Cross Transferring Activity Recognition to Word Level Sign Language Detection

Srijith Radhakrishnan*[1], Nikhil C Mohan*[2], Manisimha Varma[1], Jaithra Varma[3], Smitha N Pai[1]
Manipal Institute of Technology,
Manipal Academy of Higher Education,
Manipal 576104, Karnataka, India
{srijith.radhakrishnan,nikhil.mohan2,manthena.varma,manthena.varma2}@learner.manipal.edu
smitha.pai@manipal.edu

Abstract

The lack of large scale labelled datasets in word-level sign language recognition (WSLR) poses a challenge to detecting sign language from videos. Most WSLR approaches operate on datasets that do not model real-world settings very well, as they do not have a high degree of variability in terms of signers, background, lighting and inter signer variation. We chose the MS-ASL dataset to overcome these limitations as they model open-world settings very well. This paper benchmarks successful action recognition architectures on the MS-ASL dataset using transfer learning. We have achieved new state-of-the-art accuracy (92.35%) with an improvement of 7.03% over the previous state-of-the-art introduced by the MS-ASL paper. We have analyzed how action-recognition architectures fair in the task of WSLR, and we propose SlowFast 8×8 ResNet 101 as a robust and suitable architecture for the task of WSLR.

1. Introduction

Sign language is the primary means of communication for the deaf and dumb community. American sign language (ASL) uses complex fine-grained hand gestures and facial expressions to communicate. Technological innovations on word-level sign language recognition (WSLR) can significantly help alleviate the need for human translators and lead to convenient communication between non-signers and signers. Most existing approaches rely on using additional instruments such as depth cameras [1, 2], gloves [3] or sensors [4, 5]. However, such requirements limit the ease of use in real-world settings. Deep learning methods which are non-intrusive and purely vision-based can be beneficial in such scenarios.

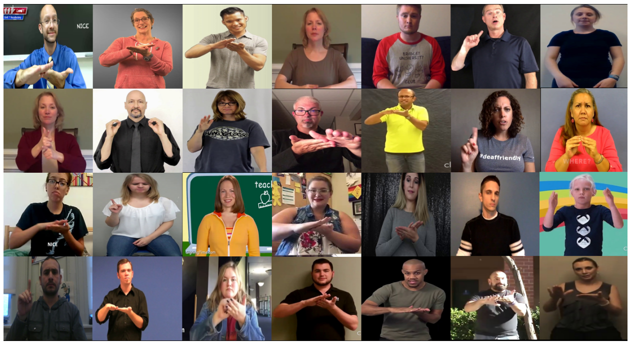


Figure 1. Illustration of the diversity of the MS-ASL dataset, which contains different signers, backgrounds, lighting and position of signers from camera.

Training deep neural networks requires huge volumes of data. The spatio-temporal nature of ASL combined with its quick gestures only makes it more essential. However, most of the existing datasets on ASL have a very limited number of instances per class and a limited number of signers as shown in Table 1. This may be due to the domain-specific knowledge required in annotating the datasets, which is labour intensive and expensive. The limited availability of training data severely restricts the scope of training and evaluating WSLR models. Most of the previous works use data with a minimal number of classes or signers generated in lab environments, due to which they might not generalize to real-world settings very well. We use the MS-ASL [6] dataset to overcome these limitations as they model real-world scenarios very well due to significant variations in view, background, lighting, and positioning as shown in Fig. 1. Moreover, the high number of signers compared to other datasets and the inter-signer variations contribute to learning more independent recognition systems that can perform well in open-world settings.

Previous deep learning based approaches use a combination of Convolutional Neural Networks (CNNs) and Re-

*They are equal contributors to this work
[1] Department of Information and Communication Technology
[2] Department of Computer Science and Engineering
[3] Department of Data Science and Computer Applications

Data-set	Class	Videos	Signers
LSA64 [7]	64	3200	10
LSE-sign [8]	2400	2400	2
GSL [9]	20	840	6
Purdue ASL [10]	104	1834	14

Table 1. Overview of word-level sign language datasets.

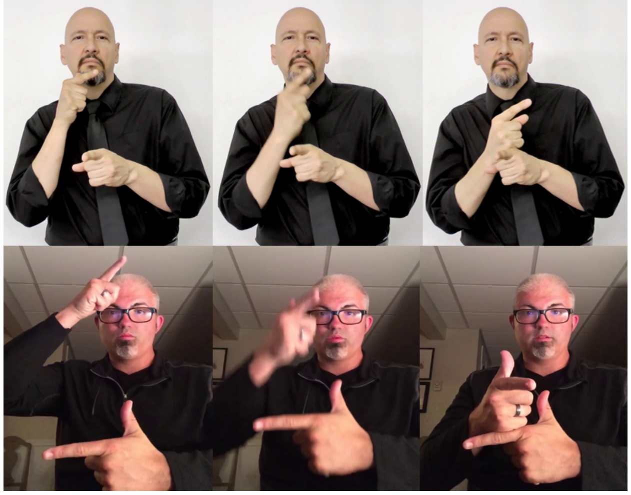


Figure 2. Extracted frames of classes 'Sister' and 'Brother'. Both the signs look similar except the hand movement begins from chin for 'sister', and from forehead for 'brother'.

current Neural Networks (RNNs) or shallow 3D-CNNs to capture the temporal information from data. To the best of our knowledge, no prior work benchmarked the latest findings from action recognition architectures on the MS-ASL dataset. With this work, we make the following contributions: (1) We evaluate action recognition architectures SlowFast [11], I3D [12], R(2+1)D [13] and P3D [14] on the MS-ASL dataset. (2) We propose SlowFast architecture as a robust and suitable architecture for sign language recognition (3) We achieved state-of-the-art accuracy on the MS-ASL dataset using the SlowFast 8×8 ResNet 101 model (4) We analyze the behaviour of different action recognition architectures on WSLR.

2. Related works

2.1. CNN + RNN

The high performance of CNNs on image data has made it an appealing choice to use it for videos. However, CNNs by themselves are incapable of recognizing temporal structure. A convenient solution to this would be to pass the frames of a video through a CNN to get their frame encodings. And then use a Recurrent Neural Network (RNN) such as LSTM to capture the long-range dependencies from the frame encodings. S. Masood *et al.* [15] used an Inception-V3 model along with LSTM network on Argentinean Sign Language. Su Yang and Qing Zhu [16] suggested a model trained using a custom convolutional neural network and an LSTM. Similarly, [17–19] proposed approaches that have leveraged the combination of CNN and RNN models.

2.2. Pose estimation models

Pose estimation have been used for human activity recognition. However, most methods do not cover hand and finger information, limiting their deployment for WSLR. Specialised approaches have been implemented to employ pose estimation for WSLR. A. Moryossef *et al.* [20] proposed an architecture that will extract optical flow features based on human pose estimation and used a linear classifier. De Coster, M. *et al.* [21] proposed a method for estimating human key points that combines feature extraction using OpenPose with end-to-end feature learning with CNN. The well-proven multi-head attention mechanism used in transformers is also employed to distinguish isolated signs in the Flemish Sign Language corpus. Similarly [22,23] have also used pose estimation for sign language recognition.

2.3. Using sensors and wearable devices

Previous works have used sensors such as Kinect to tackle WSLR. These sensors capture depth, colour, and skeletal tracking information, which are used as input to train models. S. Lang *et al.* [24] proposed a framework that takes advantage of Kinect to enable real-time 3D reconstruction, and Hidden Markov models with a constant observation density were used for recognition. García-Bautista G. *et al.* [25] collected data from the Microsoft Kinect Sensor and developed a technique to acquire hand trajectory pattern data. The hand movements were then interpreted using a Dynamic Time Warping (DTW) algorithm. Similarly, wearable devices such as gloves have also been used for sign language recognition, as seen in [26–29] . However, such methods may be intrusive and expensive for real-world applications compared to vision-based methods.

2.4. 3D-CNN

3D CNNs possess additional filters to capture temporal information. However, due to the extra kernel dimension, these models contain more parameters than 2D CNNs, making them more challenging to train on small scale datasets from scratch. Furthermore, they appear to prohibit the benefits of ImageNet pre-training. Earlier works have implemented shallow 3D CNN architectures like C3D [30] on lab generated datasets with less number of signers and samples from scratch as seen on [31–35].

3. Action recognition artitectures

Action recognition architectures require large scale datasets for training. We have implemented transfer learning on our models trained on the Kinetics 400 dataset [36]. This section provides a brief intuition of the architectures we have trained for WSLR.

3.1. P3D

The P3D [14] architecture proposes residual bottleneck blocks that decompose 3×3×3 filter size convolutions to combinations of 1×1×3 temporal and 3×3×1 spatial convolutions. This reduces the number of parameters compared to 3D CNNs, making them easier to train on small scale datasets. The P3D paper also presents a family of computationally feasible building blocks to perform 3D convolutions efficiently.

3.2. R(2+1)D

The R(2+1)D [13] architecture takes inspiration from dimension factorization by decomposing 3D convolutions into lower dimensional convolutions. It is closely related to P3D blocks in style. R(2+1)D converts the task of 3D convolution into separable spatial and temporal elements, enabling models to adapt separately to 2D spatial and 1D temporal features. This results in easier optimization as spatiotemporal filters are factorized. The R(2+1)D block increases the number of nonlinearities in the network. This is attributed to the extra activations between the 1D and 2D convolutions in each block. This allows smaller filters to map complex boundary spaces.

3.3. I3D

I3D [12] proposes to take advantage of successful architectures such as Inception [37] architecture and ResNet [38] architectures to create Spatio-temporal models by transforming them into 3D CNNs. 2D filters and pooling kernels are converted into 3D filters and pooling kernels. This captures the additional temporal dimension of videos. Square filters are made cubic, i.e. N×N filters become N×N×N by repeating the weights of the pretrained 2D filters N times along the temporal dimension and dividing by N to rescale them. Image models usually have symmetric strides along the horizontal and vertical dimensions since the spatial information is distributed uniformly across an image. However, the temporal stride should be adjusted depending upon the frame rate and image dimensions. If it extends too much in time relative to space, it may conflate frames with very different spatial information and interfere with early feature detection. While if it is too short, it may not capture scene dynamics well. To accurately represent I3D on the MS-ASL dataset, we chose the most popular 2D CNN architectures as backbones for I3D. We trained I3D InceptionV3, I3D ResNet 50 and I3D ResNet 101 models.

3.4. SlowFast

The SlowFast architecture [11] achieved state-of-the-art performance on several action recognition datasets such as Kinetics [36], Charades [39] and AVA dataset [40]. It was inspired by biological research on retinal ganglion cells in the primate visual system, 80% of the cells are composed of Parvocellular (P-cells), and 15-20% are composed of Magnocellular (M-cells). M-cells are capable of high temporal frequency and are sensitive to rapid temporal changes, but they are not sensitive to spatial detail or colour. P-cells provide high spatial and colour resolution but low temporal resolution, thus responding slowly to actions.

SlowFast is an architecture that operates on two distinct paths (Slow, Fast) with different frame rates. The Slow pathway is a CNN with a large temporal stride τ on input frames. A typical value of τ chosen for the experiment is 16, i.e. sampling 2 frames per second from a 30fps video. The primary focus of the slow pathway is to capture finer spatial semantics such as the texture, colours, and edges. This is possible due to the higher number of channels compared to the fast pathway. The ratio between the number of channels between the slow and fast pathways is denoted by β. The fast pathway captures rapidly changing characteristics by having a smaller temporal stride and a higher frame rate. In the fast pathway, more temporal resolution is utilized, The frame ratio between the slow and fast pathway is denoted by α.

The information of the two pathways is linked together via lateral connections. The connections are unidirectional and fuse the Fast pathway's features into the Slow pathway to combine the representations. Finally, on the output of each pathway, a global average pooling is conducted. The input to the fully-connected classifier layer is concatenated from two pooled feature vectors.

The intuition behind utilizing SlowFast architecture for WSLR is that when a person makes a sign with his hand, the shape of the hand does not change significantly during the gesture. However, the motion of the hand evolves at a faster rate and contains significant features for the executed gesture. As a result, the SlowFast network is an excellent alternative for addressing the sign language recognition challenge.

We have implemented SlowFast 4×16 ResNet 50, SlowFast 8×8 ResNet 50 and SlowFast 8×8 ResNet 101. The ratio of the number of channels in our model is set to be 8 in order to match the original study, i.e. $\beta = 1/8$. The frame rate ratio between the Fast and Slow paths is chosen to be 4 ($\alpha = 4$ for SlowFast 8×8) and 8 ($\alpha = 8$ for SlowFast 4×16).

Model	Accuracy(%)	Top-3(%)	Top-5(%)	#Parameters	#Frames
P3D	85.55	95.75	95.75	25M	32
R(2+1)D	88.38	91.31	96.88	53.2M	32
I3D InceptionV3	85.83	95.06	96.05	12.3M	32
I3D ResNet 50	84.70	96.60	97.45	51.9M	32
I3D ResNet 101	88.10	97.16	98.58	99.4M	32
SlowFast 4×16 ResNet 50	89.52	95.18	97.73	33.7M	32+4
SlowFast 8×8 ResNet 50	90.37	96.60	98.58	33.8M	32+8
SlowFast 8×8 ResNet 101	**92.35**	**97.73**	**98.87**	62.1M	32+8

Table 2. Top-1, top-3, top-5 accuracies achieved by each model on the MS-ASL dataset along with total number of parameters and number of frames passed as input.

4. Methodology

4.1. Data Preprocessing

The classes from MS-ASL followed a long tail distribution, as shown in Fig. 3. To construct a quality dataset, we chose the 50 classes with the highest frequency to establish our hand sign dataset. Videos from these classes were manually downloaded, screened, and trimmed to improve dataset quality. Our compiled dataset deviated from the original MS-ASL dataset as a significant amount of video links were no longer valid, rendering our dataset an even smaller subset of MS-ASL. The following observations could be made from the video samples in our dataset. (1) Some signers had slow elaborated motion while others finished the same sign within a second. (2) Several videos had repeated signs of the same class and overlapping signs from other classes. (3) Longer video clips had frames with no sign being performed at the beginning or the end of the clip. Frames with repeated signs, frames from non-target classes, and idle frames were removed. Longer videos were replaced by uniformly sampling frames from the video. Shorter samples were initially processed by resampling frames. However, on further examination, resampling frames from smaller videos corrupted the temporal information of the video. Hence, they were processed by zero-padding the videos from front. In our final dataset, each class has 48 videos on average, with a standard deviation of 6 videos over all classes. Videos were split into 85% and 15% train-test split.

Since ASL is symmetric, i.e. ASL hand signs do not pertain to specific hands and thus are the same whether performed by the left or right, including signs using both hands. A lateral inversion of videos would not affect the validity of a sample. In light of this, all videos were randomly horizontally flipped during the loading of mini-batches. Samples were also randomly scaled and cropped to improve on diversity.

Model	Accuracy
Naive Classifier	0.99
VGG+LSTM [19,41]	13.33
HCN [42]	46.08
Re-Sign [43]	45.45
I3D	81.76

Table 3. The average per class accuracy for baseline methods proposed by the MS-ASL paper [6].

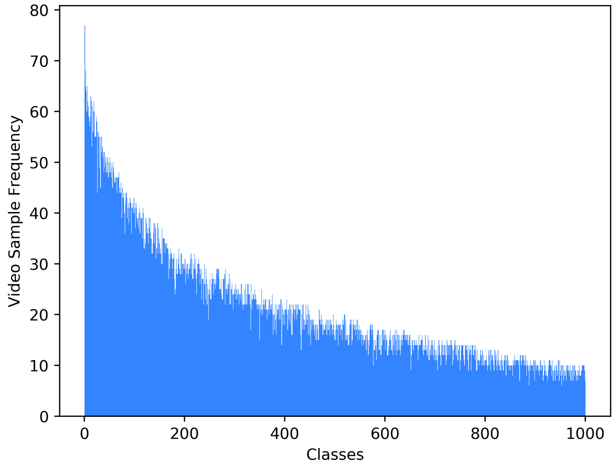


Figure 3. Distribution showing the number of video samples for each class.

4.2. Training

We used transfer learning with models pre-trained on Kinetics 400 dataset. Each architecture had different dataloader specifications. We sampled 64 consecutive frames with a fast temporal stride of 2 and a slow temporal stride of 16 for SlowFast 4×16 ResNet 50. We used a slow temporal stride of 8 for SlowFast 8×8 ResNet 50 and SlowFast 8×8 ResNet 101.

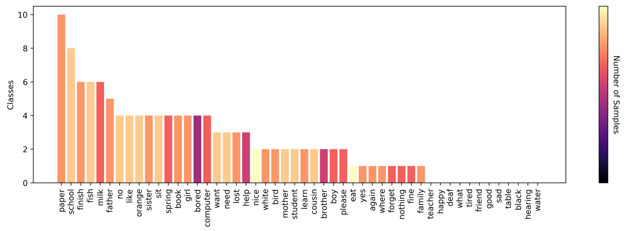


Figure 4. The bar graph shows the number of classes each class has been misclassified into, e.g. samples from 'paper' have been misclassified into ten different classes.

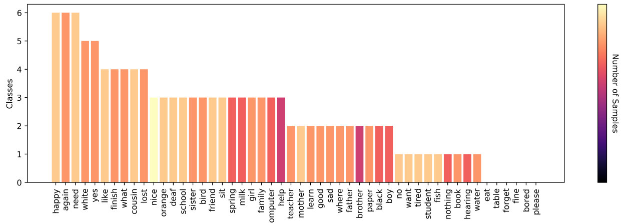


Figure 5. The bar graph shows the distribution of false positives among classes e.g. samples from six different classes have been classified as 'happy'.

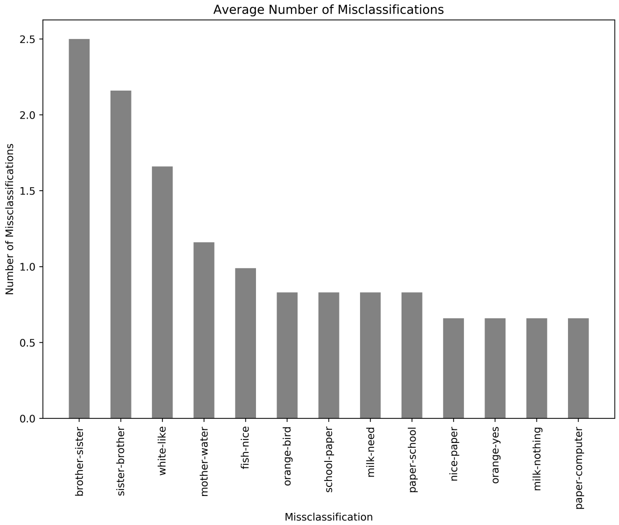


Figure 6. The graphs show the average number of times each pair is misclassified, e.g. the class 'brother' has been misclassified as 'sister' 2.5 times on average with respect to all models.

I3D, P3D and R(2+1)D uniformly sampled every other frame from 64 consecutive frames yielding 32 frames. All videos were augmented with horizontal flipping and scale ratios between (1.0, 0.8) during runtime. We randomly cropped 224×224 pixels for SlowFast, I3D, and P3D networks. R(2+1)D used input image dimensions of 112×112 pixel resolution.

We first trained the new output layer with a learning rate of $1e{-}2$ for 20 epochs. We then began unfreezing the network architecture and hyperparameter tuning until we achieved optimal performance for each model. We used a cosine scheduler within the bounds of $1e{-}2$ to $1e{-}5$ during fine-tuning for all models. We used L2 weight decay as a regularizer ranging from $1e{-}3$ to $1e{-}4$ to counter overfitting. We compiled our models using the Adam optimizer and softmax cross-entropy loss.

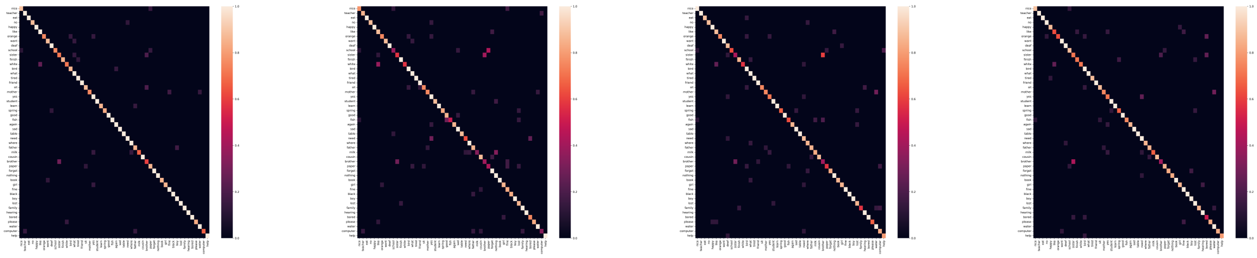
5. Results and Analysis

In this section, we compare the performance of the eight models that we have trained on the MS-ASL dataset. The test accuracy has been reported in Table 2. I3D architectures have achieved better performances by employing a ResNet backbone, as in the case of I3D ResNet 101 and I3D ResNet 50. The SlowFast 8×8 ResNet 101 has achieved a new state-of-the-accuracy 92.35% on the MS-ASL dataset with an improvement of 7.03% higher top-1 accuracy compared to the previous state of the art accuracy 85.32% introduced in the MS-ASL paper that uses the I3D Inception-V1 model. The baseline accuracies established in the MS-ASL100 dataset has been reported in Table 3.

5.1. Dataset inferences

Most models frequently misclassify pairs such as 'brother-sister', 'white-like', 'mother-water' as reported on Fig. 6. It can be observed from Fig. 2 that the gestures, 'brother' and 'sister', involve moving hands vertically from head to chest. The only variation is that, in the sign of 'brother' the hand begins from the forehead, but in the sign of 'sister' the hand begins from the chin. We believe that this high degree of similarity is the reason behind the misclassification pairs, as other pairs also represent similarities.

From Fig. 4, We observe the distribution of misclassifications of particular classes. We note that some classes have more videos that are easily mistaken for others. Samples from the class 'paper' are misclassified among ten other different classes. This is partially due to the absence of distinctive features in the sign 'paper' and due to its similarities to other classes. We observe the number of times videos are misclassified into a particular class from Fig. 5. Comparison between both figures indicates a correlation between classes with low misclassifications and classes that contribute to high false positives like the class 'happy'. Although this behaviour is expected of classes with higher samples in the dataset, we identify that the dissimilarity of signs from other signs causes this. The class 'eat', for instance, has one of the highest number of samples on the dataset and yet contributes no false positives, as its features are distinctive. Classes with high inter sign variations and a low number of samples, like the class 'computer', also



(a) SlowFast 8×8 ResNet 101

(b) I3D ResNet 101

(c) P3D

(d) R(2+1)D

Figure 7. Confusion Matrices.

display high accuracy due to their distinctive features compared to other signs.

5.2. Comparison of Architectures

Models such as I3D ResNet 50, I3D ResNet 101 and R(2+1)D often misclassify similar classes as inferred from the confusion matrices depicted in Fig. 7. However, SlowFast architectures are more robust to similar input classes, which indicates that SlowFast architectures can retain spatial information better than other models.

In action recognition, Most activities, e.g. running or swimming, repeat the motions after a point. Most of the clips on the Kinetics 400 dataset last around 10 seconds, and models with large temporal strides perform well by sampling distant frames. However, this is not the case in sign language detection. There is a high degree of variability in how quickly signers perform the signs. Most signers perform each word under a second. And unlike the classes in activity recognition, these gestures rarely repeat themselves. Hence, It is more advantageous to deploy architectures highly sensitive to temporal information in WSLR than action recognition. 3D CNNs may not be ideal for sign language as spatial and temporal dimensions are given equal importance, and variations in the temporal speed of signs can affect model performance. Architectures like R(2+1)D and P3D that decompose 3D convolutions to 2D spatial and 1D temporal convolutions work reasonably well but fall short while combining temporal and spatial information. We believe that SlowFast architecture performs better than other architecture as the high temporal sampling frequency on the fast pathway of the SlowFast architecture is able to capture highly essential temporal information very well and combine them via lateral connections to the slow stream to retain spatial information better than other models, making them highly effective for application on WSLR.

6. Conclusion

We have demonstrated that action recognition models can be successfully adapted for the task of WSLR. We evaluated state-of-the-art network architectures on the MS-ASL dataset and demonstrated that SlowFast 8×8 ResNet 101 achieves state-of-the-art-accuracy. Further, we have drawn inferences from the results.

References

[1] C. Dong, M. C. Leu, and Z. Yin, "American sign language alphabet recognition using microsoft kinect," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2015. 1

[2] Z. Zafrulla, H. Brashear, T. Starner, H. Hamilton, and P. Presti, "American sign language recognition with the kinect," in *Proceedings of the 13th International Conference on Multimodal Interfaces*, ICMI '11, (New York, NY, USA), p. 279–286, Association for Computing Machinery, 2011. 1

[3] R. Y. Wang and J. Popović, "Real-time hand-tracking with a color glove," *ACM Trans. Graph.*, vol. 28, jul 2009. 1

[4] J. Wu, L. Sun, and R. Jafari, "A wearable system for recognizing american sign language in real-time using imu and surface emg sensors," *IEEE Journal of Biomedical and Health Informatics*, vol. 20, no. 5, pp. 1281–1290, 2016. 1

[5] R.-H. Liang and M. Ouhyoung, "A real-time continuous gesture recognition system for sign language," in *Proceedings Third IEEE International Conference on Automatic Face and Gesture Recognition*, pp. 558–567, 1998. 1

[6] H. Vaezi Joze and O. Koller, "Ms-asl: A large-scale data set and benchmark for understanding american sign language," in *The British Machine Vision Conference (BMVC)*, September 2019. 1, 4

[7] F. Ronchetti, F. Quiroga, C. Estrebou, L. Lanzarini, and A. Rosete, "Lsa64: A dataset of argentinian sign language," *XX II Congreso Argentino de Ciencias de la Computación (CACIC)*, 2016. 2

[8] E. Gutierrez-Sigut, B. Costello, C. Baus, and M. Carreiras, "Lse-sign: A lexical database for spanish sign language," *Behavior Research Methods*, vol. 48, pp. 123–137, Mar 2016. 2

[9] E. Efthimiou and S.-E. Fotinea, "Gslc: Creation and annotation of a greek sign language corpus for hci," in *Proceed-*

ings of the 4th International Conference on Universal Access in Human Computer Interaction: Coping with Diversity, UAHCI'07, (Berlin, Heidelberg), p. 657–666, Springer-Verlag, 2007. 2

[10] A. Martinez, R. Wilbur, R. Shay, and A. Kak, "Purdue rvl-slll asl database for automatic recognition of american sign language," in *Proceedings. Fourth IEEE International Conference on Multimodal Interfaces*, pp. 167–172, 2002. 2

[11] C. Feichtenhofer, H. Fan, J. Malik, and K. He, "Slowfast networks for video recognition," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019. 2, 3

[12] J. Carreira and A. Zisserman, "Quo vadis, action recognition? a new model and the kinetics dataset," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4724–4733, 2017. 2, 3

[13] D. Tran, H. Wang, L. Torresani, J. Ray, Y. LeCun, and M. Paluri, "A closer look at spatiotemporal convolutions for action recognition," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6450–6459, 2018. 2, 3

[14] Z. Qiu, T. Yao, and T. Mei, "Learning spatio-temporal representation with pseudo-3d residual networks," in *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 5534–5542, 2017. 2, 3

[15] S. Masood, A. Srivastava, H. C. Thuwal, and M. Ahmad, "Real-time sign language gesture (word) recognition from video sequences using cnn and rnn," in *Intelligent Engineering Informatics* (V. Bhateja, C. A. Coello Coello, S. C. Satapathy, and P. K. Pattnaik, eds.), (Singapore), pp. 623–632, Springer Singapore, 2018. 2

[16] S. Yang and Q. Zhu, "Continuous Chinese sign language recognition with CNN-LSTM," in *Ninth International Conference on Digital Image Processing (ICDIP 2017)* (C. M. Falco and X. Jiang, eds.), vol. 10420, pp. 83 – 89, International Society for Optics and Photonics, SPIE, 2017. 2

[17] K. Bantupalli and Y. Xie, "American sign language recognition using deep learning and computer vision," in *2018 IEEE International Conference on Big Data (Big Data)*, pp. 4896–4899, 2018. 2

[18] N. Basnin, L. Nahar, and M. S. Hossain, "An integrated cnn-lstm model for bangla lexical sign language recognition," in *Proceedings of International Conference on Trends in Computational and Cognitive Engineering* (M. S. Kaiser, A. Bandyopadhyay, M. Mahmud, and K. Ray, eds.), (Singapore), pp. 695–707, Springer Singapore, 2021. 2

[19] R. Cui, H. Liu, and C. Zhang, "Recurrent convolutional neural networks for continuous sign language recognition by staged optimization," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1610–1618, 2017. 2, 4

[20] A. Moryossef, I. Tsochantaridis, R. Aharoni, S. Ebling, and S. Narayanan, "Real-time sign language detection using human pose estimation," in *ECCV Workshops*, 2020. 2

[21] M. De Coster, M. Van Herreweghe, and J. Dambre, "Sign language recognition with transformer networks," in *Proceedings of the 12th Language Resources and Evaluation Conference*, (Marseille, France), pp. 6018–6024, European Language Resources Association, May 2020. 2

[22] H. Hu, W. Zhou, and H. Li, "Hand-model-aware sign language recognition," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, pp. 1558–1566, May 2021. 2

[23] S.-K. Ko, C. J. Kim, H. Jung, and C. Cho, "Neural sign language translation based on human keypoint estimation," *Applied Sciences*, vol. 9, no. 13, 2019. 2

[24] S. Lang, M. Block, and R. Rojas, "Sign language recognition using kinect," in *Artificial Intelligence and Soft Computing* (L. Rutkowski, M. Korytkowski, R. Scherer, R. Tadeusiewicz, L. A. Zadeh, and J. M. Zurada, eds.), (Berlin, Heidelberg), pp. 394–402, Springer Berlin Heidelberg, 2012. 2

[25] G. García-Bautista, F. Trujillo-Romero, and S. O. Caballero-Morales, "Mexican sign language recognition using kinect and data time warping algorithm," in *2017 International Conference on Electronics, Communications and Computers (CONIELECOMP)*, pp. 1–5, 2017. 2

[26] S. Y. Heera, M. K. Murthy, V. S. Sravanti, and S. Salvi, "Talking hands — an indian sign language to speech translating gloves," in *2017 International Conference on Innovative Mechanisms for Industry Applications (ICIMIA)*, pp. 746–751, 2017. 2

[27] Q. Zhang, D. Wang, R. Zhao, and Y. Yu, "Myosign: Enabling end-to-end sign language recognition with wearables," in *Proceedings of the 24th International Conference on Intelligent User Interfaces*, IUI '19, (New York, NY, USA), p. 650–660, Association for Computing Machinery, 2019. 2

[28] N. Tubaiz, T. Shanableh, and K. Assaleh, "Glove-based continuous arabic sign language recognition in user-dependent mode," *IEEE Transactions on Human-Machine Systems*, vol. 45, no. 4, pp. 526–533, 2015. 2

[29] S. Mehdi and Y. Khan, "Sign language recognition using sensor gloves," in *Proceedings of the 9th International Conference on Neural Information Processing, 2002. ICONIP '02.*, vol. 5, pp. 2204–2206 vol.5, 2002. 2

[30] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3d convolutional networks," in *2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 4489–4497, 2015. 2

[31] J. Huang, W. Zhou, H. Li, and W. Li, "Sign language recognition using 3d convolutional neural networks," in *2015 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1–6, 2015. 2

[32] Z.-j. Liang, S.-b. Liao, and B.-z. Hu, "3D Convolutional Neural Networks for Dynamic Sign Language Recognition," *The Computer Journal*, vol. 61, pp. 1724–1736, 05 2018. 2

[33] S. Sharma and K. Kumar, "Asl-3dcnn: American sign language recognition technique using 3-d convolutional neural networks," *Multimedia Tools and Applications*, vol. 80, pp. 1–13, 07 2021. 2

[34] D. K. Singh, “3d-cnn based dynamic gesture recognition for indian sign language modeling,” *Procedia Computer Science*, vol. 189, pp. 76–83, 2021. AI in Computational Linguistics. 2

[35] M. Al-Hammadi, G. Muhammad, W. Abdul, M. Alsulaiman, M. A. Bencherif, and M. A. Mekhtiche, “Hand gesture recognition for sign language using 3dcnn,” *IEEE Access*, vol. 8, pp. 79491–79509, 2020. 2

[36] W. Kay, J. Carreira, K. Simonyan, B. Zhang, C. Hillier, S. Vijayanarasimhan, F. Viola, T. Green, T. Back, P. Natsev, M. Suleyman, and A. Zisserman, “The kinetics human action video dataset,” *CoRR*, vol. abs/1705.06950, 2017. 3

[37] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, “Going deeper with convolutions,” in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–9, 2015. 3

[38] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016. 3

[39] G. A. Sigurdsson, G. Varol, X. Wang, A. Farhadi, I. Laptev, and A. Gupta, “Hollywood in Homes: Crowdsourcing Data Collection for Activity Understanding,” in *Computer Vision – ECCV 2016*, (Amsterdam, Netherlands), pp. 510 – 526, Oct. 2016. 3

[40] C. Gu, C. Sun, D. A. Ross, C. Vondrick, C. Pantofaru, Y. Li, S. Vijayanarasimhan, G. Toderici, S. Ricco, R. Sukthankar, C. Schmid, and J. Malik, “Ava: A video dataset of spatio-temporally localized atomic visual actions,” in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6047–6056, 2018. 3

[41] R. Cui, H. Liu, and C. Zhang, “A deep neural framework for continuous sign language recognition by iterative training,” *IEEE Transactions on Multimedia*, vol. 21, no. 7, pp. 1880–1891, 2019. 4

[42] W. Zhu, C. Lan, J. Xing, W. Zeng, Y. Li, L. Shen, and X. Xie, “Co-occurrence feature learning for skeleton based action recognition using regularized deep lstm networks,” in *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, AAAI’16, p. 3697–3703, AAAI Press, 2016. 4

[43] O. Koller, S. Zargaran, and H. Ney, “Re-sign: Re-aligned end-to-end sequence modelling with deep recurrent cnn-hmms,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3416–3424, 2017. 4