Sign Spotting Disambiguation using Large Language Models

Low Jian He
Centre for Vision, Speech and Signal Processing (CVSSP)
University of Surrey
Guildford, Surrey, United Kingdom
jianhe.low@surrey.ac.uk

Ozge Mercanoglu Sincan
Centre for Vision, Speech and Signal Processing (CVSSP)
University of Surrey
Guildford, Surrey, United Kingdom
o.mercanoglusincan@surrey.ac.uk

Richard Bowden
Centre for Vision, Speech and Signal Processing (CVSSP)
University of Surrey
Guildford, Surrey, United Kingdom
r.bowden@surrey.ac.uk

Abstract

Sign spotting, the task of identifying and localizing individual signs within continuous sign language video, plays a pivotal role in scaling dataset annotations and addressing the severe data scarcity issue in sign language translation. While automatic sign spotting holds great promise for enabling frame-level supervision at scale, it grapples with challenges such as vocabulary inflexibility and ambiguity inherent in continuous sign streams. Hence, we introduce a novel, training-free framework that integrates Large Language Models (LLMs) to significantly enhance sign spotting quality. Our approach extracts global spatio-temporal and hand shape features, which are then matched against a large-scale sign dictionary using dynamic time warping and cosine similarity. This dictionary-based matching inherently offers superior vocabulary flexibility without requiring model retraining. To mitigate noise and ambiguity from the matching process, an LLM performs context-aware gloss disambiguation via beam search, notably *without fine-tuning*. Extensive experiments on both synthetic and real-world sign language datasets demonstrate our method's superior accuracy and sentence fluency compared to traditional approaches, highlighting the potential of LLMs in advancing sign spotting.

CCS Concepts

• **Computing methodologies** → **Computer vision**.

Keywords

Sign Language Spotting, Large Language Model, Data Annotations

ACM Reference Format:

Low Jian He, Ozge Mercanoglu Sincan, and Richard Bowden. 2025. Sign Spotting Disambiguation using Large Language Models. In *ACM International Conference on Intelligent Virtual Agents (IVA Adjunct '25), September 16–19, 2025, Berlin, Germany*. ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3742886.3756720

1 Introduction

Sign languages are complex visual languages expressed through coordinated hand gestures, handshapes, facial expressions, and body posture, governed by distinct grammatical and syntactic rules [20, 23]. With over 70 million Deaf individuals worldwide using more than 200 distinct sign languages [21], developing models to understand sign language at scale is both urgent and impactful. However, the multimodal and multi-channel nature of signing presents significant challenges for modeling, particularly when attempting to relate it to spoken language. Unlike spoken languages, sign languages are not linearly structured; they often rely on simultaneous visual cues, and exhibit fundamentally different linguistic properties, making alignment with spoken counterparts non-trivial.

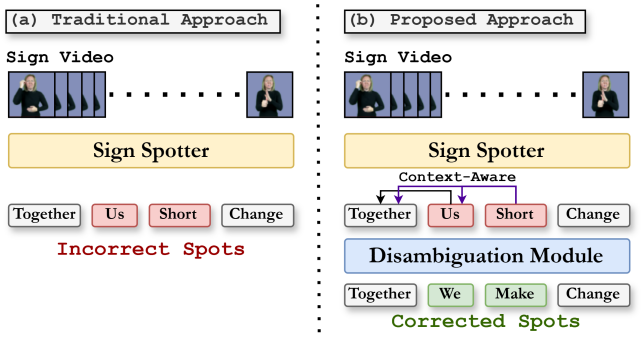

Figure 1: System Overview. (a) Traditional sign spotters classify segments independently. (b) Our proposed approach integrates context-aware disambiguation, leveraging preceding segments for linguistically coherent gloss sequences.

Among the most widely studied tasks in the field are sign language recognition and translation, due to their direct utility in accessibility and communication [15, 33, 36]. Prior work has shown that model performance on these tasks improves substantially when trained on gloss-level annotations [6, 7, 10, 16, 35] or with the aid of large-scale datasets [14]. Glosses are intermediate spoken language representations aligned at the sign level in continuous sign video; however, their annotations are costly, time-consuming, and requires expertise. For instance, annotating 90 seconds of video in How2Sign took one hour [8]. Thus, existing gloss-annotated datasets are small in scale, restricting supervised methods. In contrast, large-scale datasets, pairing videos with spoken language sentences offer broader coverage but lack temporal alignment and structural correspondence; as the order, grammar, and vocabulary of spoken language differ from sign language [1, 24, 26]. This weak supervision introduces ambiguity and hinders translation models from trivially learning precise visual-to-linguistic mappings.

To bridge this gap, one promising solution is sign spotting, which is the task of localizing and identifying individual signs within continuous signing. While spotting can potentially facilitate scalable

IVA Adjunct '25, Berlin, Germany

ACM ISBN 979-8-4007-1996-7/25/09
https://doi.org/10.1145/3742886.3756720

supervision, most existing methods are constrained to isolated dictionary look-up schemes [11, 30] or hierarchical temporal localization frameworks [32]. Although these methods can identify signs at coarse temporal resolutions, they ignore contextual cues, which are crucial for disambiguating visually similar signs. This is particularly problematic, as subtle differences in facial expressions or hand orientation can lead to drastically different meanings, and context is often the only reliable signal for resolving such ambiguities.

Motivated by this, we propose a context-aware disambiguation framework that integrates an LLM into the sign spotting pipeline as seen in Figure 1. While our approach builds on dictionary look-up-based spotting, we enhance it by extracting top-k candidate glosses for each localized segment and leveraging the LLM's next-token prediction capabilities to evaluate gloss sequences based on linguistic coherence. We cast disambiguation as a constrained decoding task, employing beam search to explore high-probability gloss combinations informed by language priors. This enables the system to move beyond isolated sign predictions and instead produce coherent, contextually grounded gloss sequences. To further improve spotting performance, we explore a range of ensemble and fusion strategies for integrating visual cues. Overall, our approach bridges low-level visual recognition with high-level linguistic reasoning and demonstrates that incorporating LLM-based priors can significantly enhance robustness and accuracy in the presence of sign ambiguity. In summary, our main contributions are as follows:

- We introduce a novel, context-aware and training-free disambiguation module that can be seamlessly integrated into existing sign spotting systems.
- We evaluate and present several ensemble and fusion techniques to enhance spotting performance.
- We show that our disambiguation framework substantially reduces word error rate (WER), demonstrating significant gains over baseline spotting architectures.

2 Related Works

Sign language spotting is a key task in sign language understanding, where the goal is to identify the temporal boundaries and the identities of signs within continuous signing videos, given a predefined vocabulary. While underutilized, spotting holds huge potential for scaling the annotation of large sign language datasets. By reducing the cost and effort of manual labeling, it can facilitate the creation of richly supervised data, leading to more accurate and robust sign language translation systems. Beyond annotation, sign spotting also offers further possibilities, including keyword-based video search, dataset curation, and interactive learning applications.

Early approaches to sign spotting largely relied on handcrafted features and traditional sequence modeling techniques. Conditional Random Fields (CRFs), for instance, were employed with adaptive thresholding to differentiate between meaningful signs and co-articulation [34]. Other methods modeled the spatial distribution of non-face skin regions using handcrafted histograms, paired with Dynamic Time Warping (DTW) for sequence alignment and retrieval [3, 31]. In parallel, spatio-temporal patterns such as Sequential Interval Patterns (SIPs) were explored as sign representations, with work proposing a hierarchical forest of SIP trees to enhance spotting robustness across varying sign instances [22].

The deep learning era then introduced more scalable and accurate methods. One-shot spotting approaches emerged by computing similarity between embeddings of isolated signs and continuous video segments, leveraging powerful video encoders like the Inflated 3D ConvNet (I3D) [5]. For example, [18] proposed a multi-supervision framework using an I3D, and spotted signs via computing similarity scores between dictionary embeddings and localized segments. Meanwhile, the Sign-Lookup approach [11] utilized a 3D Convolutional Neural Network (CNN) with a Transformer, performing a sliding window over the video input to compute similarities with dictionary signs. More recently, dictionary-free approaches have also emerged, with [32] introducing a hierarchical framework that uses I3D features at different layers to learn coarse-to-fine temporal boundaries and perform sign classifications.

3 Methodology

As illustrated in Figure 2, our proposed sign spotting disambiguation framework introduces a novel integration of LLMs into the sign spotting pipeline to enhance the selection of sign candidates through linguistic reasoning. The idea is to treat the probability distribution output from a sign spotting model as a set of candidate glosses, which are then passed to an LLM that acts as a contextual scorer. By leveraging the LLM's strong language modeling and next-token prediction capabilities, we reinterpret its output as transition probabilities within a beam search decoding process.

While the LLM introduces linguistic priors into the pipeline, the quality of the spotting predictions (output probabilities) remains critical. Thus, we investigate various strategies for improving these predictions in Section 3.1.3, including ensembling multiple sign spotting outputs and applying feature-level fusion to enhance the robustness and precision of the extracted gloss candidates.

3.1 Sign Spotter

Our sign spotter consists of three main components: (1) a multi-branch **feature extraction module**, where each stream is responsible for extracting distinct aspects of the signing input; (2) a **dictionary-based matching module**, which performs candidate retrieval based on embedding similarity; and (3) a **feature fusion stage**, where we explore various ensembling strategies to integrate motion and hand-specific information.

3.1.1 ***Feature Extraction.*** We employ two specialized neural networks to extract complementary representations from the signing video. The first is a spatiotemporal encoder based on the ***Inflated 3D ConvNet (I3D)*** [5], which captures coarse motion patterns and global appearance cues. We employ the I3D which was pretrained on the *BOBSL* dataset [1], and finetuned for sign language recognition by [25]. Here, each video is processed using a sliding window of fixed length to extract local segment-level embeddings.

Formally, we define the input video as, $V \in \mathbb{R}^{T\times H\times W\times 3}$, where T is the total number of frames, and $H \times W$ is the spatial resolution. We extract features using overlapping 16-frame windows with stride 1. For each window $V_{t:t+15}$, the I3D network produces a 1024-dimensional embedding:

$$\mathbf{f}_t^{\text{I3D}} = \text{I3D}(V_{t:t+15}) \in \mathbb{R}^{1024}$$

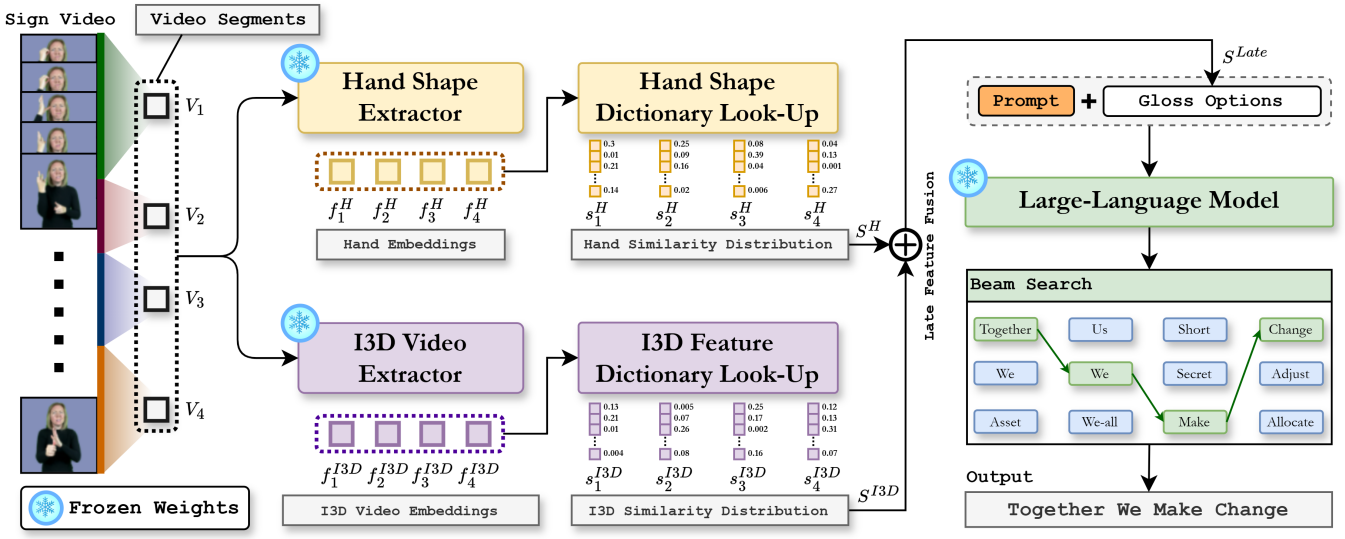

Figure 2: Overall Architecture. Our approach features two main stages: (a) Sign Spotting: Sign segments are first obtained from video. Hand shape and I3D features are then extracted for each segment, and subsequently passed to a dictionary lookup to yield a similarity distribution. Late fusion is then employed to further enhance distinguishability. (b) Disambiguation Module: This module takes the similarity distribution and passes it to an LLM. The LLM's linguistic capabilities are then leveraged, as we extract its transition probabilities and a beam search algorithm uses it to identify the most coherent sequence of sign glosses.

Aggregating over the entire sequence, we obtain the motion-based feature matrix:

$$\mathbf{F}^{\text{I3D}} = \left[\mathbf{f}_1^{\text{I3D}}, \mathbf{f}_2^{\text{I3D}}, \ldots, \mathbf{f}_{T-15}^{\text{I3D}}\right]^\top \in \mathbb{R}^{(T-15)\times 1024}$$

As the I3D uses 3D convolutions, each frame will always be influenced by its temporal context, which allows the model to capture rich motion and appearance patterns that evolve over time. This resulting feature sequence $\mathbf{F}^{\text{I3D}}$ thus serves as a temporally-aware embedding of the input video, which is later fused with complementary features for candidate retrieval.

In parallel, we employ a ***ResNeXt-101-based hand shape encoder*** to capture fine-grained spatial cues from both hands. This model is similar to DeepHand [12], which was originally trained on the 1 Million Hand Images dataset for handshape classification. In our adaptation, we leverage a deeper ResNeXt-101 backbone, pretrained on the same dataset for 60-way handshape classification. However, we remove the classification head and instead use the penultimate layer features as dense hand embeddings.

To isolate hand regions from full-frame signing videos, we first detect upper-body pose landmarks using *MediaPipe* [17], and then crop bounding boxes around the left hands (LH) and right hands (RH) at each frame. Thus, given an input video $V \in \mathbb{R}^{T\times H\times W\times 3}$, MediaPipe produces localized crops:

$$H_t^{\text{LH}}, H_t^{\text{RH}} \in \mathbb{R}^{h\times w\times 3}, \quad \forall t \in \{1, \ldots, T\}$$

where H_t^{LH} and H_t^{RH} denote the cropped left and right hand regions at time step t. These are then passed through the ResNeXt-101 encoder to extract hand-specific features:

$$\mathbf{F}_t^{\text{LH}} = \text{ResNeXt}(H_t^{\text{LH}}), \quad \mathbf{F}_t^{\text{RH}} = \text{ResNeXt}(H_t^{\text{RH}}) \in \mathbb{R}^{2048}$$

These embeddings capture high-resolution spatial information from each hand independently, providing a complementary signal to the more holistic, motion-oriented features extracted by the I3D.

3.1.2 ***Dictionary-Based Matching.*** To obtain gloss candidates, we employ a dictionary-based matching approach that leverages feature similarity. While alternative methods, such as those proposed by [25, 32], forgo dictionaries by using pretrained classifiers (e.g., a final fully connected layer for prediction), they are inherently constrained by a fixed vocabulary size. This rigidity, though potentially offering robustness for predefined signs, leads to significant Out-Of-Vocabulary (OOV) issues when applied to novel data or signs not encountered during training. Such inflexibility is a critical drawback, as a primary application of sign spotting is the annotation of large-scale datasets with potentially dynamic or undefined vocabularies. In contrast, dictionary-based methods offer the key advantage of allowing new vocabulary items to be incorporated into the lookup table without necessitating model retraining, thereby effectively addressing OOV instances. Thus, we adopt the dictionary-based approach due to its flexibility and further explore its performance benefits in the results of Sec 4.2.

Our sign dictionary is constructed from isolated British Sign Language (BSL) samples and comprises of 1,000 vocabulary items (details at Sec. 4.2). Each dictionary entry encodes the prototypical visual representation of a gloss using extracted feature embeddings. Formally, we define the dictionary as:

$$\mathcal{D} = \{(\mathbf{D}_i, g_i)\}_{i=1}^{1000}, \tag{1}$$

where $\mathbf{D}_i$ is the feature embedding for the i-th gloss and g_i is its corresponding label.

Each dictionary embedding $\mathbf{D}_i$ is composed by concatenating the I3D motion features and the hand shape features extracted from both left and right hands:

$$\mathbf{D}_i = \mathbf{F}_i^{I3D} \oplus \mathbf{F}_i^{LH} \oplus \mathbf{F}_i^{RH} \in \mathbb{R}^{5120}, \tag{2}$$

where $\oplus$ denotes feature concatenation.

This dictionary is constructed using isolated sign videos to ensure clean visual representations. Importantly, the structure of $\mathbf{D}_i$ is modular, allowing us to build variant-specific dictionaries tailored to different feature configurations *without retraining*. For instance, when exploring fusion approaches (Sec. 3.1.3), we constructed separate dictionaries using only I3D features and right-hand embeddings due to the difference in feature extraction pipeline.

Similarity Computation. For each sign unit U_x, where $x \in \{1, \ldots, X\}$ and X is the total number of candidate segments from a continuous signing video, we compute similarity scores against all dictionary entries $\mathbf{D}_i$ using two complementary metrics: (i) **Dynamic Time Warping (DTW)** [19], which aligns frame-level features of U_x and $\mathbf{D}_i$ to capture temporal structure while accounting for variations in signing speed; and (ii) **Cosine Similarity**, which is computed between pooled segment-level embeddings. Specifically, frame-wise features are temporally pooled into fixed-size vectors $\mathbf{F}_x \in \mathbb{R}^d$, enabling efficient segment-level comparison.

The final similarity score is then computed using a weighted sum as seen in equation 3, and visualized in Figure 3:

$$\text{score}(U_x, \mathbf{D}_i) = (\alpha_s - 1) \cdot \text{sim}_{\text{DTW}}(U_x, \mathbf{D}_i) + \alpha_s \cdot \text{sim}_{\cos}(U_x, \mathbf{D}_i), \tag{3}$$

where $\alpha_s \in [0, 1]$ is a hyperparameter controlling the contribution of each similarity metric. Note that DTW is a distance metric (lower means more similarity); thus, we multiply its scalar with a negative sign to ensure higher values consistently indicate greater similarity.

3.1.3 ***Feature Fusion.*** To enhance the quality of gloss selection within the Top-k candidate set, we investigate several feature fusion strategies aimed at increasing the discriminative power of the segment representations. Specifically, we explore: (i) **late fusion**, where features are combined after being independently encoded; (ii) **intermediate fusion**, which integrates features within a shared embedding space during encoding; and (iii) a **full-ensemble** strategy that jointly leverages both approaches. These fusion mechanisms are crucial for ensuring that the LLM receives more informative gloss candidates for downstream disambiguation.

Late Fusion. For late fusion, we integrate the independently computed similarity distributions derived from the I3D and RH embeddings. Initially, as depicted in Figure 3, the I3D embedding, $\mathbf{F}^{\text{I3D}}$, is evaluated against its dedicated I3D dictionary. This evaluation leverages the similarity computation defined in equation 3, resulting in a similarity distribution $\mathbf{S}^{\text{I3D}} \in \mathbb{R}^V$, where V represents the total number of vocabulary entries. In parallel, the RH embedding, $\mathbf{F}^{\text{RH}}$, undergoes an analogous process with its corresponding RH dictionary to generate $\mathbf{S}^{\text{RH}}$.

The final late fusion of these distributions is then achieved through a weighted summation:

$$\mathbf{S}^{\text{Late}} = \alpha_{\text{late}} \cdot \mathbf{S}^{\text{I3D}} + (1 - \alpha_{\text{late}}) \cdot \mathbf{S}^{\text{RH}}, \tag{4}$$

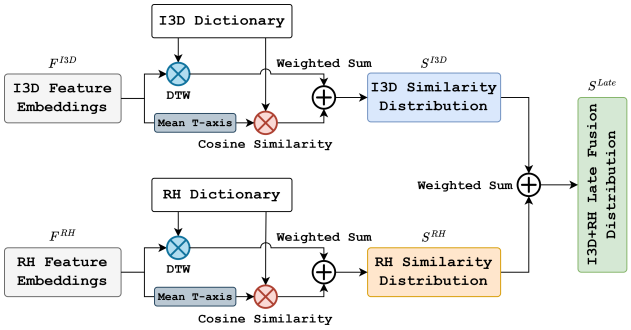

Figure 3: Illustration of Dictionary Look-Up and Late Fusion.

where $\alpha_{\text{late}} \in [0, 1]$ is a tunable hyperparameter that balances the influence of the I3D and RH shape features. We observed through preliminary experimentation that the inclusion of LH features provided negligible performance improvements; thus, they are excluded from this specific late fusion strategy.

Intermediate Fusion. For intermediate fusion, we combine feature embeddings from multiple encoders before similarity computation. Specifically, for each sign unit, the I3D ($\mathbf{F}^{\text{I3D}} \in \mathbb{R}^{1024}$), RH ($\mathbf{F}^{\text{RH}} \in \mathbb{R}^{2048}$), and LH ($\mathbf{F}^{\text{LH}} \in \mathbb{R}^{2048}$) embeddings are concatenated to form a unified feature vector $\mathbf{F}^{\text{Mid}}$:

$$\mathbf{F}^{\text{Mid}} = \mathbf{F}^{\text{I3D}} \oplus \mathbf{F}^{\text{RH}} \oplus \mathbf{F}^{\text{LH}} \in \mathbb{R}^{5120}$$

This fused vector then serves as the input for dictionary-based matching, as depicted in Figure 4. This process ultimately yields the intermediate fusion similarity distribution $\mathbf{S}^{\text{Mid}} \in \mathbb{R}^V$. Note that unlike late fusion, we exclusively employ DTW for similarity computation at this stage; as cosine similarity, which typically involves temporal averaging, was found to dilute crucial modality-specific information post-concatenation, leading to suboptimal performance.

Full-Ensemble. For the final full-ensemble strategy, we combine the concepts of both the late and intermediate fusion approaches. Specifically, we perform a weighted late fusion between the similarity distribution obtained from intermediate fusion, $\mathbf{S}^{\text{Mid}}$, and the I3D-specific distribution, $\mathbf{S}^{\text{I3D}}$, as depicted in Figure 4. The ensemble similarity distribution, $\mathbf{S}^{\text{Ensemble}}$, is computed as:

$$\mathbf{S}^{\text{Ensemble}} = \alpha_{\text{ens}} \cdot \mathbf{S}^{\text{Mid}} + (1 - \alpha_{\text{ens}}) \cdot \mathbf{S}^{\text{I3D}}, \tag{5}$$

where $\alpha_{\text{ens}} \in [0, 1]$ is a tunable hyperparameter balancing the contributions of the intermediate and I3D feature modalities.

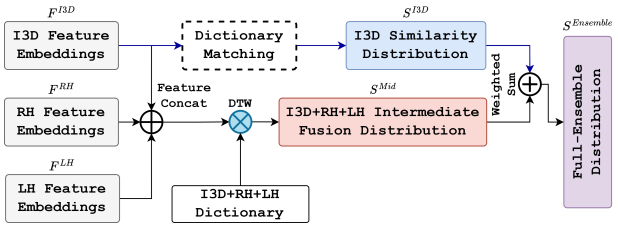

Figure 4: Illustration of Mid Fusion and Full-Ensemble.

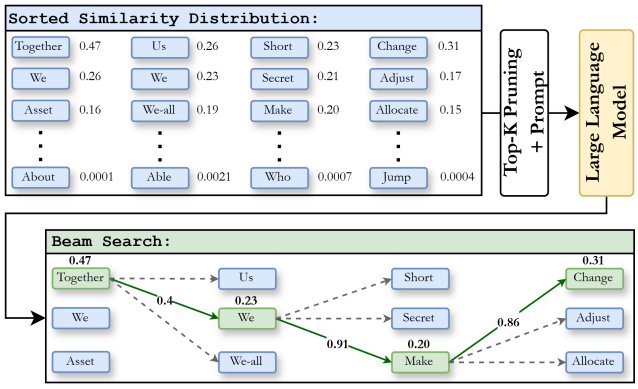

Figure 5: Visualization of the Disambiguation Process.

3.2 Linguistic Disambiguation

This section details the LLM-based disambiguation stage, which comprises of three key components: (i) **Gloss Candidate Generation**, detailing how initial glosses are prepared from similarity distributions; (ii) **Prompt Formulation**, describing the structure and content of the prompt used for the LLM; and (iii) **Beam Search Decoding**, which outlines the final disambiguation process.

3.2.1 ***Gloss Candidate Generation.*** From the preceding dictionary look-up stage, a *similarity distribution* $\mathrm{S} \in \mathbb{R}^V$ (where V is the vocabulary size) is obtained for each sign segment, indicating the relevance of each potential gloss candidate. This raw distribution undergoes normalization via a softmax function, and its Top-k gloss candidates are then selected and forwarded to the LLM along with the prompt for subsequent disambiguation (Figure 5).

Prompt:

Given the incomplete sentence of TOGETHER WE, what is the most logical next word from the options below?

List of Options: SHORT, SECRET, MAKE

Answer: Given the options, the best sentence is TOGETHER WE

LLM: MAKE

Figure 6: Example of Our Prompt Template.

3.2.2 ***Prompt Formulation.*** The prompt provided to the LLM is dynamically constructed to incorporate both contextual information and candidate glosses. As seen in Figure 6, specific elements of the prompt are instantiated with variable content. For instance, the red-highlighted "Short," "Secret," and "Make" represent the Top-k gloss candidates, while the blue-highlighted "Together we" signifies the previous sign glosses, providing contextual information.

Thus, the prompt's core function is to leverage the LLM's predictive capabilities for sentence completion using the provided context and gloss options (e.g., completing "Together we ___"). To extract the transition probabilities for each candidate, the prompt is specifically designed to compel the LLM to predict one of the gloss options as its next word; thus, allowing for direct extraction of each candidate's probability as the subsequent token in the sequence.

3.2.3 ***Beam Search Decoding.*** For the final disambiguation, a beam search algorithm is employed to combine the two critical probability types: ***emission probabilities*** (representing gloss candidates from the sign spotter) and ***transition probabilities*** (reflecting linguistic coherence from the LLM).

Thus, for the x^{th} sign unit, where $x \in \{1, \ldots, X\}$ (X denoting the total number of units), we first obtain a set of top-k gloss candidates $C_x = \{(g_x^j, s_x^j)\}_{j=1}^k$. Here, s_x^j, which is derived from the pre-computed dictionary similarity (e.g., S^{Late}, S^{Mid}, or $\mathrm{S}^{\text{Ensemble}}$), serves as the emission probability for candidate gloss g_x^j. Meanwhile, the LLM provides our transition probabilities $p(g_x|g_{1:x-1})$, indicating the likelihood of gloss g_x given the sequence of previously selected glosses $g_{1:x-1}$, providing us with linguistic guidance.

Beam search then aims to find the optimal gloss sequence $\hat{g}_{1:X}$ by maximizing a combined score that leverages these probabilities:

$$\hat{g}_{1:X} = \underset{g_{1:X} \in \prod_{x=1}^{X} C_x}{\arg\max} \sum_{x=1}^{X} \left(\log p(g_x|g_{1:x-1}) + \alpha_{\text{bs}} s_x \right) \tag{6}$$

In this objective, $\prod_{x=1}^{X} C_x$ denotes the Cartesian product of the candidate sets. The conditional probability $p(g_x|g_{1:x-1})$ is derived from the frozen LLM's logits, while the term s_x represents the emission probability for the selected gloss g_x at the x^{th} sign unit. The hyperparameter α_{bs} weighs the the LLM's linguistic score and the emission probability. To sufficiently explore the hypothesis space, we employ the algorithm with a beam width (BW) of 5.

4 Experiments and Results

Our evaluation of the proposed system employs a two-tiered strategy. First, an ***isolated gloss disambiguation*** evaluation validates the processes detailed in Sec. 3.2 on synthetic data to ensure the module's efficacy prior to integration. Subsequently, for the ***full system evaluation***, the validated disambiguation module is integrated with a sign spotter, and assessed on continuous sign videos and a large sign dictionary comprising 1000 vocabulary entries.

4.1 Isolated Gloss Disambiguation Evaluation

Synthetic Data Generation for Disambiguation Evaluation. To evaluate the gloss disambiguation process in isolation and conduct comprehensive ablation studies, we utilize synthetically generated data. This pipeline is designed to simulate the complexities and ambiguities inherent in real-world sign spotting, encompassing dictionary construction, pseudo-gloss sentence generation, and controlled noise injection to mimic realistic emission probabilities.

4.1.1 ***Synthetic Dictionary Construction.*** Existing large-scale British Sign Language (BSL) dictionaries often lack crucial, high-frequency vocabulary. To address this, we construct a synthetic BSL-based dictionary informed by word frequency research. We leverage a word frequency list derived from the Google Web Trillion Word Corpus [27], cross-referencing it with findings from a BSL lexical frequency study [9], which identified the 100 most frequent BSL signs, and BSLDict [18]. We thus constructed four vocabulary lists of varying sizes: 1500, 2000, 3000, and 4373 words. The maximum size was capped at 4373, as it corresponded to the highest number of overlapping words between the Google frequency list and the

Table 1: Hyperparameter Optimization of the α from ranges 0 to 1 for Different Fusion Equations based on Top-K Accuracy

Equation	Accuracy ↑	$\alpha = 0$	$\alpha = 0.1$	$\alpha = 0.2$	$\alpha = 0.3$	$\alpha = 0.4$	$\alpha = 0.5$	$\alpha = 0.6$	$\alpha = 0.7$	$\alpha = 0.8$	$\alpha = 0.9$	$\alpha = 1.0$
S^{I3D}, Eq. 3	Top-5	0.7888	0.8141	0.8155	**0.8272**	0.8255	0.8222	0.8222	0.8166	0.8066	0.8103	0.8056
	Top-1	0.4605	0.4788	0.4834	**0.5151**	0.5136	0.5050	0.5050	0.5050	0.5050	0.5136	0.5136
S^{RH}, Eq. 3	Top-5	0.5511	0.5608	0.5670	0.5725	0.5917	**0.5965**	**0.5965**	0.5923	0.5923	0.5875	0.5917
	Top-1	0.3212	0.3242	0.3200	0.3219	0.3219	0.3261	0.3280	0.3280	0.3338	**0.3397**	**0.3397**
S^{Late}, Eq. 4	Top-5	0.5923	0.6558	0.6973	0.7625	0.7738	0.8054	0.8132	0.8215	0.8259	**0.8276**	0.8272
	Top-1	0.3338	0.4186	0.4793	0.5122	0.4952	**0.5451**	0.5343	0.5259	0.5148	0.5020	0.5151
$S^{Ensemble}$, Eq. 5	Top-5	0.8272	0.8230	0.8197	0.8317	0.8373	**0.8415**	**0.8415**	**0.8415**	0.8393	0.8223	0.8260
	Top-1	0.5151	0.5039	0.5095	0.5137	0.5176	0.5109	**0.5190**	0.5160	0.5080	0.5118	0.5023

BSLDict sign vocabulary. This constructed dictionary also serves as a template for potential future sign spotting applications.

4.1.2 ***Pseudo-Gloss Generation.*** To generate synthetic sign spotted units, we employ a pseudo-gloss generation strategy. We first curate a corpus of random English sentences using a random sentence generator [29]. Motivated by recent work in sign language translation [33], we then create pseudo-glosses by applying Part-of-Speech (POS) tagging via the SpaCy library. Only tokens tagged as "NOUN", "NUM", "ADV", "PRON", "PROPN", "ADJ", and "VERB" are retained, simulating the typical lexical content of sign glosses.

4.1.3 ***Simulating Ambiguity via Noise Injection.*** To realistically simulate the dictionary-matching similarity distributions obtained from visual sign spotters, we initially embed both the synthetic sentences and dictionary vocabulary items using the FastText library and perform direct matching. However, we recognize that this naive approach yields significantly better matches than complex visual embedding; thus, we incorporate two types of noise augmentation to introduce real-life ambiguity:

Word Replacement (WR):. The top-1 option of the similarity distribution is replaced with another random word at a specified probability, simulating an incorrect classification by a sign spotter.

Distribution Corruption (DC):. From a given distribution, we randomly select k semantically dissimilar words. We then artificially increase their similarity scores to a value exceeding the highest pre-existing score. This introduces significant noise, ensuring that random, irrelevant words appear in the top-k candidates, thereby stringently testing the disambiguation capability of our method.

4.1.4 ***Evaluation and Results.*** Based on the methods above, we conduct a series of **hyperparameter optimization** (HPO) and **ablation studies** to select the optimal configuration for the disambiguation module and evaluate its overall effectiveness.

Hyperparameter Optimization. We first perform HPO on the weighted sum parameters used in both the similarity computation and feature fusion stages. Specifically, we tune the balancing hyperparameters $\alpha \in [0, 1]$ for four distinct combination strategies: (i) the sum of DTW and Cosine Similarity for **I3D features** (Eq. 3, used for S^{I3D} derivation), (ii) the sum of DTW and Cosine Similarity for **RH features** (Eq. 3, used for S^{RH} derivation), (iii) the **late fusion** strategy (Eq. 4), and (iv) the **full-ensemble** method (Eq. 5). Performance is measured using Top-1 and Top-5 accuracy.

Based on table 1, which summarizes the HPO results, we find that the I3D similarity computation benefits from a greater contribution from DTW. In contrast, the RH similarity computation achieves its peak Top-1 accuracy at $\alpha_s = 0.9$, indicating stronger reliance on cosine similarity. For feature fusion, the late fusion strategy shows optimal Top-5 accuracy (0.8276) when I3D features contribute more ($\alpha_{late} = 0.9$). Meanwhile, the full-ensemble method also performs strongly, with peak performances observed at $\alpha_{ens} = 0.6$, suggesting a slightly greater emphasis on the intermediate fusion distribution (S^{Mid}). Overall, the explicit feature fusion methods consistently outperform the vanilla S^{I3D} and S^{RH} baselines.

Dictionary Size. As detailed in Section 4.1.1, our synthetic dictionary generation yielded varying vocabulary sizes. Here, we analyze performance across dictionary sizes of 1500, 2000, and 4373 words under different noise augmentation conditions (Table 2). These conditions include Word Replacement (WR) rates of 50% and 100%, and Distribution Corruption (DC) rates of 5, 10, and 15. Consistently, the dictionary size of 4373 achieves the lowest Word Error Rates (WER) across all augmentation conditions, demonstrating that a larger vocabulary enhances the sign spotting and disambiguation process due to a broader range of potential matches.

Table 2: Comparisons between Different Dictionary Sizes

WR	DC	$D1500_{WER\downarrow}$	$D2000_{WER\downarrow}$	$\mathbf{D4373_{WER\downarrow}}$
50%	5	0.4812	0.4749	**0.3824**
100%	5	0.7670	0.7691	**0.6206**
50%	10	0.4606	0.4432	**0.3723**
100%	10	0.8110	0.7817	**0.6557**
50%	15	0.5043	0.4585	**0.4041**
100%	15	0.8111	0.7891	**0.7163**

Augmentation Strength. Additionally, we also evaluate the robustness of our approach to noise and conduct an ablation study comparing different LLMs. Specifically, we compare *Phi-3 Mini* [2], selected for its strong performance among lightweight models, with *Gemma-2 9B* [28], the most capable model that fit our RTX3090 GPU. Our evaluation includes tests with increasing Word Replacement (WR) rates (25% to 100%) and Distribution Corruption (DC) levels (5 to 20), detailed in Table 3, alongside extreme conditions (DC=30; WR=50%, 100%) with varying Beam Widths, presented in Table 4.

Table 3: Comparison at Different Word Replacement and Distribution Corruption Rates

WR	DC	Phi-3 Mini *(WER)*		Gemma-2 9B *(WER)*	
		Top-1 ↓	Top-5 ↓	Top-1 ↓	Top-5 ↓
Fixed Distribution Corruption					
25%	5	0.2446	0.1981	0.1799	0.1316
50%	5	0.3035	0.2385	0.2555	0.1714
75%	5	0.4667	0.3783	0.3420	0.2648
100%	5	0.6231	0.5036	0.4807	0.3416
Fixed Word Replacement					
50%	5	0.3192	0.2470	0.2661	0.1880
50%	10	0.3648	0.2562	0.2687	0.2156
50%	15	0.4065	0.3453	0.3205	0.2517
50%	20	0.4396	0.3867	0.4086	0.3495

Table 4: Table of Results under Extreme Noise Scenarios

WR	DC	BW	Phi-3 Mini *(WER)*		Gemma-2 9B *(WER)*	
			Top-1 ↓	Top-5 ↓	Top-1 ↓	Top-5 ↓
50%	30	5	0.4898	0.4835	0.4634	0.4378
50%	30	10	0.3843	0.3420	0.3016	0.2334
50%	30	30	0.4091	0.3207	0.3776	0.3136
100%	30	5	0.8313	0.7945	0.8274	0.8052
100%	30	15	0.7376	0.6520	0.6316	0.5491
100%	30	30	0.7820	0.7282	0.6686	0.6219
100%	30	50	0.6080	0.5054	0.6915	0.5789

In Table 3, we see that Gemma-2 outperformed Phi-3 across all scenarios, highlighting the benefit of utilizing a larger LLM. However, overall, both models demonstrated clear disambiguation capabilities, even at 100% WR, as they successfully reduced Top-1 WER to under 0.63 (100% $\rightarrow <$ 63%). This suggests that successful corrections are achievable as long as target glosses remain within the top candidates. However, Phi-3 struggled significantly at high DC (DC=20), yielding a WER of 0.44. This performance, offering only a 6% reduction in WER (50% → 44%), highlights its sensitivity to the quality of Top-k candidates compared to Gemma-2.

Meanwhile, under extreme noise, as seen in Table 4, a Beam Width of 5 consistently led to much poorer WER for both models. This is likely due to target glosses being pushed out of the limited Top-5 search space. Conversely, higher Beam Widths (10-15) generally improved performance, effectively balancing candidate exploration with noise introduction. Ultimately, these evaluations consistently show that the LLM algorithm's performance is directly constrained by the quality of its input probability distributions; thus, accurate initial gloss candidate generation is salient.

4.2 Full System Evaluation

Following validation on synthetic data, we integrate the disambiguation module into the sign spotter for real-world evaluation. A key challenge, however, is the scarcity of public datasets offering both gloss-annotated continuous sign language and a paired isolated sign video dictionary. While datasets like Phoenix14T [4], CSLDaily [37], and MeinDGS [13] provide continuous glosses, they lack isolated video dictionaries. Conversely, BSLDict [30] offers dictionaries without continuous gloss-annotated video.

Therefore, we leverage an internally collected continuous sign language dataset, and pair it with a dictionary of 1000 vocabulary entries, each associated with an isolated sign video. We then conduct evaluations in two parts, first evaluating all previously discussed fusion methods without disambiguation against a baseline (Sec 4.2.1), then assessing the impact of integrating the LLM disambiguation module (Sec 4.2.2).

4.2.1 ***Performance of Sign Spotters (No Disambiguation).*** Table 5 presents the performance of various sign spotter configurations without the disambiguation module. We employ the I3D Sign Spotter (without dictionary-matching) [25] as a baseline, and it performed worst with WER of 0.9089. This high error rate stems from its fixed 2,281-gloss BOBSL vocabulary, leading to OOV issues that are impractical to mitigate through retraining due to data scarcity.

In contrast, approaches utilizing the dictionary-matching algorithm achieved substantially lower WERs (around 0.5). This highlights the benefits of dictionary-matching, which offers superior vocabulary flexibility and a training-free nature that allows easy incorporation of new gloss entries. Among these, the ***Late Fusion*** and ***Full-Ensemble*** methods proved most effective, achieving WERs of 0.4724 and 0.4924, respectively. Their improved performance is attributed to feature fusion techniques that enhanced discrimination for dictionary matching. Notably, the Late Fusion approach consistently performed best, aligning with its highest Top-1 Gloss Accuracy (54.51%) observed during α_{late} tuning (Table 1).

4.2.2 ***Impact of LLM Beam Search Integration.*** Integrating the disambiguation module into the top two performing dictionary-matching approaches led to further performance improvements, as seen in Table 6. Specifically, the Late Fusion WER decreased from 0.4724 to 0.4438 (Phi-3) and 0.4473 (Gemma-2); while the Ensemble

Table 5: Result Comparisons for Different Fusion Methods without Disambiguation Module

Approach	Top-1 ↓ *(WER)*
BOBSL I3D Sign Spotter [25]	0.9089
I3D (DTW+Cosine Fusion)	0.5117
RH (DTW+Cosine Fusion)	0.6909
RH+I3D Late Fusion	**0.4724**
I3D+RH+LH Intermediate Fusion	0.5060
IF+I3D Full-Ensemble	0.4924

Table 6: Result Comparisons with Disambiguation Module

Approach	Phi-3 *(WER)*		Gemma-2 *(WER)*	
	Top-1 ↓	Top-5 ↓	Top-1 ↓	Top-5 ↓
RH+I3D Late Fusion	**0.4438**	**0.3562**	**0.4473**	**0.3481**
IF+I3D Full-Ensemble	0.4567	0.3779	0.4647	0.3644

method's WER dropped from 0.4924 to 0.4567 (Phi-3) and 0.4647 (Gemma-2). This highlights the disambiguation module's ability to incorporate linguistic knowledge when forming the final sequence of glosses, as it moves beyond a simple output of individually spotted glosses. Additionally, utilizing beam search also yields multiple highly probable gloss combinations, increasing the likelihood of obtaining accurate gloss sequences. This benefit persists even if the top-ranked combination is imperfect, as the correct sequence may still reside among the lower-ranked alternatives.

4.3 Qualitative Results

This section presents qualitative results visualizing the performance of our best-performing approach: the **Late Fusion with integrated disambiguation module**. We first illustrate its Top-3 gloss predictions against the ground truth (Sec 4.3.1). Then, we provide comparisons between the baseline I3D sign spotter and the Late Fusion model, both with and without the disambiguation module, to highlight the impact of our proposed components (Sec 4.3.2).

4.3.1 ***Top-3 vs Ground-Truth.*** An analysis of Top-3 predictions against ground truth in Table 7 demonstrates the disambiguation approach's accuracy. Predictions were generally strong, but the most precise sequences often appeared at ranks 2 or 3, particularly when glosses were semantically similar. For instance, in Example 1, the Top-3 outputs were near-identical, with differences only in the glosses "ME", "I", and "MYSELF". This confusion stems from the significant semantic and signing motion overlap between these signs. Similar ambiguities were also observed with "ME"/"MY" (Example 2) and "TRAIN"/"RAIL" (Example 3). Fortunately, such instances typically result in minimal translational discrepancies, thus a contextually sound option is often present within the Top-3 candidates.

4.3.2 ***Qualitative Comparisons against Baseline.*** Additionally, we also offer a qualitative comparison between the baseline I3D Sign Spotter and our proposed method in Table 8. Here, the baseline spotter consistently produced highly inaccurate gloss sequences. This deficiency likely stems from its fixed vocabulary, which can lead to severe OOV issues and produce outputs which are frequently irrelevant to the ground truth. In contrast, our dictionary-based approach demonstrated substantially improved performance, capitalizing on the the vocabulary flexibility afforded by the training-free dictionary and the enhanced gloss classification via feature fusion. The subsequent integration of the disambiguation module (Ours w/ LLM) then further refined these outputs. For instance, it corrected "US" to "WE-ALL" (Example 2, 4) and "SHORT" to "MAKE" (Example 3), clearly demonstrating its error-correcting capabilities.

Despite these improvements, failure cases were present, primarily stemming from semantic ambiguities between predicted and target glosses (e.g., "DIVERSITY" or "VARIETY" for "MULTIPLE-DIFFERENT" in Example 2). Although such inaccuracies elevate WER, the resulting translations generally still remain comprehensible. Thus, the LLM integration provides significant contributions, its value being evident even with these identified limitations.

Table 7: Comparison of Top-3 predictions vs Ground Truth

Ground Truth:	I LOVE WALKING SUNDAY AFTERNOON
Top-3 Prediction:	1. ME LOVE WALKING SUNDAY AFTERNOON
	2. I LOVE WALKING SUNDAY AFTERNOON
	3. MYSELF LOVE WALKING SUNDAY AFTERNOON
Ground Truth:	NAME ME FS_CHRIS FS_WOOD
Top-3 Prediction:	1. NAME MY FS_CHRIS FS_WOOD
	2. NAME MY FS_CHRIS JUNE
	3. NAME ME FS_CHRIS FS_WOOD
Ground Truth:	NEXT TRAIN ARRIVE PLATFORM PEOPLE BOARD IMPOSSIBLE
Top-3 Prediction:	1. NEXT RAIL ARRIVE PLATFORM CUSTOMER BOARD IMPOSSIBLE
	2. NEXT TRAIN ARRIVE PLATFORM CUSTOMER BOARDING IMPOSSIBLE
	3. NEXT RAIL ARRIVE PLATFORM ACKNOWLEDGE BOARDING IMPOSSIBLE

Table 8: Qualitative Comparisons against Baseline Method

Ground Truth	I WORK UNIVERSITY
Baseline [25]	HEART WALL UNIVERSITY
Ours	**I WORK UNIVERSITY**
Ours w/ LLM	**I WORK UNIVERSITY**
Ground Truth	WE-ALL PROVIDE SKILLS GENIUS MULTIPLE-DIFFERENT AREA
Baseline [25]	NEED SECOND WARM MOISTURE EXPLAIN EXPLAIN
Ours	US **PROVIDE SKILLS GENIUS** VARIETY **AREA**
Ours w/ LLM	**WE-ALL PROVIDE SKILLS GENIUS** DIVERSITY **AREA**
Ground Truth	TOGETHER WE MAKE CHANGE BEEN
Baseline [25]	EXPECT MAKE GO MANY MANY
Ours	**TOGETHER WE** SHORT **CHANGE** PAIN
Ours w/ LLM	**TOGETHER WE MAKE CHANGE** PAIN
Ground Truth	WE-ALL IDEAS BOTH-FORWARD OPEN-MINDED IN EVERYTHING WE-DO
Baseline [25]	SECOND WALK MOISTURE EXPLAIN REASON WELL GO
Ours	US **IDEAS BOTH-FORWARD OPEN-MINDED** AFFECT **EVERYTHING WE-DO**
Ours w/ LLM	**WE-ALL IDEAS BOTH-FORWARD OPEN-MINDED** AFFECT **EVERYTHING WE-DO**

5 Conclusion

In this work, we introduced a novel framework for sign spotting that integrates an LLM and uses a modified beam search decoding approach. By combining LLM-derived linguistic priors with visually grounded emission probabilities, our approach effectively disambiguates gloss sequences. The efficacy of this method was demonstrated by a significant reduction in Top-1 WER on synthetic data from 100% to under 63%. Meanwhile, on real-world sign language videos, our dictionary-matching system, coupled with I3D and ResNeXt101 feature fusion, lowered Top-1 WER from 90.89% to 47.25%. The subsequent integration of the LLM disambiguation module led to further improvements, achieving a Top-1 WER of 44.73% and a Top-5 WER of 34.81%. Qualitative analyses complemented these quantitative gains, highlighting clear advantages over baseline methods. While acknowledging limitations such as dictionary noise, our findings strongly suggest that sign spotting approaches can be advanced through the introduction of linguistic contexts.

Acknowledgments: This work was supported by the SNSF project 'SMILE II' (CRSII5 193686), the Innosuisse IICT Flagship (PFFS-21-47), EPSRC grant APP24554 (SignGPT-EP/Z535370/1) and through funding from Google.org via the AI for Global Goals scheme. This work reflects only the author's views and the funders are not responsible for any use that may be made of the information it contains.

References

[1] Samuel Albanie, Gül Varol, Liliane Momeni, Hannah Bull, Triantafyllos Afouras, Himel Chowdhury, Neil Fox, Bencie Woll, Rob Cooper, and Andrew Zisserman McParland. 2021. BBC-Oxford British Sign Language Dataset. *arXiv preprint arXiv:2111.03635* (2021).

[2] Sally Beatty. 2024. Tiny but Mighty: The Phi-3 Small Language Models with Big Potential. https://news.microsoft.com/source/features/ai/the-phi-3-small-language-models-with-big-potential/. Accessed: 6-6-2025.

[3] Donald J. Berndt and James Clifford. 1994. Using Dynamic Time Warping to Find Patterns in Time Series. In *Proceedings of the 3rd international conference on knowledge discovery and data mining*.

[4] Necati Cihan Camgoz, Simon Hadfield, Oscar Koller, Hermann Ney, and Richard Bowden. 2018. Neural Sign Language Translation. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 7784–7793.

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

[6] Yutong Chen, Fangyun Wei, Xiao Sun, Zhirong Wu, and Stephen Lin. 2022. A simple multi-modality transfer learning baseline for sign language translation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 5120–5130.

[7] Yutong Chen, Ronglai Zuo, Fangyun Wei, Yu Wu, Shujie Liu, and Brian Mak. 2022. Two-stream network for sign language recognition and translation. *Advances in Neural Information Processing Systems* 35 (2022), 17043–17056.

[8] Amanda Duarte, Shruti Palaskar, Lucas Ventura, Deepti Ghadiyaram, Kenneth DeHaan, Florian Metze, Jordi Torres, and Xavier Giro-i Nieto. 2021. How2sign: a large-scale multimodal dataset for continuous american sign language. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2735–2744.

[9] Jordan Fenlon, Adam Schembri, Ramas Rentelis, David Vinson, and Kearsy Cormier. 2014. Using conversational data to determine lexical frequency in British Sign Language: The influence of text type. *Lingua* 143 (2014), 187–202.

[10] Lianyu Hu, Wei Feng, Liqing Gao, Zekang Liu, and Liang Wan. 2024. Cornet+: Sign language recognition and translation via spatial-temporal correlation. *arXiv preprint arXiv:2404.11111* (2024).

[11] Tao Jiang, Necati Cihan Camgöz, and Richard Bowden. 2021. Looking for the signs: Identifying isolated sign instances in continuous video footage. In *2021 16th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2021)*. IEEE, 1–8.

[12] Oscar Koller, Hermann Ney, and Richard Bowden. 2016. 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*. 3793–3802.

[13] Reiner Konrad, Thomas Hanke, Gabriele Langer, Dolly Blanck, Julian Bleicken, Ilona Hofmann, Olga Jeziorski, Lutz König, Susanne König, Rie Nishio, et al. 2020. MEINE DGS – annotiert. Öffentliches Korpus der Deutschen Gebärdensprache, 3. Release / MY DGS – Annotated. Public Corpus of German Sign Language, 3rd release.

[14] Zecheng Li, Wengang Zhou, Weichao Zhao, Kepeng Wu, Hezhen Hu, and Houqiang Li. 2025. Uni-sign: Toward unified sign language understanding at scale. *arXiv preprint arXiv:2501.15187* (2025).

[15] JianHe Low, Ozge Mercanoglu Sincan, and Richard Bowden. 2025. SAGE: Segment-Aware Gloss-Free Encoding for Token-Efficient Sign Language Translation. *arXiv preprint arXiv:2507.09266* (2025).

[16] JianHe Low, Harry Walsh, Ozge Mercanoglu Sincan, and Richard Bowden. 2025. Hands-On: Segmenting Individual Signs from Continuous Sequences. In *The 19th IEEE International Conference on Automatic Face and Gesture Recognition (FG)*.

[17] Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Uboweja, Michael Hays, Fan Zhang, Chuo-Ling Chang, Ming Guang Yong, Juhyun Lee, et al. 2019. Mediapipe: A framework for building perception pipelines. *arXiv preprint arXiv:1906.08172* (2019).

[18] Liliane Momeni, Gul Varol, Samuel Albanie, Triantafyllos Afouras, and Andrew Zisserman. 2020. Watch, read and lookup: learning to spot signs from multiple supervisors. In *Proceedings of the Asian Conference on Computer Vision*.

[19] Meinard Müller. 2007. *Information retrieval for music and motion*. Springer.

[20] National Deaf Children's Society. 2024. What is Sign Language? https://www.ndcs.org.uk/information-and-support/language-and-communication/sign-language/what-is-sign-language/. Accessed: 22.03.2024.

[21] Adrián Núñez-Marcos, Olatz Perez-de Viñaspre, and Gorka Labaka. 2023. A survey on Sign Language machine translation. *Expert Systems with Applications* 213 (2023), 118993.

[22] Eng-Jon Ong, Oscar Koller, Nicolas Pugeault, and Richard Bowden. 2014. Sign spotting using hierarchical sequential patterns with temporal intervals. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 1923–1930.

[23] Sense. [n. d.]. Sign Language. https://www.sense.org.uk/information-and-advice/communication/sign-language/. Accessed: 22.03.2024.

[24] Bowen Shi, Diane Brentari, Greg Shakhnarovich, and Karen Livescu. 2022. Open-domain sign language translation learned from online video. *arXiv preprint arXiv:2205.12870* (2022).

[25] Ozge Mercanoglu Sincan, Necati Cihan Camgoz, and Richard Bowden. 2024. Using an LLM to turn sign spottings into spoken language sentences. *arXiv preprint arXiv:2403.10434* (2024).

[26] Garrett Tanzer and Biao Zhang. 2024. YouTube-SL-25: A Large-Scale, Open-Domain Multilingual Sign Language Parallel Corpus. *arXiv preprint arXiv:2407.11144* (2024).

[27] Rachel Tatman. 2017. English Word Frequency. https://www.kaggle.com/datasets/rtatman/english-word-frequency. Accessed: 30-6-2024.

[28] Gemma Team. 2024. Gemma. doi:10.34740/KAGGLE/M/3301

[29] The Word Finder. 2024. Random Sentence Generator. https://www.thewordfinder.com/random-sentence-generator/. Accessed: 1-8-2024.

[30] Gül Varol, Liliane Momeni, Samuel Albanie, Triantafyllos Afouras, and Andrew Zisserman. 2022. Scaling up sign spotting through sign language dictionaries. *International Journal of Computer Vision* 130, 6 (2022), 1416–1439.

[31] Ville Viitaniemi, Tommi Jantunen, Leena Savolainen, Matti Karppa, and Jorma Laaksonen. 2014. S-pot–a benchmark in spotting signs within continuous signing. In *LREC proceedings*. European Language Resources Association (LREC).

[32] Ryan Wong, Necati Cihan Camgöz, and Richard Bowden. 2022. Hierarchical i3d for sign spotting. In *European Conference on Computer Vision Workshops*. Springer, 243–255.

[33] Ryan Wong, Necati Cihan Camgöz, and Richard Bowden. 2024. Sign2GPT: Leveraging Large Language Models for Gloss-Free Sign Language Translation. In *The Twelfth International Conference on Learning Representations (ICLR)*.

[34] Hee-Deok Yang, Stan Sclaroff, and Seong-Whan Lee. 2008. Sign language spotting with a threshold model based on conditional random fields. *IEEE transactions on pattern analysis and machine intelligence* 31, 7 (2008), 1264–1277.

[35] Biao Zhang, Mathias Müller, and Rico Sennrich. 2023. SLTUNET: A simple unified model for sign language translation. *arXiv preprint arXiv:2305.01778* (2023).

[36] Benjia Zhou, Zhigang Chen, Albert Clapés, Jun Wan, Yanyan Liang, Sergio Escalera, Zhen Lei, and Du Zhang. 2023. Gloss-free sign language translation: Improving from visual-language pretraining. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 20871–20881.

[37] Hao Zhou, Wengang Zhou, Weizhen Qi, Junfu Pu, and Houqiang Li. 2021. Improving sign language translation with monolingual data by sign back-translation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 1316–1325.