A Small Model for Big Articulators: Sign Language Detection With a Tiny Machine Learning Model

Frederick Chan, Gina-Anne Levow, Qi Cheng
University of Washington
Seattle, WA
fredchan@uw.edu, levow@uw.edu, qicheng2@uw.edu

Abstract

This paper introduces a small (1,013 parameter) machine learning model for sign language detection in videos of isolated American Sign Language (ASL) signs. Our model aims to alleviate the time-consuming nature of producing sign clips for psycholinguistic study stimuli, sign dictionaries, and sign databases. Given a video where the signer starts from a resting position, signs a sign, and returns to the resting position for an arbitrary number of repetitions, the model detects frames in which signing occurs that can be used to segment video into clips of individual signs. We train and evaluate our model on data with precise coding of signing onset and offset from ASL-LEX 2.0, so that our model's annotations are suitable for psycholinguistics research. The model works on both real signs and pseudosigns, two types of stimuli needed for certain psycholinguistic studies. Our model's small size compared to the state-of-the-art (100K parameters or more) enables quick, bulk processing even on resource-constrained hardware. It achieves this by computing Instantaneous Visual Change (IVC), a 1D measure of changes in brightness in the input video, extracting features from the IVC-over-time signal with a convolution, and classifying the video frames as signing or non-signing with three neural layers.

Keywords: sign language detection, sign language segmentation, machine learning, small models, convolutional neural networks (CNN)

1. Introduction

Sign languages are natural languages used by deaf and hard-of-hearing people that make use of both manual and non-manual movements, including posture and facial modulations (Sandler and Lillo-Martin, 2006). Like spoken languages, they have complex systems of phonology, grammar, and lexicon, but their use of the visual modality gives rise to many unique linguistic features that can prove challenging for machine processing (Sandler and Lillo-Martin, 2006; Toshpulatov et al., 2025). Development in sign language processing has lagged significantly behind spoken language processing, partly due to the complex nature of the phonology of sign languages, the difficulty of working with videos as opposed to audio, and a lack of high-quality, large-scale data sets. (Toshpulatov et al., 2025; Neidle et al., 2012).

In the past decade, the application of neural networks in other fields and increased availability of computing capacity have led to an explosion of neural network-based approaches to sign language tasks (Toshpulatov et al., 2025), which are able to handle the size and complexity of sign language video data. Since neural networks tend to benefit from larger parameter sizes, state-of-the-art sign language models tend to feature upwards of 100K parameters or more (Moryossef et al., 2020; Borg and Camilleri, 2019), which requires significant computational power and runtime.

In this paper, we demonstrate that small neural networks (around 1K parameters), can be used to detect signing in a kind of signing video commonly created for psycholinguistic studies and various sign language resources, like dictionaries.

A small model could also be added into existing sign language annotation tools, such as ELAN (Wittenburg et al., 2006), thus integrating easily into existing workflows.

1.1. Sign Language Detection and Segmentation

In an automatic sign language detection task, a video frame is given, with or without surrounding frames, to a model which labels it as signing or non-signing. By classifying all the frames of a video, these labels can be used to segment it into time spans in which signing occurs.

Larger, state-of-the-art sign language models perform well on complex, continuous signing videos, where signers produce phrases or sentences, often as they would in normal discourse (Stassi et al., 2025; Borg and Camilleri, 2019; Moryossef et al., 2020, 2023). In this paper, we focus on detecting sign language in a simpler type of video that is produced when creating psycholinguistics research stimuli (Emmorey et al., 2011; Gu et al., 2022; Caselli et al., 2021), sign language dictionaries, and sign databases (Sehyr et al., 2021). That is, videos where a signer in front of a static background begins in a resting position, signs a sign, and moves back to the resting position, an

arbitrary number of times. These videos are then edited down into clips, each containing one sign, for use in the final product or study.

Despite the simple nature of these videos, the process of editing the video into individual sign clips remains a major pain point. A psycholinguistic study may make use of dozens or hundreds of such clips, while a dictionary may need thousands. Producing a whole set takes many hours of work, and a means of clipping these videos quickly and without needing an army of undergraduates to do the grunt work would be ideal. The model we demonstrate in this paper aims to fill this need.

Since such videos are simpler compared to continuous signing, certain factors that are common in natural discourse but could confound a computer model, such as phonological assimilation between signs and simultaneous morphology, are not present to confuse the model. We show that a small, simple model is sufficient to achieve good performance on this simpler domain.

1.2. Definition of Signing Onset and Offset

Since part of the goals of our model is to be useful for creating stimuli for psycholinguistics, we need a precise definition for the onset and offset of a sign. Unlike spoken language, where the articulators are hidden in the vocal tract and the onset of the audio is the same as the onset of the word, motion begins before the sign onset, since the arms and hands must move into position before producing the sign (Emmorey et al., 2022).

In this paper, we use the definitions from in ASL-LEX 2.0 (Sehyr et al., 2021). For signs with body contact and two-handed signs, the onset is the first video frame where the fully-formed handshape has contacted the body, and the offset is the last frame where the hand has contacted the body. For signs without contact, the onset is the first frame where the fully-formed handshape has arrived at the target location near the body or in neutral space, and the offset is the last frame before the hand(s) have begun to transition to a resting position (Sehyr et al., 2021).

Using these definitions allows researchers to precisely control for the onset of a sign when measuring brain activity while a subject comprehends sign language stimuli. For example, Emmorey et al. (2022) looks for specific Event Related Potentials (ERPs), such as N400 priming effects, relative to the sign onset. Studies involving reaction time, such as Caselli et al. (2021) and Sehyr and Emmorey (2022), also need control for the signing onset to get accurate and comparable data between trials.

To serve these needs, we want as little error between our model's predicted sign onsets and the true sign onsets as possible. We therefore made use of two video corpora, ASL-LEX 2.0 and the stimuli videos from Caselli et al. (2021) (described further in Section 3.1), that have rigorously annotated sign onsets and offsets based on the definitions previously described.

1.3. Existing Models

Presently, the state-of-the-art in sign language detection and segmentation achieves high accuracy on the continuous, naturalistic videos. However, they are also quite large, which costs a considerable amount of computational power to run. Furthermore, the onsets and offsets in the training data of these models are not defined as precisely as the datasets we use here, and their suitability for the preparation of stimuli for psycholinguistics is unknown.

Borg and Camilleri (2019) created sign language segmentation models that extract features from raw video data, optical flow, motion history, and frame difference using the pretrained VGG-16 Convolutional Neural Network (Simonyan and Zisserman, 2015), which then classifies frames as signing and non-signing with a Recurrent Neural Network (RNN). The smallest of these models uses 3.3 million parameters, and classifies 20 frames of a 5 fps video in upwards of 3 seconds on CPU.

Moryossef et al. (2020) uses an RNN to classify frames using only the position of the joints of the signer as input, which are extracted from the raw video using the pose estimation model, OpenPose (Cao et al., 2019). This allows the model to ignore irrelevant motion in the video and focus on the major articulators involved in signing. While much smaller than Borg and Camilleri (2019), this model involves another neural network as a preprocessing step and still uses 102 thousand parameters.

Moryossef et al. (2023) proposes several models using joint positions extracted from MediaPipe (Lugaresi et al., 2019) combined with other techniques to improve segmentation performance. These include using optical flow features, 3D hand normalization, and location of nonmanual keypoints. The smallest of these models uses 454 thousand parameters.

He et al. (2025) uses two pretrained models that extract hand mesh and skeleton pose features from video, and trains two multi-layer perceptrons, a feature mixer, and a Transformer encoder within the same model to classify video frames from those features. The total parameter count cannot be inferred from the paper as only the feature sizes but not the layer sizes of the model are not reported, nor is the model's code published at the time of writing. However, given its complexity, it is reasonable to assume that the model is much larger than

1,013 parameters.

1.4. Instantaneous Visual Change (IVC)

The model introduced in this paper is able to achieve results with much smaller parameter sizes by using a simple 1-dimensional signal extracted from the input video called *Instantaneous Visual Change* (IVC). IVC is the sum of squared differences in each pixel across sequential frames of video:

$$IVC(t) = \sum_i \left([\, x_i\,(\,t\,) - x_i\,(\,t-1\,)\,] \right)^2$$

where x is the grayscale value of pixel i at time t.

IVC was introduced in Brookshire et al. (2017), which established a correlation between the quasiperiodic nature of the IVC-over-time signal in sign language videos with neural entrainment in the visual cortex of native signers who view them. The IVC signal is comparable to the spoken audio signal, with the spectral power densities of both being similar (Brookshire et al., 2017). The authors hypothesize that babies can "tune in" to the presence of signing in visual stimuli during language acquisition, because the quasiperiodicity is characteristic of a linguistic signal.

Given this, it stands to reason that a machine could also exploit this property of the IVC signal to detect the presence of signing in a video. In our model, we train a convolutional layer, which is popular for shape and object detection in computer vision, to learn shapes in the IVC signal that are characteristic of signing. Then, it classifies frames into signing or non-signing based on the presence or absence of these shapes.

2. Model Architecture

2.1. Pre-processing

First, each video is downsampled to 29.93 frames per second, the frame rate of the videos in ASL-LEX 2.0 and the Caselli et al. (2021) study.

Next, the IVC of the frames of the input video is computed using a version of Brookshire (2017)'s IVC module that we modified to work in Python 3. This module converts the color space to grayscale using OpenCV's (Bradski, 2000) $RGB \leftrightarrow GRAY$ color conversion mode (OpenCV team, 2025). Note that the first frame of the video has an IVC of 0; no video frame precedes it, so there is no visual change.

Then, the framewise IVC is normalized by dividing the IVC by the standard deviation of all the IVC in the video. This reduces the effect of variations in the physiology of the signer, distance from the camera, and video resolution. No denoising or resizing of the video frames themselves are performed.

A sliding window across all the frames of the video is then computed, creating windows that are 16 frames long (about 0.54 seconds). For each window, the model predicts the probability that the middle frame (frame 8) contains signing. This gives the model 7 frames of past IVC history, the IVC of the frame being classified, and the IVC of the 8 frames that follow.

2.2. The Model

The model starts with a feature extractor: a single 1D convolutional layer of 4 kernels, with a width of 4 frames, and a stride of 1. This feature extractor learns to detect shapes in the IVC signal that are characteristic of signing.

The convolution produced by the 4 kernels are then concatenated into a single feature vector. This feature vector is fed into a neural network of 3 densely connected neural layers:

- Layer 1: 16 neurons with ReLU activation
- Layer 2: 8 neurons with ReLU activation
- Regression: 1 neuron with sigmoid activation

The regression layer outputs the middle frame's probability of signing. During inference, frames with $\geq$50% probability are considered signing frames.

In total, the model consists of 1,013 parameters.

2.3. Hyperparameter Selection

The hyperparameters of this model were chosen using the GridSearch method in KerasTuner (O'Malley et al., 2019), over the settings below. All possible configurations within the search space were tested on a limited set of training data of 270 videos, and the hyperparameters of the best performing model were used to train a randomly initialized model on the full training set. The search space is as follows:

- Number of convolutional kernels: 1, 2, 4, 8, 16
- Convolutional kernel size: 4, 8, 16
- Number of ReLU dense layers: 0, 1, 2
- Size of ReLU dense layers: 4, 8, 16, 32

The results of the hyperparameter search are shown in Figure 1.

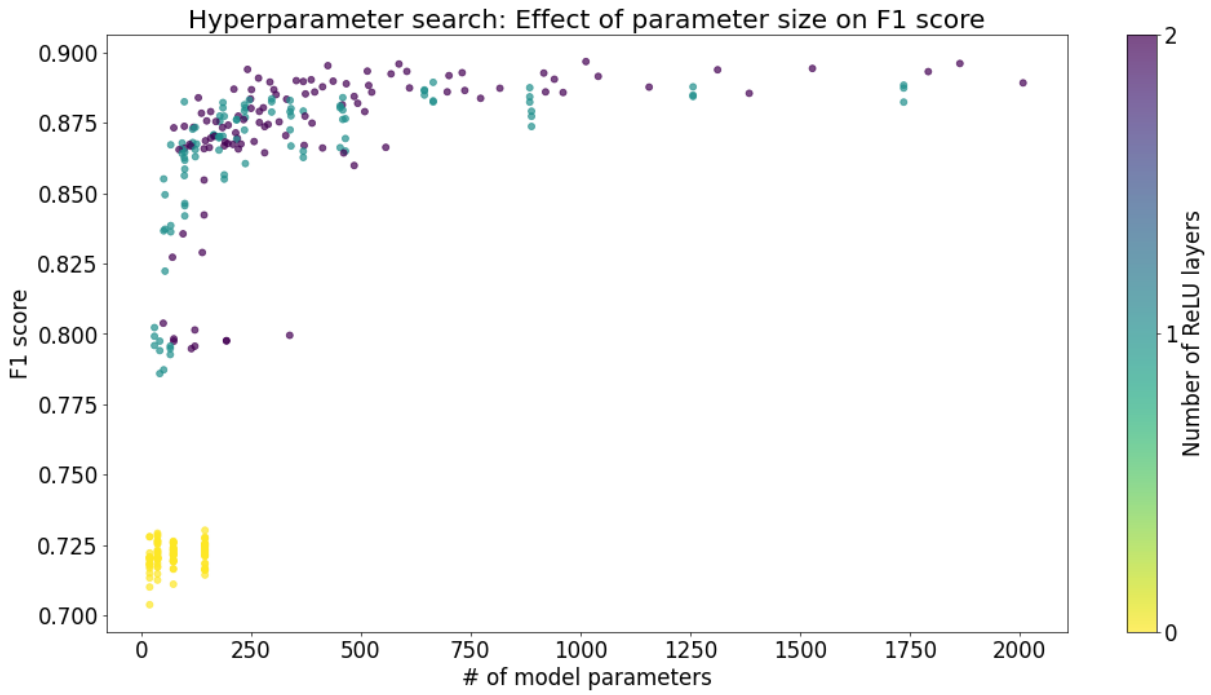

Figure 1: Results of the hyperparameter grid search. A number of different small-model hyperparameter configurations were tested on a limited set of training data (270 videos), before the best performing model was retrained on the full set. The hyperparameters in the search varied by number of convolutional kernels, kernel size, number of ReLU layers, and ReLU layer size. The best models use 2 ReLU layers of varying size. The cluster on the bottom left are models with no ReLU layers, only convolutions and a logistic regression.

3. Training

3.1. Training Data

The videos used to train the model were originally created for ASL-LEX 2.0 (Sehyr et al., 2021), a large-scale database of ASL signs that also contains the signing onset and offset times for each sign video. The ASL-LEX 2.0 set consists of 2,792 videos of unique single signs, featuring the same deaf native signer (female, middle-aged, White, born in the North-East USA, resides in California) filmed from the front and from the waist up, against a static blue background. The videos appear to have been filmed in different batches; while many videos have identical camera and lighting conditions, this varies between different batches, and the signer may have different outfits between them. Onsets and offsets for each video were annotated by two independent, trained coders, with 90.4% agreement for onsets and 99% agreement for offsets. The minimum non-normalized IVC in this dataset is 0, with a maximum of 4,246,152,807. From this dataset, we set aside 20% of the videos to use as a validation set.

For the test set, 569 stimulus videos that were created for the Caselli et al. (2021) psycholinguistics study were used. This set contains 276 unique real ASL signs and 293 ASL-based pseudosigns, which were created by changing one or two phonological parameters *(e.g. handshape, location, movement)* of a real sign so that it is not a sign in ASL. The signer in these videos is a different deaf, native signer from the one in the ASL-LEX 2.0 videos, and was filmed against a static white background in slightly different lighting conditions owing to the time of day. The minimum non-normalized IVC in this dataset is 0, with a maximum of 475,561,711.

In the videos of both datasets, the signer starts from a neutral, sitting position before beginning the sign. They produce the sign, then return to a neutral position. Since the neutral position and the transitions to and from signing do not carry lexical meaning, frames containing these are considered non-signing.

These two datasets were selected because they were among the only easily-available corpora of isolated sign videos with high-quality timing annotations necessary for training and evaluation.

Both datasets undergo the pre-processing described in Section 2.1, creating a large number of frame windows to classify. Then, windows are randomly dropped until there is an equal number of signing and non-signing windows in the split. Balancing the signing and non-signing examples

improves its ability to classify both classes, rather than considering one class the "default," with worse performance. This resulted in 111,968 frame windows to train on.

3.2. Hardware, Framework, and Schedule

We trained the model on a batch size of 32, using the Adam optimizer for 100 epochs, by which time the validation accuracy has stopped showing significant improvement. Each epoch took about 30 seconds each, for a total of 50 minutes.

The model was trained on an NVIDIA GeForce 3070Ti Laptop GPU. The Keras neural network library was used with the PyTorch backend.

4. Evaluation

The evaluation metrics in this section can also be seen in Table 1. As a performance baseline we could directly compare to, we also trained and evaluated a logistic regression with no convolution created with scikit-learn (Pedregosa et al., 2011), using the same training and test data preprocessed with the procedure in Section 2.1.

4.1. Frame-wise Performance

The frame-wise performance metrics were computed after balancing the number of signing and non-signing windows, as described in Section 3.1. This results in 28,494 windows to classify in the validation set and 11,480 windows in the test set, each with an equal number of signing and non-signing examples.

On the validation set, the model achieved an Accuracy/Precision/Recall/F1 of 87/84/91/88%. On the test set, it achieved 87/85/88/87%.

Among pseudosigns in the test set, it achieved 85/85/86/86%, compared with real signs where it achieved 88/85/92/88%.

Our model achieves better performance than the logistic regression on both the test (F1 of 88% > 71%, 17 percentage points better) and validation sets (F1 of 86% > 70%, 16 percentage points better).

4.2. Video-wise Performance

To ascertain how well the model at segmenting videos, instead of just classifying frames, we evaluated the model on whole videos using Intersection over Union and Segment Ratio. Unlike in the frame-wise performance section, no windows were dropped. Thus, the model classifies all the frames of each video.

4.2.1. Intersection Over Union (IOU)

Intersection Over Union (IOU) is a measure of the similarity between two sets A and B, using the formula:

$$IOU(A, B) = \frac{|A \cap B|}{|A \cup B|}$$

Here, we are comparing the the ground truth set of signing frames in a video against the model's predicted set. IOU is always in the interval [0, 1], and higher scores are better, indicating higher similarity.

In this paper, we compute the IOU between the prediction and the ground truth for each video in the test set, then take the mean IOU over all the videos.

On the test set, the mean IOU was 0.78 (σ=0.13). Among real sign videos in the test set, the mean IOU was 0.79 (σ=0.14), compared with pseudosign videos with a mean IOU of 0.77 (σ=0.13). Figure 2 shows the segmentation output for three sign videos in the test set at the best, median, and worst case IOU.

On the validation set, the mean IOU was 0.80 (σ=0.17).

4.2.2. Segment Ratio

Segment Ratio (Seg) is a measure of how well the number of segments S_{pred} predicted by a model matches the actual number of segments S_{true} in the video. It is computed using the formula:

$$Seg(S_{pred}, S_{true}) = \frac{S_{pred}}{S_{true}}$$

Ideally, Seg is 1, with a higher number indicating over-segmentation, and a lower number indicating under-segmentation.

In this paper, we compute the Seg between the prediction and the ground truth for each video in the test set, then take the mean Seg over all the videos.

On the test set, the mean Seg was 1.43 (σ=0.76). Among real sign videos in the test set, the mean Seg was 1.45 (σ=0.78), compared with pseudosign videos with a mean Sig of 1.41 (σ=0.74).

On the validation set, the mean Seg was 1.52 (σ=0.89).

4.3. Onset Error

To see how well the model aligns the onset of the predicted segments with the actual onsets of the sign, we computed Onset Error (OE) on videos in the test and validation set with the formula:

$$OE(O_{pred}, O_{true}) = abs(O_{pred} - O_{true})$$

Where O_{true} is the frame number of the true onset of the sign in the video, and O_{pred} is the

	Frame-level				Video-level	
	Accuracy	**Precision**	**Recall**	**F1**	**IOU**	**Seg**
Our model						
Validation set	87%	84%	91%	88%	.80	1.52
Test set	87%	85%	88%	86%	.78	1.43
Real signs	88%	85%	92%	88%	.79	1.45
Pseudosigns	85%	85%	86%	86%	.77	1.41
Concatenated	86%	80%	88%	84%	.72	2.09
IVC + Logistic Regression						
Validation set	70%	70%	72%	71%	.65	1.38
Test set	70%	69%	70%	70%	.64	1.49
Concatenated	71%	64%	70%	67%	.50	2.05
State-of-the-art (Continuous signing, not a direct comparison)						
Bull et al. (2020) ("body" model)					.69	2.96
Moryossef et al. (2023) (E0 model)					.77	1.37
He et al. (2025)				86%	.76	0.98

Table 1: Performance metrics for our small model, compared with a simple logistic regression. State-of-the-art models are also shown for **indirect** comparison (see Section 5 for caveats). Metrics for Bull et al. (2020) and Moryossef et al. (2023) are from Stassi et al. (2025), which evaluated both models using a common dataset and methodology. Metrics for the model in He et al. (2025) are from its own paper. Note that He et al. (2025) is a BIO tagger that classifies three labels, unlike our binary classifier, and that this may affect the F1 score comparison.

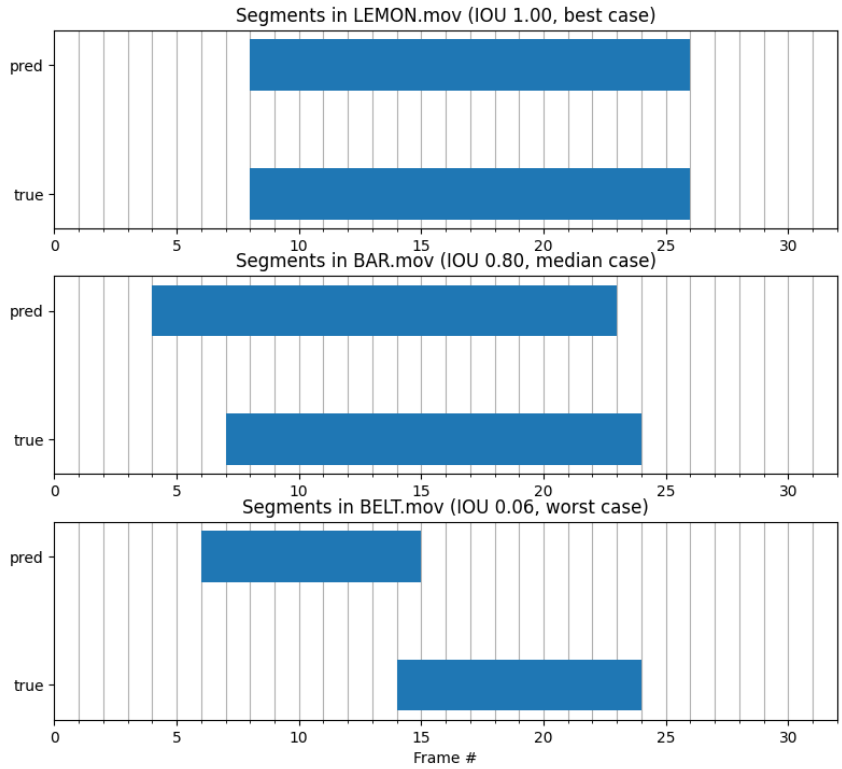

Figure 2: Predicted versus ground truth segments in three videos in the test set. Top: best case IOU (1.00) for a video containing the ASL sign LEMON, where the model predicts the segment perfectly. Middle: median case IOU (0.80) for BAR, with 4 frames of difference. Bottom: worst case IOU (0.06) for BELT, with little overlap.

frame number of the onset of the first predicted segment in the video.

On the test set, the mean OE was 2.82 frames (σ=2.97), or about 0.09s (σ=0.1s). The mean OE for real signs in the test set was 2.86 frames (σ=3.34), and 2.77 frames (σ=2.58) for pseudosigns.

On the validation set, the mean OE was 2.34 frames (σ=3.76), or about 0.08s (σ=0.12s).

If we compute OE without the absolute value, we can see if the model tends to predict the onset too early or too late. On average, the model predicts onset 2.5 frames earlier than the ground truth in the test set, and 1.05 frames earlier in the validation set.

4.3.1. Concatenated Long Videos

Although neither the training nor the test data include long-form videos for segmentation, we can simulate these videos by concatenating the IVC of randomly selected clips from the test set, normalizing the concatenated IVC as if it were one video, and running the model. 500 concatenated videos were tested this way, each consisting of a random number of clips, ranging from 10 to 100 videos.

We chose to concatenate the IVC instead of editing the video clips together and calculating the IVC afterwards, as the latter would create artifacts between clips that would not appear in a long video recorded in a single take. Namely, the transition between clips filmed in different conditions would create a frame with an unusually high IVC, since most of the pixels in the scene would be different. Similarly, we dropped the first frame of each video, starting from the second video in each concatenation, to remove 0-IVC frames that appear simply because it is the first frame.

On the set of 500 concatenated videos, the mean IOU was 0.72 (σ=0.02), and the mean Seg was 2.09 (σ=0.19).

4.4. Runtime

When running inference on the same hardware as training, pre-processing of the entire test set takes about 3 minutes and 40 seconds, and the model predicts the labels in about 6.6 seconds, for a total of 3 minutes and 46.6 seconds.

5. Conclusion

In this paper we demonstrate that, when the problem domain is limited to detecting signing in a stream of single signs, a tiny model of only 1,013 parameters is able to achieve an F1 score of 88% and a mean IOU of 0.78 on videos of single signs and an F1 score of 84% and a mean IOU of 0.72 on simulated videos of multiple isolated signs. Using logistic regression as a base of comparison, our model has significantly better F1 and IOU, showing that the model architecture yields meaningful improvements over a simple baseline.

We can compare our metrics to state-of-the-art sign language segmentation models. According to a paper that evaluated both models on the same task, Stassi et al. (2025) finds that the IOU of the best performing model from Bull et al. (2020) achieves an IOU of .69, and Moryossef et al. (2023) achieves an IOU of .77. Note that this is not a direct comparison; the state-of-the-art models were evaluated on continuous signing videos, which are more complex. The purpose of the comparison is to see whether our small model works as well on the simpler task as a complex model works on a complex task. Since our score appears within this range, it seems that our small model is able to do this successfully.

6. Limitations

While the datasets used for training and evaluating this model present good coverage for signs with different lexical and phonological properties (*e.g.* parts of speech, movement types, sign location) the fact that there are so few people represented in the data means that we do not know how well the model generalizes to signers of different ethnicity, ages, body shapes, etc,. The fact that the model achieves similarly high accuracy and F1 scores for the validation set as the test set, which do not have any signer overlap, is a good sign, but it is still far from rigorous.

If precise signing onset and offset annotations were available for a dataset like ASL Citizen (Desai et al., 2023), which features isolated sign videos crowdsourced from many Deaf and Hard of Hearing signers, such a dataset would be ideal for training and evaluating the model. Unfortunately, we have so far been unable to source such a similar dataset of good quality.

Since the sign language segmentation task can apply to other sign languages also, sign corpora from languages other than ASL should be considered for the future.

7. Future Work

The IVC signal alone is insufficient to capture the full complexity of the movements of sign language. Fundamentally, it is a measure of the amount of change in the visual field, analogous to volume in a spoken language's audio signal, so there is a limit to the amount of information that can be recovered from the original signing.

Although pose estimation introduces a lot of complexity and runtime to the process, making direct

use of the position and motion of the articulators involved in sign language (*e.g.* arms, hands, face, torso) could allow a model to pick up on more nuanced information. With MediaPipe (Lugaresi et al., 2019) now available even in a web browser, a small classifier that uses pose estimation features may be practical on midrange consumer hardware.

8. Funding

Research reported in this publication was supported by the National Institute for Deafness and Other Communication Disorders under Award Number R21DC022046, and the University of Washington Linguistics Fund. The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Institutes of Health or the University of Washington.

9. Bibliographical References

Mark Borg and Kenneth P Camilleri. 2019. Sign language detection "in the wild" with recurrent neural networks. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1637–1641. IEEE.

Gary Bradski. 2000. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*.

Geoffrey Brookshire. 2017. ivc: a small python module to compute the instantaneous visual change of videos. https://github.com/gbrookshire/ivc/tree/master. Accessed: 2026-03-22.

Geoffrey Brookshire, Jenny Lu, Howard C Nusbaum, Susan Goldin-Meadow, and Daniel Casasanto. 2017. Visual cortex entrains to sign language. *Proceedings of the National Academy of Sciences*, 114(24):6352–6357.

Hannah Bull, Michèle Gouiffès, and Annelies Braffort. 2020. Automatic segmentation of sign language into subtitle-units. In *European Conference on Computer Vision*, pages 186–198. Springer.

Z. Cao, G. Hidalgo Martinez, T. Simon, S. Wei, and Y. A. Sheikh. 2019. Openpose: Realtime multi-person 2d pose estimation using part affinity fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Naomi K Caselli, Karen Emmorey, and Ariel M Cohen-Goldberg. 2021. The signed mental lexicon: Effects of phonological neighborhood density, iconicity, and childhood language experience. *Journal of Memory and Language*, 121:104282.

Aashaka Desai, Lauren Berger, Fyodor Minakov, Nessa Milano, Chinmay Singh, Kriston Pumphrey, Richard Ladner, Hal Daumé III, Alex X Lu, Naomi Caselli, et al. 2023. ASL citizen: a community-sourced dataset for advancing isolated sign language recognition. *Advances in Neural Information Processing Systems*, 36:76893–76907.

Karen Emmorey, Katherine J Midgley, and Phillip J Holcomb. 2022. Tracking the time course of sign recognition using erp repetition priming. *Psychophysiology*, 59(3):e13975.

Karen Emmorey, Jiang Xu, and Allen Braun. 2011. Neural responses to meaningless pseudosigns: evidence for sign-based phonetic processing in superior temporal cortex. *Brain and Language*, 117(1):34–38.

Shengyun Gu, Deborah Chen Pichler, L Viola Kozak, and Diane Lillo-Martin. 2022. Phonological development in american sign language-signing children: Insights from pseudosign repetition tasks. *Frontiers in psychology*, 13:921047.

Low Jian He, Harry Walsh, Ozge Mercanoglu Sincan, and Richard Bowden. 2025. Hands-on: Segmenting individual signs from continuous sequences. In *2025 IEEE 19th International Conference on Automatic Face and Gesture Recognition (FG)*, page 1–5. IEEE.

Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Uboweja, Michael Hays, Fan Zhang, Chuo-Ling Chang, Ming Yong, Juhyun Lee, et al. 2019. Mediapipe: A framework for perceiving and processing reality. In *Third workshop on computer vision for AR/VR at IEEE computer vision and pattern recognition (CVPR)*, volume 2019.

Amit Moryossef, Zifan Jiang, Mathias Müller, Sarah Ebling, and Yoav Goldberg. 2023. Linguistically motivated sign language segmentation. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 12703–12724, Singapore. Association for Computational Linguistics.

Amit Moryossef, Ioannis Tsochantaridis, Roee Aharoni, Sarah Ebling, and Srini Narayanan. 2020. Real-time sign language detection using human pose estimation. In *European Conference on Computer Vision*, pages 237–248. Springer.

Carol Neidle, Ashwin Thangali, and Stan Sclaroff. 2012. Challenges in development of the American Sign Language lexicon video dataset

(ASLLVD) corpus. In *Proceedings of the LREC2012 5th Workshop on the Representation and Processing of Sign Languages: Interactions between Corpus and Lexicon*, pages 143–150, Istanbul, Turkey. European Language Resources Association (ELRA).

Tom O'Malley, Elie Bursztein, James Long, François Chollet, Haifeng Jin, Luca Invernizzi, et al. 2019. Kerastuner.

OpenCV team, 2025. OpenCV 4.12.0 documentation: Color conversions. Accessed: 2026-03-22.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Wendy Sandler and Diane Carolyn Lillo-Martin. 2006. *Sign language and linguistic universals*. Cambridge University Press.

Zed Sevcikova Sehyr, Naomi Caselli, Ariel M Cohen-Goldberg, and Karen Emmorey. 2021. The ASL-LEX 2.0 project: A database of lexical and phonological properties for 2,723 signs in American Sign Language. *The Journal of Deaf Studies and Deaf Education*, 26(2):263–277.

Zed Sevcikova Sehyr and Karen Emmorey. 2022. The effects of multiple linguistic variables on picture naming in american sign language. *Behavior Research Methods*, 54(5):2502–2521.

Karen Simonyan and Andrew Zisserman. 2015. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*.

Ariel E Stassi, J Matías Di Martino, and Gregory Randall. 2025. A brief review and analysis of two methods for automatic sign language segmentation. *Image Processing On Line*, 15:59–77.

Mukhiddin Toshpulatov, Wookey Lee, Jaesung Jun, and Suan Lee. 2025. Deep learning pathways for automatic sign language processing. *Pattern Recognition*, 164:111475.

Peter Wittenburg, Hennie Brugman, Albert Russel, Alex Klassmann, and Han Sloetjes. 2006. ELAN: A professional framework for multimodality research. In *5th international conference on language resources and evaluation (LREC 2006)*, pages 1556–1559.

10. Language Resource References

Naomi K Caselli, Karen Emmorey, and Ariel M Cohen-Goldberg. 2021. *The signed mental lexicon: Effects of phonological neighborhood density, iconicity, and childhood language experience*. Elsevier.

Zed Sevcikova Sehyr, Naomi Caselli, Ariel M Cohen-Goldberg, and Karen Emmorey. 2021. *The ASL-LEX 2.0 Project: A database of lexical and phonological properties for 2,723 signs in American Sign Language*. Oxford University Press.