the dataset consists of ambiguous signs and their meaning depends on the context. Increasing the number of classes, also increases the number of such ambiguous signs, leading to a fall in the accuracy. Based on the observations, we can say that the performance on a smaller dataset does not scale well with a larger dataset.

4.5. Conclusion and Future Work

This work addresses the fundamental problem of sign language recognition in order to bridge the communication barriers between hearing and vocally impaired people, and the rest of the society. Previous works concerned with this problem either jointly considered both spatial and temporal information or relied mainly on the temporal information. To tackle this issue, this paper proposes a novel pose-based architecture for word-level sign language recognition which aims to predict the meaning of the sign language videos. Further, we showed that modelling spatial and temporal information separately with GCN and BERT provides drastic performance gains over the existing state-of-the-art pose-based models. We validated our model on one of the largest publicly available sign language datasets to show the efficacy of our model. As a part of the future work, we plan to include image-based features into our model to jointly consider both the pose and image related information in order to comprehend the sign language videos.

Acknowledgement

We would like to thank our colleagues, Naveen Madapana, Eleonora Giunchiglia, and Aishwarya Chandrasekaran for their constructive feedback.

References

[1] M Al-Rousan, Khaled Assaleh, and A Tala'a. Video-based signer-independent arabic sign language recognition using hidden markov models. *Applied Soft Computing*, 9(3):990–999, 2009.

[2] Ahmet Alp Kindiroglu, Ogulcan Ozdemir, and Lale Akarun. Temporal accumulative features for sign language recognition. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 0–0, 2019.

[3] Purva C Badhe and Vaishali Kulkarni. Indian sign language translator using gesture recognition algorithm. In *2015 IEEE International Conference on Computer Graphics, Vision and Information Security (CGVIS)*, pages 195–200. IEEE, 2015.

[4] Patrick Buehler, Andrew Zisserman, and Mark Everingham. Learning sign language by watching tv (using weakly aligned subtitles). In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2961–2968. IEEE, 2009.

[5] Necati Cihan Camgoz, Simon Hadfield, Oscar Koller, and Richard Bowden. Using convolutional 3d neural networks for user-independent continuous gesture recognition. In *2016 23rd International Conference on Pattern Recognition (ICPR)*, pages 49–54. IEEE, 2016.

[6] Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Openpose: realtime multi-person 2d pose estimation using part affinity fields. *arXiv preprint arXiv:1812.08008*, 2018.

[7] Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Openpose: realtime multi-person 2d pose estimation using part affinity fields. *arXiv preprint arXiv:1812.08008*, 2018.

[8] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308, 2017.

[9] Guilhem Cheron, Ivan Laptev, and Cordelia Schmid. P-cnn: Pose-based cnn features for action recognition. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, December 2015.

[10] Hsu-kuang Chiu, Ehsan Adeli, Borui Wang, De-An Huang, and Juan Carlos Niebles. Action-agnostic human pose forecasting. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1423–1432. IEEE, 2019.

[11] Kyunghyun Cho, Bart Van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259*, 2014.

[12] Helen Cooper, Eng-Jon Ong, Nicolas Pugeault, and Richard Bowden. Sign language recognition using sub-units. *The Journal of Machine Learning Research*, 13(1):2205–2231, 2012.

[13] Runpeng Cui, Hu Liu, and Changshui Zhang. Recurrent convolutional neural networks for continuous sign language recognition by staged optimization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7361–7369, 2017.

[14] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[15] Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2625–2634, 2015.

[16] Georgios D Evangelidis, Gurkirt Singh, and Radu Horaud. Continuous gesture recognition from articulated poses. In *European Conference on Computer Vision*, pages 595–607. Springer, 2014.

[17] Christoph Feichtenhofer, Axel Pinz, and Andrew Zisserman. Convolutional two-stream network fusion for video action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1933–1941, 2016.

[18] Hamid Reza Vaezi Joze and Oscar Koller. Ms-asl: A large-scale data set and benchmark for understanding american sign language. *arXiv preprint arXiv:1812.01053*, 2018.

[19] Hamid Reza Vaezi Joze and Oscar Koller. Ms-asl: A large-scale data set and benchmark for understanding american sign language. *arXiv preprint arXiv:1812.01053*, 2018.

[20] M Kalfaoglu, Sinan Kalkan, and A Aydin Alatan. Late temporal modeling in 3d cnn architectures with bert for action recognition. *arXiv preprint arXiv:2008.01232*, 2020.

[21] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

[22] PVV Kishore, G Anantha Rao, E Kiran Kumar, M Teja Kiran Kumar, and D Anil Kumar. How many people use asl in the united states? why estimates need updating. *Sign Lang. Stud.*, 16(3):306–335, 2006.

[23] PVV Kishore, G Anantha Rao, E Kiran Kumar, M Teja Kiran Kumar, and D Anil Kumar. Selfie sign language recognition with convolutional neural networks. volume 10, page 63. Modern Education and Computer Science Press, 2018.

[24] Oscar Koller, Hermann Ney, and Richard Bowden. Deep hand: How to train a cnn on 1 million hand images when your data is continuous and weakly labelled. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3793–3802, 2016.

[25] Vaishali S Kulkarni and SD Lokhande. Appearance based recognition of american sign language using gesture segmentation. *International Journal on Computer Science and Engineering*, 2(03):560–565, 2010.

[26] Dongxu Li, Cristian Rodriguez, Xin Yu, and Hongdong Li. Word-level deep sign language recognition from video: A new large-scale dataset and methods comparison. In *The IEEE Winter Conference on Applications of Computer Vision*, pages 1459–1469, 2020.

[27] Dongxu Li, Xin Yu, Chenchen Xu, Lars Petersson, and Hongdong Li. Transferring cross-domain knowledge for video sign language recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6205–6214, 2020.

[28] Maosen Li, Siheng Chen, Xu Chen, Ya Zhang, Yanfeng Wang, and Qi Tian. Actional-structural graph convolutional networks for skeleton-based action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[29] Jeroen F Lichtenauer, Emile A Hendriks, and Marcel JT Reinders. Sign language recognition by combining statistical dtw and independent classification. *IEEE transactions on pattern analysis and machine intelligence*, 30(11):2040–2046, 2008.

[30] Kian Ming Lim, Alan WC Tan, and Shing Chiang Tan. Block-based histogram of optical flow for isolated sign language recognition. *Journal of Visual Communication and Image Representation*, 40:538–545, 2016.

[31] Zhouhan Lin, Minwei Feng, Cicero Nogueira dos Santos, Mo Yu, Bing Xiang, Bowen Zhou, and Yoshua Bengio. A structured self-attentive sentence embedding. 2017.

[32] Peter J Liu, Mohammad Saleh, Etienne Pot, Ben Goodrich, Ryan Sepassi, Lukasz Kaiser, and Noam Shazeer. Generating wikipedia by summarizing long sequences. *arXiv preprint arXiv:1801.10198*, 2018.

[33] Julieta Martinez, Michael J. Black, and Javier Romero. On human motion prediction using recurrent neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[34] Sathish Nagarajan and TS Subashini. Static hand gesture recognition for sign language alphabets using edge oriented histogram and multi class svm. *International Journal of Computer Applications*, 82(4), 2013.

[35] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *European conference on computer vision*, pages 483–499. Springer, 2016.

[36] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019.

[37] Lionel Pigou, Sander Dieleman, Pieter-Jan Kindermans, and Benjamin Schrauwen. Sign language recognition using convolutional neural networks. In *European Conference on Computer Vision*, pages 572–578. Springer, 2014.

[38] Lionel Pigou, Mieke Van Herreweghe, and Joni Dambre. Gesture and sign language recognition with temporal residual networks. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 3086–3093, 2017.

[39] Leonid Pishchulin, Mykhaylo Andriluka, Peter Gehler, and Bernt Schiele. Poselet conditioned pictorial structures. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 588–595, 2013.

[40] Hiroaki Sakoe and Seibi Chiba. Dynamic programming algorithm optimization for spoken word recognition. *IEEE transactions on acoustics, speech, and signal processing*, 26(1):43–49, 1978.

[41] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In *Advances in neural information processing systems*, pages 568–576, 2014.

[42] Thad Starner, Joshua Weaver, and Alex Pentland. Real-time american sign language recognition using desk and wearable computer based video. *IEEE Transactions on pattern analysis and machine intelligence*, 20(12):1371–1375, 1998.

[43] Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. Videobert: A joint model for video and language representation learning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 7464–7473, 2019.

[44] Graham W Taylor, Rob Fergus, Yann LeCun, and Christoph Bregler. Convolutional learning of spatio-temporal features. In *European conference on computer vision*, pages 140–153. Springer, 2010.

[45] Alaa Tharwat, Tarek Gaber, Aboul Ella Hassanien, Mohamed K Shahin, and Basma Refaat. Sift-based arabic sign language recognition system. In *Afro-european conference for industrial advancement*, pages 359–370. Springer, 2015.

[46] Jonathan J Tompson, Arjun Jain, Yann LeCun, and Christoph Bregler. Joint training of a convolutional network and a graphical model for human pose estimation. In *Advances in neural information processing systems*, pages 1799–1807, 2014.

[47] Alexander Toshev and Christian Szegedy. Deeppose: Human pose estimation via deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1653–1660, 2014.

[48] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 4489–4497, 2015.

[49] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[50] Sy Bor Wang, Ariadna Quattoni, L-P Morency, David Demirdjian, and Trevor Darrell. Hidden conditional random fields for gesture recognition. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pages 1521–1527. IEEE, 2006.

[51] Bruce Xiaohan Nie, Caiming Xiong, and Song-Chun Zhu. Joint action recognition and pose estimation from video. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[52] Quan Yang. Chinese sign language recognition based on video sequence appearance modeling. In *2010 5th IEEE Conference on Industrial Electronics and Applications*, pages 1537–1542. IEEE, 2010.

[53] Yi Yang and Deva Ramanan. Articulated pose estimation with flexible mixtures-of-parts. In *CVPR 2011*, pages 1385–1392. IEEE, 2011.

[54] Guangle Yao, Xianyuan Liu, and Tao Lei. Action recognition with 3d convnet-gru architecture. In *Proceedings of the 3rd International Conference on Robotics, Control and Automation*, pages 208–213, 2018.

[55] Joe Yue-Hei Ng, Matthew Hausknecht, Sudheendra Vijayanarasimhan, Oriol Vinyals, Rajat Monga, and George Toderici. Beyond short snippets: Deep networks for video classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4694–4702, 2015.

[56] Zahoor Zafrulla, Helene Brashear, Thad Starner, Harley Hamilton, and Peter Presti. American sign language recognition with the kinect. In *Proceedings of the 13th international conference on multimodal interfaces*, pages 279–286, 2011.

[57] Jihai Zhang, Wengang Zhou, Chao Xie, Junfu Pu, and Houqiang Li. Chinese sign language recognition with adaptive hmm. In *2016 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6. IEEE, 2016.

[58] Hao Zhou, Wengang Zhou, and Houqiang Li. Dynamic pseudo label decoding for continuous sign language recognition. In *2019 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1282–1287. IEEE, 2019.