WSLP 2025

First Workshop on Sign Language Processing (WSLP)

Proceedings of the Workshop

December 20-24, 2025

Order copies of this and other ACL proceedings from:

Association for Computational Linguistics (ACL)
317 Sidney Baker St. S
Suite 400 - 134
Kerrville, TX 78028
USA
Tel: +1-855-225-1962
acl@aclweb.org

ISBN 979-8-89176-304-3

Introduction

We are pleased to present the Proceedings of the First Workshop on Sign Language Processing (WSLP 2025), held in conjunction with IJCNLP–AACL 2025 at IIT Bombay. This volume brings together research contributions, shared-task system descriptions, and community perspectives that reflect the rapidly growing interest in computational approaches to signed languages, particularly those that have historically been under-resourced.

This workshop was conceived with a clear goal: to broaden the scope of sign language technology beyond the few dominant datasets and to foreground linguistic, geographic, and community diversity. Indian Sign Language (ISL)—one of the world's largest and most vibrant sign languages—has long lacked large-scale, publicly available resources. However, with recent efforts such as the iSign corpus and CISLR, reproducible research and benchmark creation have become possible, enabling WSLP to establish the first public leaderboards for ISL-to-English translation, isolated sign recognition, and word-presence prediction.

These proceedings include ten accepted papers, covering a wide spectrum of themes:

- Creation of new multimodal and multilingual sign language datasets. - Motion-aware and pose-based modelling for continuous sign language translation. - Cross-linguistic phonological analysis. - Data augmentation strategies using large language models. - Lightweight real-time systems tailored to regional languages and low-resource environments.

The workshop also featured a three-track shared task on Indian Sign Language, hosted on Codabench. By releasing open datasets, encouraging pose-only, privacy-aware modelling, and lowering computational barriers, the shared task has set a new foundation for reproducible and equitable research in ISL processing.

We are grateful to our invited speakers, Dr. Amit Moryossef and Dr. Andesha Mangla, whose talks provided deep insight into the future of sign language technology, transcription systems, and the role of ISL in Deaf education. Their contributions underscored the importance of bridging technical innovation with linguistic expertise and community needs.

We also thank the IJCNLP–AACL 2025 Organizing Committee for their support, the reviewers for their thoughtful evaluations, and the many participants whose enthusiasm and contributions made this workshop possible. Most importantly, we acknowledge the Deaf community and the ISL interpreters, educators, and linguists whose work and guidance remain central to the advancement of sign language technology.

We hope these proceedings will serve as a resource for researchers, developers, and community members working toward inclusive, equitable, and deployable sign-language AI.

ISBN: 979-8-89176-304-3

WSLP 2025 Organizing Committee:

Sanjeet Singh, Abhinav Joshi, Keren Artiaga, Mohammed Hasanuzzaman, Facundo Manuel Quiroga, Sabyasachi Kamila, and Ashutosh Modi

Program Committee

Program Chairs

Keren Artiaga, ADAPT Centre, Munster Technological University, Ireland
Mohammed Hasanuzzaman, The Queen's University Belfast and ADAPT Centre, Munster Technological University, Ireland
Abhinav Joshi, Indian Institute of Technology, Kanpur, India
Sabyasachi Kamila, Manipal Institute of Technology Bengaluru, MAHE, Manipal, India
Ashutosh Modi, Indian Institute of Technology, Kanpur, India
Facundo Manuel Quiroga, Universidad Nacional de La Plata, Argentina
Sanjeet Singh, Indian Institute of Technology, Kanpur, India

Table of Contents

Workshop on Sign Language Processing (WSLP) and Shared Task

Sanjeet Singh[1], Abhinav Joshi[1], Keren Artiaga[2], Mohammed Hasanuzzaman[2,3], Facundo Manuel Quiroga[4], Sabyasachi Kamila[5] Ashutosh Modi[1]

[1]IIT Kanpur, India, [2]ADAPT Centre, MTU, Ireland,
[3]Queen's University Belfast, UK, [4]Universidad Nacional de La Plata, Argentina,
[5]Manipal Institute of Technology Bengaluru, Manipal Academy of Higher Education (MAHE), India
{sanjeet, ajoshi, ashutoshm}@cse.iitk.ac.in,
keren.artiaga@adaptcentre.ie, m.hasanuzzaman@qub.ac.uk,
sabyasachi.kamila@manipal.edu, fquiroga@lidi.unlp.edu.ar

Abstract

We organized the First Workshop on Sign Language Processing (WSLP 2025), co-located with IJCNLP-AACL 2025 at IIT Bombay, to bring together researchers, linguists, and members of the Deaf community and accelerate computational work on under-resourced sign languages.

Workshop accepted 10 papers (including 2 official shared-task submissions) that introduced new large-scale resources (continuous ISL fingerspelling corpus, cross-lingual HamNoSys corpora), advanced multilingual and motion-aware translation models, explored LLM-based augmentation and glossing strategies, and presented lightweight deployable systems for regional languages such as Odia.

We ran a three-track shared task on Indian Sign Language that attracted over 60 registered teams and established the first public leaderboards for sentence-level ISL-to-English translation, isolated word recognition, and word-presence prediction.

By centring geographic, linguistic, and organiser diversity, releasing open datasets and benchmarks, and explicitly addressing linguistic challenges unique to visual-spatial languages, we significantly broadened the scope of sign-language processing beyond the traditional European and East-Asian datasets, laying a robust foundation for inclusive, equitable, and deployable sign-language AI in the Global South.

1 Introduction

Sign languages are the primary means of communication for millions of Deaf and Hard-of-Hearing individuals worldwide, yet they remain among the most under-resourced modalities in natural language processing (Yin et al., 2021; Moryossef et al., 2020; Jiang et al., 2024). Unlike spoken languages, which are produced and perceived sequentially, sign languages are inherently visual-gestural and multi-channel: manual features (handshape, orientation, movement, location) combine simultaneously with non-manual markers (facial expressions, head tilts, eye gaze, torso shifts) to convey lexical, morphological, and syntactic information (Brentari, 2019).

Until recently, the absence of large, publicly available video corpora with aligned translations severely limited systematic research on Indian Sign Language (ISL), one of the world's major sign languages. The release of the iSign (Joshi et al., 2024) corpus in 2024—containing over 118,000 ISL–English sentence pairs—finally made reproducible, large-scale experiments possible.

We therefore organized the First Workshop on Sign Language Processing (WSLP 2025), co-located with IJCNLP-AACL 2025 at IIT Bombay, to capitalise on this breakthrough and to create a dedicated venue for sign-language technology in South Asia and beyond. We placed particular emphasis on linguistic diversity, lightweight and privacy-preserving modelling, and close collaboration with the Deaf community.

A central component of the workshop was a three-track shared task on ISL that we designed to establish the first public benchmarks while encouraging approaches suitable for real-world deployment in resource-constrained settings:

1. **Task 1 – ISL → English Translation**: end-to-end translation of continuous signing (video or pose sequences) into written English.

2. **Task 2 – Isolated Word/Gloss Recognition**: classification of short, single-sign clips into one of thousands of lexical categories.

3. **Task 3 – Word Presence Prediction**: binary

Proceedings of the Workshop on Sign Language Processing (WSLP), pages 1–9

decision on whether a query word appears anywhere in a full sentence video (a lightweight sign-spotting formulation).

By supplying both raw video and pre-extracted MediaPipe pose keypoints and hosting evaluation on Codabench, we lowered barriers to entry and enabled fair comparison across diverse modelling choices. Fig. 2 shows an Indian Sign Language (ISL) signer producing the sign alongside the corresponding pose-keypoint sequence extracted with MediaPipe.

We accepted a total of 10 papers (8 main-conference contributions and 2 official shared-task submissions). These span new large-scale resources (continuous ISL fingerspelling, cross-lingual HamNoSys corpora), multilingual and motion-aware translation architectures, LLM-based data augmentation, glossing strategies, and deployable regional systems (e.g., ISL-to-Odia).

This report summarises the workshop contributions, presents the official shared-task results, analyses the most effective techniques, and outlines immediate research directions for the community.

The rest of the paper is organised as follows. Section 2 discusses a few prominent Continuous Sign language datasets. Section 3 provides an overview of the accepted papers. Section 4 describes the shared task and its datasets. Section 5 discusses linguistic and computational challenges, Section 6 discusses the conclusion and future work, while Section 7 addresses diversity, inclusion, and ethical considerations.

2 Related Work

Sign language translation (SLT) has been predominantly focused on a small number of datasets, such as RWTH-PHOENIX-Weather 2014T (Camgöz et al., 2018) for German Sign Language and CSL-Daily (Zhou et al., 2021; Hu et al., 2023, 2021) for Chinese Sign Language. This concentration has led to disproportionate attention on these languages, creating a self-perpetuating cycle where new projects prioritise these datasets for relevance and impact. As a result, many other sign languages, including Indian Sign Language (ISL), remain severely under-resourced.

While a few studies have explored SLT beyond these benchmarks, they are limited. For example, Joshi et al. (2023) and Joshi et al. (2024) investigated ISL, while Lin et al. (2023) examined American Sign Language (ASL). Comprehensive surveys of SLT research can be found in Liang et al. (2023) and Núñez-Marcos et al. (2023), which highlight the field's challenges, including data scarcity and the need for multimodal approaches.

Table 1 summarises key continuous SLT datasets, illustrating the diversity in hours, vocabulary size, and signers across various sign languages.

This disparity in research focus is exactly what motivated us to organise WSLP 2025. By centring on low-resource sign languages like ISL and promoting collaborations, we aim to address these imbalances and foster more equitable development of sign-language technologies.

3 Overview of Accepted Papers

We accepted eight papers at WSLP 2025. The contributions naturally cluster into four main thematic areas:

Resource Creation and Linguistic Analysis We received two papers that significantly expand publicly available resources. One introduces the first large-scale continuous fingerspelling corpus for Indian Sign Language, extracted from public news broadcasts and validated by a professional ISL interpreter. The other constructs a balanced 4,000-sign HamNoSys corpus across British, German, French, and Greek Sign Languages and presents the first large-scale cross-linguistic phonological similarity analysis using normalised edit distance.

Gloss-Related Translation Two papers push the boundaries of low-resource glossing and translation. One systematically evaluates fine-tuning of large pre-trained language models (T5, Flan-T5, mBART, Llama) on multiple public gloss datasets, establishing new state-of-the-art results and revealing a stark performance asymmetry between gloss-to-text and text-to-gloss directions. The second shows that oversampling related high-resource English→ASL gloss pairs dramatically improves Spanish→Mexican Sign Language glossing, lifting BLEU from 62 to 85 on a small 3,000-sentence corpus.

Continuous Sign Language Translation and Multilingual Modelling Three papers focus on end-to-end translation. One proposes a motion-aware architecture that explicitly incorporates optical flow and achieves the current best published BLEU-4 of 8.58 on the open-domain iSign test set. The remaining two explore multilingual training across German, Greek, Argentinian, and Indian

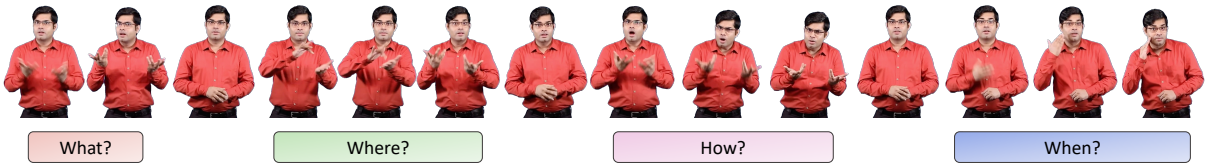

Figure 1: An ISL signer demonstrating the simultaneous use of manual signs, facial non-manuals, and signing space. Words (“What", "where", "How", and "when”) are expressed through coordinated handshape, movement, eye gaze, and head tilt. (Joshi et al., 2024)

Year	Dataset	Sign Language	Hours	# Videos	# Vocab	# Signers	Source
2020	GSL	Greek	9.51	10,295	310	7	(Adaloglou et al., 2021)
2018	KETI	Korean	28	14,672	419	14	(Ko et al., 2018)
2022	LSA-T	Argentine	21.78	14,880	14,239	103	(Dal Bianco et al., 2022)
2021	How2Sign	ASL	79	35,191	15,686	9	(Duarte et al., 2021)
2022	Open-ASL	ASL	288	98,417	33,549	>200	(Shi et al., 2022)
2022	SP-10	Multilingual	14	16,700	79	–	(Hilzensauer and Krammer, 2015)
2023	ISLTranslate	Indian	–	31,000	11,000	–	(Joshi et al., 2023)
2024	iSign	Indian	252	118,000	40,000	–	(Joshi et al., 2024)
2021	BBC-Oxford	British	1,467	–	2,281	39	(Albanie et al., 2021)
2021	SWISSTXT-NEWS	Swiss-German	9	181	10,561	–	(Camgöz et al., 2021)
2021	VRT-NEWS	Flemish	9	120	6,875	–	(Camgöz et al., 2021)
2010	SIGNUM	German	55.3	33,210	450	25	(von Agris and Kraiss, 2010)

Table 1: Summary of a few prominent continuous sign language translation datasets

Sign Languages using unified pose representations; they demonstrate that joint pre-training followed by short language-specific fine-tuning outperforms monolingual baselines on three of the four corpora, and that LLM-based text-side paraphrasing yields consistent gains on medium-scale datasets.

Regional and Deployable Systems One paper presents a real-time 12-class ISL recognition system that translates directly into Odia script using a lightweight 2D CNN and MediaPipe pipeline. With 98.33% accuracy and explicit optimisation for low-resource devices, it is designed specifically for rural and educational deployment contexts.

Below is a concise overview of the eight accepted papers: Table 2 provides brief overview of accepted papers at the First Workshop on Sign Language Processing (WSLP 2025).

Finetuning Pre-trained Language Models for Bidirectional Sign Language Gloss to Text Translation

This work presents the first large-scale bidirectional evaluation of modern pre-trained language models (T5, Flan-T5, mBART, and Llama) on gloss-to-text and text-to-gloss translation. Using three established datasets (RWTH-PHOENIX-Weather 2014T, SIGNUM, and ASLG-PC12), fine-tuned PLMs consistently and significantly outperform Transformers trained from scratch, establishing new state-of-the-art results. The study highlights a striking performance asymmetry: text-to-gloss translation remains far more difficult than the reverse direction, underscoring the value of leveraging massive textual pre-training for low-resource sign-language tasks.

Cross-Linguistic Phonological Similarity Analysis in Sign Languages Using HamNoSys

A balanced corpus of 4,000 signs (1,000 each from British, German, French, and Greek Sign Languages) is encoded in HamNoSys. Normalised edit distance is then used to compute intra- and inter-language phonological similarity. The analysis reveals both universal tendencies (e.g., frequent handshapes and movement types) and language-specific patterns in non-manual features and spatial articulation, offering the first quantitative typological insights into sign-language phonology at this scale.

Enhancing Indian Sign Language Translation via Motion-Aware Modeling

This paper benchmarks existing sign-language translation architectures on Indian Sign Language and introduces SpaMo-OF, a model that explicitly integrates dense optical-flow motion cues with multi-scale spatial features. The approach achieves a BLEU-4 score of 8.58 on the open-domain iSign test set—currently the highest published result for continuous ISL-to-English translation—and establishes a strong, reproducible baseline for future work on ISL.

Continuous Fingerspelling Dataset for Indian Sign Language

The first large-scale continuous fingerspelling

corpus for ISL is released, comprising 1,308 real-world segments (70.85 minutes, 14,814 characters) extracted from ISH News broadcasts with synchronized on-screen text. Professional interpreter validation yields 90.67% exact-match accuracy on a 150-sample subset. A ByT5-small baseline achieves 82.91% character error rate after fine-tuning, and the dataset is made publicly available to support transcription, localisation, and generation tasks.

Multilingual Sign Language Translation with Unified Datasets and Pose-Based Transformers

A single pose-based transformer is trained jointly on four typologically diverse sign languages (German, Greek, Argentinian, and Indian). A simple two-stage training schedule—multilingual pre-training followed by short language-specific fine-tuning—outperforms monolingual baselines on three of the four corpora and narrows the gap on the fourth, providing clear evidence of effective cross-lingual transfer in extremely low-resource sign-language translation.

Augmenting Sign Language Translation Datasets with Large Language Models

GPT-4 is used to generate high-quality paraphrases of target-language sentences for text-side data augmentation. On the medium-scale PHOENIX14T corpus, augmentation raises BLEU-4 from 9.56 to 10.33. Results are mixed on smaller or heavier-tailed datasets, leading to a detailed analysis of when LLM-based augmentation is most beneficial—primarily for corpora with richer, longer-tail vocabularies.

Low-Resource Sign Language Glossing Profits From Data Augmentation

Spanish-to-Mexican Sign Language glossing on a tiny 3,000-sentence corpus is dramatically improved (BLEU 62 $\rightarrow$ 85) simply by oversampling related English-to-American Sign Language gloss pairs at roughly 4× ratio. The study demonstrates that cross-sign-language data augmentation is a powerful, language-agnostic technique for extreme low-resource scenarios.

Indian Sign Language Recognition and Translation into Odia

A real-time, lightweight 12-class ISL recognition system is developed using MediaPipe keypoints and a 2D CNN, achieving 98.33% accuracy on a diverse custom dataset. Recognised signs are mapped to Odia script via a curated dictionary and displayed in a simple GUI. Explicitly optimised for low-cost devices, the system targets rural classrooms and regional accessibility needs in Odisha.

Taken together, these eight contributions illustrate a vibrant, rapidly maturing research community that is increasingly focused on multilingual modelling, pose-based efficiency, community-driven resource creation, and deployable solutions tailored to the linguistic and infrastructural realities of the Global South.

4 Shared Task

A major highlight of WSLP 2025 was the three-track shared task on Indian Sign Language processing that we hosted on Codabench to ensure transparent, reproducible evaluation. We deliberately designed the tasks to cover the full spectrum from isolated recognition to continuous translation and sign spotting, while encouraging lightweight, privacy-preserving models that can eventually run on low-cost devices.

The shared task attracted strong interest: 33 teams registered for Task 1, 12 for Task 2, and 15 for Task 3. Given the specialised nature of the domain and the non-trivial cost of training on large video collections, only a subset of teams completed the full submission cycle (August 15 – October 15, 2025). From these, we accepted two outstanding system-description papers into the main workshop proceedings.

The three tasks were as follows:

Task 1: ISL to English Translation Participants developed end-to-end systems that translate continuous signing (raw video or pose sequences) into written English sentences. The training data were obtained from the public iSign corpus (Joshi et al., 2024) (118,000 sentence pairs). We have also released a new high-quality evaluation split, scraped from public YouTube sources and manually cleaned, consisting of 5,278 sentence pairs for validation and 5,252 for testing. Primary metrics were BLEU-4, ROUGE-L, and chrF.

Task 2: Isolated Word/Gloss Recognition Systems classified short, single-sign clips into one of several thousand lexical categories—an essential building block for dictionaries, annotation tools, and lookup applications. Following the methodology of Joshi et al. (2022), we curated a new dataset by scraping publicly available ISL YouTube content and manually cleaning it: The Dataset contains 4,398 training examples, 109 validation examples, and 526 test examples. Evaluation used Top-1, Top-5, and Top-10 accuracy.

Paper Title	Main Contribution / Task
Augmenting Sign Language Translation Datasets with Large Language Models	LLM-based text-side paraphrasing for SLT augmentation (PHOENIX14T, GSL, LSA-T)
Continuous Fingerspelling Dataset for Indian Sign Language	First large continuous ISL fingerspelling corpus (1,308 segments, 70.85 min) + ByT5 baseline
Cross-Linguistic Phonological Similarity Analysis in Sign Languages Using HamNoSys	4,000-sign HamNoSys corpus (BSL, DGS, LSF, GSL) + edit-distance phonological analysis
Enhancing Indian Sign Language Translation via Motion-Aware Modeling	SpaMo-OF architecture with optical flow; BLEU-4 8.58 on iSign (strongest ISL baseline)
Finetuning Pre-trained Language Models for Bidirectional Sign Language Gloss to Text Translation .	Bidirectional PLM fine-tuning (T5, mBART, Llama) → new SOTA on glosstext
Indian Sign Language Recognition and Translation into Odia	Real-time 12-class ISL→Odia system (98.33% acc.) for low-resource deployment
Low-Resource Sign Language Glossing Profits From Data Augmentation	Cross-SL data augmentation (ASL→MSL glossing); BLEU 62→85
Multilingual Sign Language Translation with Unified Datasets and Pose-Based Transformers	Multilingual pose-based model (DE, EL, AR, IN SL); outperforms monolingual on 3/4 corpora

Table 2: Overview of accepted papers at the First Workshop on Sign Language Processing (WSLP 2025)

Task 3: Word Presence Prediction (Sign Spotting) Given a query word and a full sentence video (or pose sequence), systems predicted whether the query appears anywhere in the sentence. The dataset, created by scraping ISL YouTube videos and manually cleaned, inspired by Joshi et al. (2024), comprises of 25,432 training pairs, 1,413 validation pairs, and 1,413 test pairs. The evaluation metrics used were Accuracy, macro Precision, Recall, and F1.

The two accepted shared-task papers represent the top official submissions:

- For Task 2 (Isolated Word/Gloss Recognition), the accepted paper used a lightweight pose-only Temporal Convolutional Network and achieved 54.00% Top-1 and 78.00% Top-5 accuracy on the 4,361-class test set. - For Task 3 (Word Presence Prediction), the accepted paper proposed an end-to-end pose encoder with a binary classification head, obtaining 61.88% accuracy and 60.00% macro F1.

Task 1 received no official system-description paper, but several workshop contributions report strong independent results on the same data.

Full details, baselines, leaderboards, and data access are available at the official shared-task website: `https://exploration-lab.github.io/ISL-Shared-Task/`.

We believe the shared task not only produced the first public, large-scale benchmarks for these three core ISL subtasks but also confirmed the practical advantages of pose-only modelling for speed, robustness, and privacy—advantages that will be crucial for real-world deployment in classrooms and rural areas across India.

5 Linguistic and Computational Challenges of Indian Sign Language

Sign Languages, like all natural sign languages, differ fundamentally from spoken languages in their visual-gestural modality (Sinha, 2017; Brentari, 2019). These differences are not merely superficial—they profoundly affect data collection, annotation, modelling assumptions, and evaluation. Below we outline the major linguistic and computational challenges that emerged repeatedly across workshop/Shared task discussions and that any future Indian Sign Language system must explicitly address. Fig. 1 shows an ISL signer demonstrating the simultaneous use of manual signs, facial non-manuals, and signing space. Words (“What", "where", "How", and "when”) are expressed through coordinated handshape, movement, eye gaze, and head tilt.

5.1 Visual-Spatial Grammar and Simultaneous Articulation

Unlike spoken languages, which are produced and perceived linearly, ISL exploits three-dimensional signing space and multiple independent articulators (two hands, head, torso, eyebrows, mouth, eye gaze) (Sinha, 2017). A single utterance routinely conveys information in parallel:

- The dominant hand may articulate a lexical sign while the non-dominant hand functions as a classifier depicting shape or location (Sinha, 2017).
- Facial non-manuals simultaneously mark questions, negation, conditionals, or intensity (Sinha, 2017).

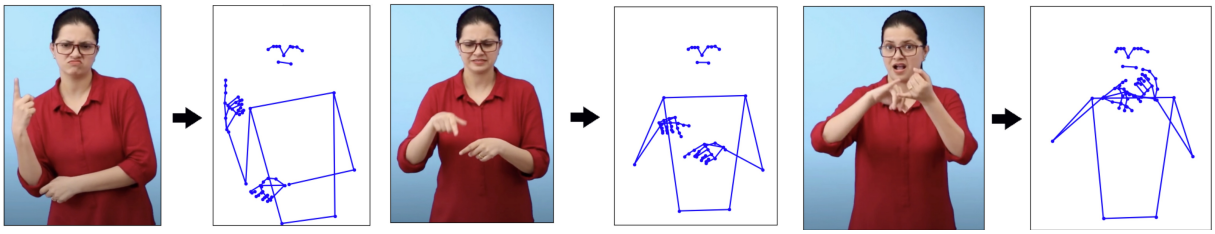

Figure 2: An Indian Sign Language (ISL) signer producing the sign alongside the corresponding pose-keypoint sequence extracted with MediaPipe Holistic (right).

- Eye gaze and head tilt establish and maintain reference points in the signing space (Sinha, 2017).

Sequential architectures (Transformers, RNNs) that dominate spoken-language NLP struggle to capture these inherently parallel dependencies without explicit spatial modelling.

5.2 Use of Signing Space for Reference Tracking

ISL heavily relies on spatial loci: entities (people, objects, places) are assigned arbitrary locations in the signing space upon first mention and subsequently referred to by pointing or verb directionality (Sinha, 2017). The same lexical sign can change meaning entirely depending on its start and end locus (e.g., GIVE from locus A to B vs. B to A). This dynamic, discourse-dependent reference system makes sentence-level training data highly context-sensitive and complicates the creation of truly parallel corpora.

5.3 Crucial Role of Non-Manual Features

Non-manual markers are not optional prosody—they are grammatical (Sinha, 2017). Raised eyebrows mark yes/no questions and conditionals; furrowed brows mark wh-questions; headshakes spread over entire clauses for negation. Current pose-based pipelines (MediaPipe, OpenPose) capture hand and body keypoints accurately, while discarding most facial information. As a result, even high-accuracy isolated-sign recognisers often produce grammatically incorrect continuous output.

5.4 Productive Morphology and Classifier Constructions

ISL exhibits highly productive classifier predicates: handshapes representing semantic classes (human, vehicle, flat object, etc.) combine with motion, location, and orientation morphemes to describe complex spatial events (Sinha, 2017). A single classifier construction can convey "two cars passing each other on a narrow bridge" without any lexical signs. Standard tokenisation and vocabulary strategies borrowed from spoken languages collapse this rich morphology into rare or out-of-vocabulary tokens.

5.5 Fingerspelling and Name Signs

Proper names, technical terms, and new vocabulary are fingerspelled letter-by-letter (Sinha, 2017). Fingerspelling sequences are fast, co-articulated, and highly variable across signers. Moreover, once introduced, names are immediately replaced by arbitrary short signs—often based on physical characteristics or initials (Sinha, 2017). This creates extreme coreference complexity within a single video.

5.6 Dialectal and Sociolectal Variation

India's linguistic diversity extends to ISL: vocabulary for numbers, colours, kinship terms, and food varies significantly across regions (Delhi, Bengal, Tamil Nadu, Kerala) (Jepson, 1991; Zeshan, 2003; Johnson and Johnson, 2008; Zeshan et al., 2023). Age, education, and degree of contact with Deaf schools further influence signing style. Anecdotal evidence suggests that around 75% of vocabulary is similar nationwide, with the remaining 25% showing high regional variation (Jepson, 1991; Zeshan, 2003; Johnson and Johnson, 2008; Zeshan et al., 2023). A model trained primarily on Delhi-region signers may perform poorly elsewhere.

5.7 Data Scarcity and Ethical Annotation

Even the largest public ISL corpus ((Joshi et al., 2024), 118k sentences) is orders of magnitude smaller than typical spoken-language corpora. Gloss annotation requires fluent Deaf annotators, who are scarce and expensive. Automatic alignment between video and text remains an open research problem.

These linguistic realities explain why direct application of spoken-language architectures yields disappointing results on ISL and why pose-only models—despite their computational efficiency—still fall short on grammatical correctness. Addressing them will require hybrid approaches that combine spatial graph networks, dedicated non-manual feature extractors, classifier-aware tokenisation, and continued collaboration with Deaf linguists and community members.

6 Conclusion and Future Directions

We believe the First Workshop on Sign Language Processing marked a pivotal moment for research on Indian Sign Language and other under-resourced sign languages. By bringing together 10 high-quality papers, three competitive shared tasks, and a diverse international community, we established the first public benchmarks for continuous ISL translation, isolated recognition, and sign spotting, while releasing substantial new datasets and reproducible baselines.

More importantly, we demonstrated that rapid and meaningful progress is possible when linguistic expertise, community-driven data collection, and efficient pose-based modelling converge. By deliberately prioritising geographic and linguistic diversity, close collaboration with Deaf scholars, and solutions viable in low-resource environments, we have helped shift sign-language technology from a narrow focus on a few high-resource languages toward a genuinely inclusive, global endeavour.

Looking ahead, we see the most pressing challenges in integrating non-manual features, properly modelling spatial grammar and classifier constructions, expanding multi-dialect and multilingual transfer, and developing evaluation protocols grounded in native-signer judgements. Sustained community involvement, privacy-preserving data collection, and lightweight real-time systems remain essential if we are to translate research gains into real-world impact in classrooms, homes, and workplaces across India and beyond. We intend to build on this momentum by establishing WSLP as an annual venue that continues to drive equitable and deployable sign-language technology.

7 Diversity, Inclusion, and Ethical Considerations

We explicitly designed WSLP 2025 to promote diversity, equity, and ethical practice in sign-language technology.

By centring the shared task on Indian Sign Language—one of the world's largest yet most under-resourced sign languages—and by encouraging work on other non-dominant varieties (Mexican, Argentinian, Greek, etc.), we deliberately broke the historical dominance of a handful of European and East-Asian datasets. The accepted papers and shared-task submissions introduced new resources and benchmarks that create tangible opportunities for researchers from the Global South and for Deaf-led innovation.

Our organizing committee itself reflects broad geographic and institutional diversity, with members based in India, Argentina, Ireland, and the United Kingdom; several of us have long-standing collaborations with Deaf communities and are users of non-dominant sign languages. The authors of the accepted papers come from India, Nigeria, Mexico, Argentina, Germany, and the United States, ensuring a wide range of linguistic and socio-cultural perspectives.

At the same time, we are keenly aware of several limitations that remain common in this emerging area:

- Current public ISL corpora are still heavily biased toward educated, urban, northern-Indian signers and specific domains; rural, elderly, and regional-dialect signers remain severely underrepresented. - Automatic metrics (BLEU, accuracy) cannot capture grammatical correctness or cultural appropriateness; large-scale native-signer evaluation was not feasible this year. - Community participation, while stronger than in many previous efforts, was still limited mainly to dataset creation and validation rather than full co-design.

We view these gaps not as shortcomings but as clear priorities for future editions. We are committed to expanding representation, incorporating non-manual features, developing signer-based evaluation protocols, and deepening participatory design with Deaf end-users through sustained partnership with the Indian Sign Language Research and Training Centre (ISLRTC) and regional Deaf associations.

8 Invited Talks

The workshop featured two invited talks that bridged cutting-edge research with real-world impact and community-grounded perspectives on Indian Sign Language.

Amit Moryossef

Bio. Dr. Amit Moryossef is a researcher and entrepreneur in sign-language technology. He completed his Ph.D. at Bar-Ilan University and a postdoc at the University of Zurich. He founded sign.mt, a real-time sign-language translation platform that was recently acquired by Nagish, where he currently leads research. His work has received multiple best-paper awards at ACL and EMNLP.

Title of the talk: The Future of Sign Language Translation is Transcription

Abstract. Sign Language Processing has long been overlooked in mainstream language technology due to the challenges of bridging visual-gestural languages with text-based AI. In this talk, I will show how SignWriting—a universal transcription system—creates a structured, scalable bridge between video-based sign language input and spoken-language text, redefining both translation from sign language and generation into it. Leveraging this framework enables accurate, real-time, and multilingual applications while cleanly separating the roles of computer vision and natural language processing. This division allows researchers to contribute within their own expertise and paves the way for truly inclusive, bidirectional sign-language AI.

Andesha Mangala (Assistant Professor, Indian Sign Language Research and Training Centre, Delhi)

Bio. Dr. Andesha Mangla is an Assistant Professor of Sign Linguistics at the Indian Sign Language Research and Training Center (ISLRTC), a national institute that aims to promote the use of ISL. She completed her PhD in Linguistics from Delhi University, focusing on the role of ISL in deaf education. She has around 15 years' experience in training ISL interpreters and deaf ISL teachers in linguistics and English language, as well as developing resources related to Indian Sign Language, including the ISL Dictionary and ISL translations of NCERT textbooks. Her interest areas include sign linguistics, ISL in deaf education and language teaching.

Title of the talk: ISLRTC's Contributions to Indian Sign Language and Deaf Education

Abstract. The Indian Sign Language Research and Training Centre (ISLRTC) is an autonomous body under the Department of Empowerment of Persons with Disabilities, Ministry of Social Justice and Empowerment, Govt. of India, dedicated to the development and promotion of ISL. Since its establishment in September 2015, ISLRTC has undertaken many initiatives to train manpower in ISL, develop ISL resources, increase ISL accessibility and spread awareness about ISL amongst the general public. ISLRTC's activities aim to achieve the mandates as given in the Rights of Persons with Disabilities Act 2016 and the recommendations of the National Education Policy 2020. Collaborations with government education bodies like NCERT and NIOS have enabled large-scale developments to promote ISL in education, while working with non-governmental organizations including deaf-led organizations like India Signing Hands and Deaf Enabled Foundation, have helped to reach the deaf community. This talk aims to describe the activities of ISLRTC in context of the RPWD Act and NEP.

References

Nikolas Adaloglou, Theocharis Chatzis, Ilias Papastratis, Andreas Stergioulas, Georgios Th Papadopoulos, Vassia Zacharopoulou, George J Xydopoulos, Klimnis Atzakas, Dimitris Papazachariou, and Petros Daras. 2021. A comprehensive study on deep learning-based methods for sign language recognition. *IEEE Transactions on Multimedia*, 24:1750–1762.

Samuel Albanie, Gül Varol, Liliane Momeni, Hannah Bull, Triantafyllos Afouras, Himel Chowdhury, Neil Fox, Bencie Woll, Rob Cooper, Andrew McParland, and Andrew Zisserman. 2021. BOBSL: BBC-Oxford British Sign Language Dataset.

Diane Brentari. 2019. *Sign Language Phonology*. Key Topics in Phonology. Cambridge University Press.

Necati Cihan Camgöz, Ben Saunders, Guillaume Rochette, Marco Giovanelli, Giacomo Inches, Robin Nachtrab-Ribback, and Richard Bowden. 2021. Content4all open research sign language translation datasets. In *2021 16th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2021)*, pages 1–5. IEEE.

Necati Cihan Camgöz, Simon Hadfield, Oscar Koller, Hermann Ney, and Richard Bowden. 2018. Neural sign language translation. In *IEEE Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT.

Pedro Dal Bianco, Gastón Ríos, Franco Ronchetti, Facundo Quiroga, Oscar Stanchi, Waldo Hasperué, and Alejandro Rosete. 2022. Lsa-t: The first continuous argentinian sign language dataset for sign language translation. In *Ibero-American Conference on Artificial Intelligence*, pages 293–304. Springer.

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 (CVPR)*.

Marlene Hilzensauer and Klaudia Krammer. 2015. A multilingual dictionary for sign languages: "spreadthesign". In *Proceedings of the Conference*.

Hezhen Hu, Weichao Zhao, Wengang Zhou, and Houqiang Li. 2023. Signbert+: Hand-model-aware self-supervised pre-training for sign language understanding. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, pages 1–20.

Hezhen Hu, Weichao Zhao, Wengang Zhou, Yuechen Wang, and Houqiang Li. 2021. Signbert: Pre-training of hand-model-aware representation for sign language recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11087–11096.

Jill Jepson. 1991. Urban and rural sign language in india. *Language in Society*, 20.

Zifan Jiang, Gerard Sant, Amit Moryossef, Mathias Müller, Rico Sennrich, and Sarah Ebling. 2024. SignCLIP: Connecting text and sign language by contrastive learning. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, Miami, Florida, USA. Association for Computational Linguistics.

Jane E Johnson and Russell J Johnson. 2008. Assessment of regional language varieties in indian sign language. *SIL Electronic Survey Report*, 6.

Abhinav Joshi, Susmit Agrawal, and Ashutosh Modi. 2023. Isltranslate: Dataset for translating indian sign language. *arXiv preprint arXiv:2307.05440*.

Abhinav Joshi, Ashwani Bhat, Pradeep S, Priya Gole, Shashwat Gupta, Shreyansh Agarwal, and Ashutosh Modi. 2022. CISLR: Corpus for Indian Sign Language recognition. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10357–10366, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Abhinav Joshi, Romit Mohanty, Mounika Kanakanti, Andesha Mangla, Sudeep Choudhary, Monali Barbate, and Ashutosh Modi. 2024. iSign: A benchmark for Indian Sign Language processing. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 10827–10844, Bangkok, Thailand. Association for Computational Linguistics.

Sang-Ki Ko, Jae Gi Son, and Hyedong Jung. 2018. Sign language recognition with recurrent neural network using human keypoint detection. In *Proceedings of the 2018 Conference on Research in Adaptive and Convergent Systems*, pages 326–328, New York, NY, USA. Association for Computing Machinery.

Zeyu Liang, Huailing Li, and Jianping Chai. 2023. Sign language translation: A survey of approaches and techniques. *Electronics*, 12(12).

Kezhou Lin, Xiaohan Wang, Linchao Zhu, Ke Sun, Bang Zhang, and Yi Yang. 2023. Gloss-free end-to-end sign language translation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12904–12916, Toronto, Canada. 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*. Springer.

Adrián Núñez-Marcos, Olatz Perez de Viñaspre, and Gorka Labaka. 2023. A survey on sign language machine translation. *Expert Systems with Applications*, 213:118993.

Bowen Shi, Diane Brentari, Greg Shakhnarovich, and Karen Livescu. 2022. Open-domain sign language translation learned from online video. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Samar Sinha. 2017. *Indian Sign Language: An Analysis of Its Grammar*. Gallaudet University Press.

Ulrich von Agris and Karl-Friedrich Kraiss. 2010. SIGNUM database: Video corpus for signer-independent continuous sign language recognition. In *Proceedings of the LREC2010 4th Workshop on the Representation and Processing of Sign Languages: Corpora and Sign Language Technologies*, pages 243–246, Valletta, Malta. European Language Resources Association (ELRA).

Kayo Yin, Amit Moryossef, Julie Hochgesang, Yoav Goldberg, and Malihe Alikhani. 2021. Including signed languages in natural language processing. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, Online. Association for Computational Linguistics.

Ulrike Zeshan. 2003. Indo-pakistani sign language grammar: a typological outline. *Sign Language Studies*.

Ulrike Zeshan, Nirav Pal, Deepu Manavalamamuni, Ankit Vishwakarma, Sibaji Panda, Jagdish Choudhary, and Inu Aggarwal. 2023. *Indian Sign Language*. National Institute of Open Schooling.

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 (CVPR)*, pages 1316–1325.

Indian Sign Language Recognition and Translation into Odia

Astha Swarupa Nayak[1], Naisargika Subudhi[2], Tannushree Rana[3], Muktikanta Sahu[4], and Rakesh Chandra Balabantaray[5]*
Department of Computer Science and Engineering
International Institute of Information Technology Bhubaneswar
Odisha
[1]B121009@iiit-bh.ac.in, [2]B121031@iiit-bh.ac.in, [3]B421063@iiit-bh.ac.in,
[4]muktikanta@iiit-bh.ac.in, [5]rakesh@iiit-bh.ac.in

Abstract

Sign language is a vital means of communication for the deaf and hard-of-hearing community. However, translating Indian Sign Language (ISL) into regional languages like Odia remains a significant technological challenge due to the languages rich morphology, agglutinative grammar, and complex script. This work presents a real-time ISL recognition and translation system that converts hand gestures into Odia text, enhancing accessibility and promoting inclusive communication. The system leverages MediaPipe for real-time key-point detection and uses a custom-built dataset of 1,200 samples across 12 ISL gesture classes, captured under diverse Indian backgrounds and lighting conditions to ensure robustness. Both 2D and 3D Convolutional Neural Networks (CNNs) were explored, with the 2D CNN achieving superior performance 98.33% test accuracy compared to the 3D CNNs 78.33%. Recognized gestures are translated into Odia using a curated gesture-to-text mapping dictionary, seamlessly integrated into a lightweight Tkinter-based GUI. Unlike other resource-heavy systems, this model is optimized for deployment on low-resource devices, making it suitable for rural and educational contexts. Beyond translation, the system can function as an assistive learning tool for students and educators of ISL. This work demonstrates that combining culturally curated datasets with efficient AI models can bridge communication gaps and create regionally adapted, accessible technology for the deaf and mute community in India.

1 Introduction

Communication is a fundamental aspect of human interaction. However, individuals with hearing and speech impairments face significant barriers in engaging in verbal communication with the wider community. To overcome these challenges, sign language has been developed and adopted as an effective visual-gestural medium of communication. Sign language comprises hand gestures, movements, facial expressions, and body postures to convey meaning. Several physical and dynamic parameters, such as hand shape, hand orientation, motion trajectory, spatial positioning, and non-manual signals like facial expressions, play crucial roles in forming meaningful signs.

Globally, there are over 200 distinct sign languages, each with its own grammatical rules and syntactic structures (World Federation of the Deaf). Indian Sign Language (ISL) is one such rich and complex language that is widely used by the deaf community across India. ISL is not a direct manual representation of spoken Indian languages; rather, it possesses its own unique linguistic features and visual grammar. Although Indian Sign Language (ISL) serves as an effective communication medium for the deaf community across India, communication gaps still arise when regional languages-speakers are unfamiliar with ISL. In regions like Odisha, where Odia is the dominant spoken language, the absence of accessible translation systems between ISL and Odia limits seamless interaction. This communication barrier can hinder educational, social, and professional opportunities for hearing-impaired individuals when they engage with Odia-speaking communities.

The diversity of sign languages, along with regional variations within ISL itself, further complicates mutual understanding. Moreover, translating ISL into regional spoken languages like Odia, which is predominantly used in the Indian state of Odisha, can help bridge this gap and promote inclusivity. Developing a real-time ISL-to-Odia translation system has the potential to empower hearing-impaired individuals by enabling smoother interaction in academic, professional, and social environments.

*Corresponding author: rakesh@iiit-bh.ac.in

Proceedings of the Workshop on Sign Language Processing (WSLP), pages 10–17

Sign language recognition systems primarily aim to track and interpret dynamic hand gestures and poses. However, building an accurate recognition system introduces several challenges. Vision-based sign recognition systems are prone to environmental factors such as varying lighting conditions, complex backgrounds, skin tone variations, and occlusions, which can hinder accurate gesture detection. To minimize these challenges and ensure robust gesture tracking, advanced computer vision frameworks like MediaPipe Holistic, developed by Google, can be leveraged. MediaPipe Holistic provides highly accurate real-time tracking of hand landmarks, pose, and facial key points, which are essential for extracting reliable sign features.

While several studies have successfully applied CNNs for sign language recognition in American Sign Language (ASL) (Natarajan et al., 2022) and British Sign Language (BSL), limited research exists for Indian Sign Language (ISL) and its translation into regional languages like Odia. Given the large hearing-impaired population in India and the cultural relevance of Odia, an ISL-to-Odia translation system is both necessary and impactful.

Communication is a fundamental human right that enables participation in educational, social, and professional spheres. However, individuals from the hearing and speech-impaired community often face significant communication barriers, especially in multilingual regions like India. Most existing technological solutions for sign language recognition and translation predominantly focus on translating ISL into English or Hindi. These systems overlook the linguistic diversity of India and fail to cater to regional languages such as Odia. Considering that approximately 18.9% of persons with disabilities in India reported hearing impairments (Census of India, 2011), and with over 42.5 million Odia speakers nationwide (Census of India, 2011), there is a significant need for accessible communication technologies tailored to this linguistic group.

In Odisha, the absence of a real-time ISL-to-Odia translation system poses a significant barrier for the deaf and hard-of-hearing community. Without accessible tools, effective communication with Odia-speaking peers, educators, and service providers remains limited, leading to social exclusion, reduced educational access, and restricted professional participation.

Real-time ISL-to-Odia translation is technically challenging due to the complexity of sign language, which involves dynamic hand gestures, facial expressions, and body movements. Accurate, real-time translation requires advanced computer vision and deep learning models capable of handling spatial and temporal features. Challenges also include the lack of comprehensive ISL-Odia datasets, difficulties in direct word mapping, and the need for lightweight models suitable for real-time use. To bridge this communication gap and promote inclusivity, there is a pressing need for a robust ISL-to-Odia translation system that can accurately recognize ISL gestures and generate grammatically correct Odia text in real time, enabling seamless interaction between hearing-impaired and hearing individuals.

This work aims to address this unmet need by developing a real-time ISL-to-Odia translator that leverages MediaPipe Holistic for landmark detection, a convolutional neural network (CNN) for gesture classification, and a custom ISL dataset. The system ensures accurate and culturally aligned translation, tailored to the linguistic and contextual nuances of the Odia language.

2 Literature Review

One of the initial approaches we studied was based on the VGG19 model (Shanavas et al., 2024), a deep convolutional neural network known for its high accuracy in image classification tasks. It could process hand gestures at 30 frames per second with an impressive accuracy of 95%, while maintaining robustness against lighting and background variations. However, its single-modal nature and focus on static image processing made it less effective in highly dynamic environments where sign gestures change rapidly over time, reducing its reliability for real-time applications.

Another promising technique utilized a combination of MediaPipe for extracting key facial, hand, and pose landmarks, and Long Short-Term Memory (LSTM) networks (Rehan and Mullick, 2023) for capturing temporal dependencies across gesture sequences. This method demonstrated improved understanding of dynamic sign gestures and achieved an accuracy range of 91% to 93%. Along with giving low accuracy, the system also relied on a fixed 30-frame input sequence, which limited its flexibility and responsiveness in real-time interactive settings.

A separate approach leveraged OpenPose

(Neyra-Gutiérrez and Shiguihara-Juárez, 2020) for keypoint detection and applied neural network-based summarization techniques for recognizing Peruvian Sign Language (PSL). It achieved an accuracy of 91.56% and was found to be computationally efficient. However, the model did not incorporate 3D keypoints or facial cues, resulting in reduced expressiveness and context-awareness in gesture interpretation. In another study, a hybrid model combining 3D Convolutional Neural Networks with Support Vector Machines (SVM) was employed to extract spatial and temporal features from gesture sequences in Chinese Sign Language (CSL) (Zhao et al., 2021). This method achieved 92.6% accuracy and showed strong capability in modeling motion patterns. However, its slower recognition speed and dependence on limited CSL datasets made it less suitable for real-time deployment and regional language adaptation.

In (Himasree et al., 2024), the authors proposed a novel Sparse Gabor Descriptor (SGD)-based technique along with random forest for gesture recognition with an accuracy of 94%. Similarly, The system utilizes a Vision Transformer (ViT) trained on a comprehensive video dataset to classify various sign language elements, while integrating a sophisticated language model, PHI-1.5B, to refine translated text for grammatical correctness and structural integrity and achieved robust and contextually relevant translation of ISL gestures into textual representations in (P and Francis, 2024).

The system proposed in (Kondo et al., 2024) employs the Mediapipe pose estimation library to pinpoint the exact positions of finger joints within video frames and converts these positions into one-dimensional angular features. These features are then organized sequentially to create a two-dimensional input vector for the ViT model.

The authors proposed a progressive sign language translation model to effectively separate sign language users from the background and reduce environmental interference, thus significantly improving the generalization ability in (Zou et al., 2024).

In another work proposed in (Prabha et al., 2024), the system focuses on breaking down video input into individual image frames and building three different models: EfficientNetV2, EfficientNetV2L, and ConvNeXtLarge algorithm. The accuracy yielded by the three models EfficientNet_V2, EfficientNet_V2L and ConvNextLarge are 94.20%, 92.54% and 95.21% respectively.

In order to identify an Isolated Sign Word (ISW) in Continuous Sign Language Videos (CSLV), aka Sign-Spotting, the authors proposed a Grammar-Based Inductive Learning (GBIL) framework utilizing a Grammar-Based Dictionary (GBD) that comprises pre-defined syntactic structures of tokens for handshape, location, and movement related to every Isolated Sign Word. GBIL can improve the cross-domain performance of sign spotting by integrating a grammar logic-based inference on top of deep learning architectures in (Amperayani et al., 2024).

The authors presented R-SLR, a sign language recognition system that can recognize the signs in real-time in (Ghosh et al., 2024). R-SLR identifies the hand in a video stream and extracts the region of interest. We extract the features from the pre-processed frames and classify the signs using the pre-trained DenseNet 201 model. The model performance is tested and it achieves 96.5% accuracy.

Recent research introduced an LSTM-based model with MediaPipe Holistic for Bangla Sign Language (BdSL) recognition (Das et al., 2025), achieving 88.33% accuracy by extracting keypoints and analyzing temporal gesture sequences. While effective for translating 100 isolated signs in real-time, the system faces limitations in vocabulary coverage, sentence formation, and sensitivity to lighting conditions, restricting broader usability.

Traditional gesture recognition algorithms (Badhe and Kulkarni, 2015) using FFT (Fast Fourier Transform) and template matching also showed promising accuracy (97.5%) for ISL. However, their rigid architecture, limited flexibility, and reliance on predefined gesture templates made them less adaptable for dynamic and continuous sign language input in real-world settings.

After evaluating all these models, the approach that stood out as the most relevant for our objectives was the CNN-based model tailored for Indian Sign Language and American Sign Language. This approach achieved the highest accuracy of 99.72% (Antad et al., 2024) among the surveyed models. It used convolutional neural networks for real-time detection of hand gestures, offering a practical blend of high performance, computational efficiency, and ease of implementation. The model's proven effectiveness with ISL and its adaptability to regional translation tasks made it an ideal choice for the current stage of our project.

Based on this comprehensive analysis, we concluded that the CNN-based model best met the requirements of our system. Its high accuracy, real-time processing capability, and compatibility with Indian Sign Language made it highly suitable for building our ISL recognition and translation system aimed at converting sign gestures into Odia text, thereby enhancing communication accessibility for the Odia-speaking deaf community.

3 Proposed Solution

To build a robust and context-aware gesture recognition system, we utilize OpenCV in combination with MediaPipe Holistic to capture human body landmarks in real time using a webcam. MediaPipe Holistic provides comprehensive tracking of facial features, body posture, and hand movements, which is essential for accurately detecting and interpreting sign language gestures.

The captured key pointscomprising coordinates of various body, face, and hand landmarks are extracted frame-by-frame and stored in structured files. These files form the basis of a custom dataset specifically designed for gesture recognition tasks. We focus on 12 commonly used ISL gestures: Indian, Language, Hello, Bye, Good Morning, Good Evening, Thank You, Welcome, I, You, How Are You, and Fine. Each gesture was recorded multiple times to capture variations in style, speed, and hand positioning. The final dataset consists of 1,200 samples (100 per class), collected under diverse Indian backgrounds and lighting conditions, and processed using Media Pipe for real-time key point extraction. Figure 1 shows the mapping of these commonly used gestures to Odia.

ISL WORDS	EQUIVALENT ODIA	ISL WORDS	EQUIVALENT ODIA
Indian	ଭାରତୀୟ	Thank You	ଧନ୍ୟବାଦ
Language	ଭାଷା	Welcome	ସ୍ୱାଗତମ୍
Hello	ନମସ୍କାର	I	ମୁଁ
Bye	ଶୁଭ ବିଦାୟ	You	ତୁମେ
Good Morning	ଶୁଭ ସକାଳ	How are You	ତୁମେ କେମିତି ଅଛ
Good Evening	ଶୁଭ ସନ୍ଧ୍ୟା	Fine	ଭଲ ଅଛି

Figure 1: Common Indian Sign Language (ISL) Words and Their Equivalent Odia Translations.

To make the model robust to real-world conditions, all gestures are captured in complex Indian backgrounds, featuring variations in lighting, background objects, and clothing. This ensures the dataset reflects real-life environmental complexity and improves the model's ability to generalize during real-time deployment.

This carefully curated and diverse dataset enables the training of a reliable sign language recognition model tailored for Indian cultural and visual contexts.

The proposed system presents a comprehensive and innovative approach to Indian Sign Language (ISL) recognition and its translation into the Odia language, with a strong focus on real-time applicability and inclusivity. It leverages the synergy between MediaPipe and Convolutional Neural Network (CNN) architectures for accurate gesture recognition, followed by dictionary-based mapping for regional language translation.

The integration of Mediapipe and CNN architecture within the system follows a cohesive and structured approach. Video frames captured by the webcam are processed using Mediapipe to extract relevant landmarks and features corresponding to facial expressions, body poses, and hand gestures. These extracted features are then fed into the CNN architecture for further analysis and classification, resulting in the recognition of specific sign language gestures. The overall system flow ensures seamless interaction between different components, enabling efficient and accurate sign language interpretation in real-time scenarios. By leveraging the capabilities of Mediapipe and CNN architecture, the system architecture demonstrates a powerful and effective approach to sign language recognition. Through continuous refinement and optimization, the system aims to provide enhanced support for individuals with hearing and speech impairments, empowering them to communicate effectively and participate fully in society.

To build a robust ISL-to-text translation system, a custom dataset was collected using a webcam and the MediaPipe library, capturing a diverse set of signs, including greetings, numbers, and alphabets. MediaPipe enables real-time extraction of normalized hand key-points, ensuring consistent input that is unaffected by background or lighting conditions. These key-points are preprocessed and fed into deep learning models for gesture classification. For model selection, both 2D CNN and 3D CNN architectures were implemented and evaluated to determine the one best suited to our performance and system requirements.

We developed and compared two custom multi-layered models, one comprising 2D CNN layers and another comprising 3D CNN layers, to de-

termine the optimal model for our sign language translation system. This comparative approach allowed us to identify the most computationally efficient architecture that maintains high accuracy for real-time translation of sign language gestures, balancing performance requirements with the temporal modelling capabilities essential for capturing sequential hand movements. The detailed rationale and architectural overview of the proposed system is elaborated in Figure 2.

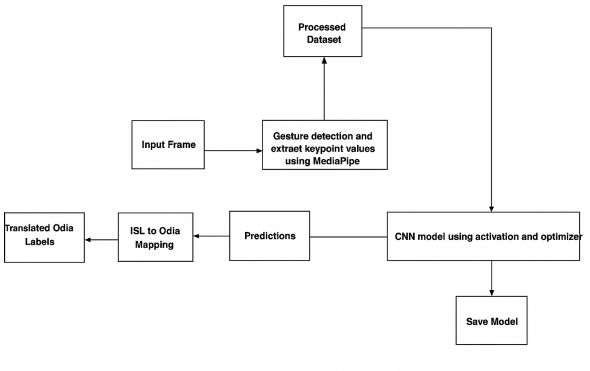

Figure 2: System Architectural Pipeline.

4 Results and Discussion

To assess the effectiveness of the proposed Indian Sign Language (ISL) to Odia text conversion system, we implemented and evaluated two distinct deep learning models: a 3D Convolutional Neural Network (3D CNN) and a 2D Convolutional Neural Network (2D CNN). Both were trained and tested on a custom ISL gesture dataset comprising 12 classes, captured in diverse Indian backgrounds to enhance real-world adaptability. The models were assessed based on training accuracy, validation accuracy, test accuracy, generalization, convergence speed, and deployment feasibility.

The 3D CNN was trained on short video sequences of 30 frames (84Œ20), allowing the model to learn spatiotemporal dynamics of gestures. The architecture consisted of stacked 3D convolutional blocks with Conv3D, Batch Normalization, MaxPooling3D, Dropout, and Dense layers. Despite its ability to capture temporal transitions, the 3D CNN demonstrated slower convergence and lower generalization:

- Training Accuracy: 88.62%
- Validation Accuracy: 77.63%
- Test Accuracy: 78.33%

Although the model improved over 75 training epochs, its validation and test accuracy (shown in Figure 3 and Figure 4) lagged, showing signs of overfitting. This outcome suggests that while 3D CNNs are suited for motion-aware tasks, the gesture variability and limited dataset size hinder their generalization. Additionally, its large parameter count (~570K) increased the risk of resource consumption.

Figure 3: Training and Validation Accuracy of 3D-CNN.

Figure 4: Training and Validation Loss of 3D-CNN.

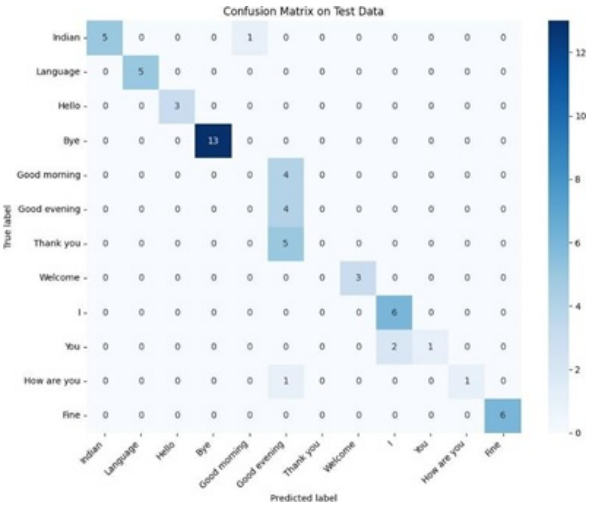

Figure 5: Confusion Matrix of 3D-CNN.

The 3D CNN confusion matrix (shown in Figure 5) reveals several misclassifications despite correct predictions in many categories. For instance, it confuses “Good evening” with “Good morning,” and one sample of “How are you” is

misclassified, indicating challenges in capturing fine-grained spatial features despite temporal modelling. Key errors included confusing "Indian" with "Bye", "Good evening" with "Good morning", "You" with "I" and "How are you" with "Good evening", along with two other isolated misclassifications. These results indicate that the model had difficulty distinguishing between gestures with subtle spatial or temporal similarities, leading to reduced accuracy in certain classes.

A more efficient 2D CNN was developed, utilizing skeletal keypoint features (x, y, z coordinates) extracted from each frame using MediaPipe Holistic. These features were flattened and treated as input for the model. The 2D CNN, built using Conv2D, Batch Normalization, MaxPooling2D, Dropout, and Dense layers, showed remarkable performance showing:

- Training Accuracy: 95.04%
- Validation Accuracy: 99.56%
- Test Accuracy: 98.33%

Figure 6: Training and Validation Accuracy of 2D-CNN.

Figure 7: Training and Validation Loss of 2D-CNN.

Figure 6 and Figure 7 show the respective graphs relating to training and validation accuracy and loss obtained. The model not only converged faster (within 50 epochs) but also generalized better on unseen data. It was less prone to overfitting, required fewer computational resources (~160K parameters), and performed well under varying lighting, orientation, and hand shape conditions commonly found in Indian settings.

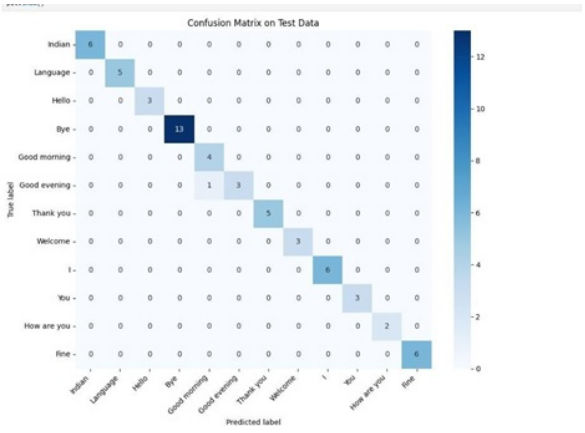

Figure 8: Confusion Matrix of 2D-CNN.

A confusion matrix (shown in Figure 8) revealed minimal misclassifications, further proving the 2D CNNs robustness. The 2D CNN also performed well in predicting gestures such as "Bye" (13/13), "Fine" (6/6), and "I" (6/6). It maintained high accuracy for "Indian" and "Language", and showed good performance in static gestures like "You" (3/3) and "How are you" (2/2). However, it struggled slightly with "Good evening", misclassifying one instance as "Good morning". Moreover, the system's strong performance can be attributed to the quality and diversity of the custom-built dataset, tailored specifically for Indian gesture styles.

The comparative analysis of the results as given in Table 1 demonstrates that the 2D CNN outperforms the 3D CNN in terms of accuracy, generalization, training efficiency, and real-time applicability. While 3D CNNs are conceptually powerful for capturing motion, their computational demands and sensitivity to data variance limit their performance on small to mid-sized datasets. The 2D CNN, however, demonstrated high reliability, minimal errors, and fast training, making it ideal for integration into the ISL to Odia text translation system. This validates our system's current implementation and supports the use of 2D CNNs for practical sign language translation applications.

Furthermore, the successful end-to-end mapping of gestures to the Odia language not only fills a significant gap in regional assistive technologies but also represents the first known implementation of direct Indian Sign Language to Odia conversion using deep learning. The system sets a strong precedent for future work in inclusive communication tools tailored for India's diverse linguistic landscape.

Aspect	3D CNN	2D CNN
Training Accuracy	88.62%	95.04%
Validation Accuracy	77.63%	99.56%
Test Accuracy	78.33%	98.33%
Model Complexity	High (~570K parameters)	Moderate (~160K parameters)
Training Duration	75 epochs (slow convergence)	50 epochs (fast convergence)
Strengths	Captures motion over time; useful for complex sequences	Lightweight, highly accurate, real-time ready
Weaknesses	Overfits easily, resource-heavy, and has lower generalization	Limited temporal modelling
Suitability for Real-time Deployment	Less suitable	Highly suitable

Table 1: Comparative Analysis of Performance of 3D CNN and 2D CNN Models for Gesture Recognition.

4.1 ISL to Odia Language Mapping

The model incorporates a feature that translates ISL gestures into Odia script. This is achieved through a direct mapping system using a predefined dictionary of related Odia words. Each recognized ISL gesture is mapped to its corresponding Odia word or phrase, enabling the system to provide text output in Odia.

To facilitate this, the system uses a Tkinter-based UI that allows users to seamlessly switch to Odia translation. The UI displays the translated Odia text, providing a smooth and intuitive way for users to interact with the system. This feature enhances the user experience and ensures effective communication for Odia-speaking users in the deaf and mute community. Figure 9 shows one such demo translation.

5 Conclusion and Future Scope

This work presents a comprehensive system for recognizing Indian Sign Language (ISL) gestures and translating them into the Odia language, aiming to empower the deaf and hard-of-hearing community in Odisha. By integrating computer vision and deep learning techniques, the system successfully identifies hand gestures corresponding to commonly used ISL signs and maps them to their respective Odia translations.

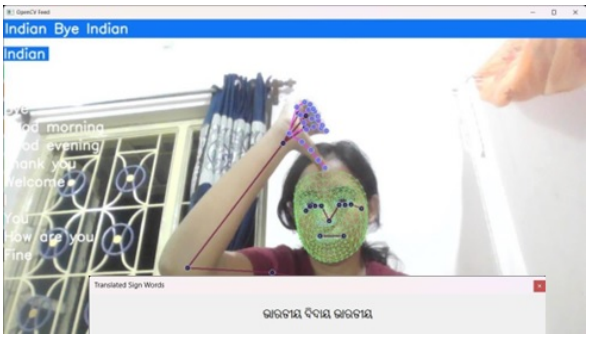

Figure 9: A Demo Translation Result.

The core of the current system utilizes MediaPipe Holistic to extract precise hand landmark coordinates, which are then processed using a Convolutional Neural Network (CNN). This combination enables efficient recognition of hand gestures captured in real time. This prototype demonstrates the potential for bridging communication gaps and improving accessibility for the hearing-impaired population, particularly those in Odia-speaking regions. Furthermore, it serves as a helpful learning tool for new individuals to learn sign language, promoting wider awareness and understanding.

In terms of model architecture, we plan to enhance temporal feature extraction using a deeper LSTM-based framework, consisting of multiple stacked LSTM layers followed by dense layers with ReLU activations for high-level feature abstraction and classification. This architecture, proven effective in prior ISL-related work, will enable our system to understand the sequence and flow of gestures more accurately, which is vital for real-time translation. We also aim to extend the system from isolated gesture recognition to sentence-level or continuous ISL translation. This will involve modelling temporal dependencies over extended gesture sequences, dynamic segmentation of signs, and restructuring of the translated output to form grammatically correct Odia sentences. This advancement will significantly improve the usability of the system in natural communication contexts.

Additionally, plans are underway to integrate the system with Odia speech synthesis, allowing

the translated Odia text to be converted into voice output. This feature will make the tool more accessible, especially for users with additional literacy or visual impairments. In the long term, the system can be embedded into real-time video chat platforms to support inclusive conversations between deaf users and Odia-speaking individuals, both in-person and online.

References

Venkata Naga Sai Apurupa Amperayani, Ayan Banerjee, and Sandeep KS Gupta. 2024. Grammar-based inductive learning (gbil) for sign-spotting in continuous sign language videos. In *2024 IEEE 7th International Conference on Industrial Cyber-Physical Systems (ICPS)*. IEEE.

Sonali M Antad, Siddhartha Chakrabarty, Sneha Bhat, Somrath Bisen, and Sneha Jain. 2024. Sign language translation across multiple languages. In *2024 International Conference on Emerging Systems and Intelligent Computing (ESIC)*, pages 741–746. IEEE.

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

Aonmoy Das, Ananna Dev Aishi, Masbah Uddin Toha, and Md Fazlul Kader. 2025. Bangla sign language translator for deaf and speech impaired people using deep lstm. *International Journal of Speech Technology*, pages 1–18.

Monalisa Ghosh, Debjani De, Lovely Anand, and Satyakam Baraha. 2024. R-slr: Real-time sign language recognition system. In *2024 3rd International Conference on Artificial Intelligence For Internet of Things (AIIoT)*, pages 1–6. IEEE.

J Himasree, PL Jeevitha, K Deekshitha, Aashrita Kolisetty, and Soumyalatha Naveen. 2024. Video-based hand gesture recognition using random forest for sign language interpretation. In *2024 Asia Pacific Conference on Innovation in Technology (APCIT)*, pages 1–6. IEEE.

Tamon Kondo, Ryouta Murai, Duk Shin, and Yousun Kang. 2024. Evaluating the accuracy of real-time japanese sign language word recognition with vision transformer models trained on angular features. In *2024 International Technical Conference on Circuits/Systems, Computers, and Communications (ITC-CSCC)*, pages 1–6. IEEE.

B Natarajan, E Rajalakshmi, R Elakkiya, Ketan Kotecha, Ajith Abraham, Lubna Abdelkareim Gabralla, and V Subramaniyaswamy. 2022. Development of an end-to-end deep learning framework for sign language recognition, translation, and video generation. *IEEE Access*, 10:104358–104374.

André Neyra-Gutiérrez and Pedro Shiguihara-Juárez. 2020. Feature extraction with video summarization of dynamic gestures for peruvian sign language recognition. In *2020 IEEE XXVII International Conference on Electronics, Electrical Engineering and Computing (INTERCON)*. IEEE.

Gadha Lekshmi P and Rohith Francis. 2024. Sign2text: Deep learning-based sign language translation system using vision transformers and phi-1.5b. In *2024 IEEE International Conference on Artificial Intelligence in Engineering and Technology (IICAIET)*, pages 282–287.

P Anantha Prabha, Mohammed Daanish, Naveen Kumar, and Nithish Kumaar. 2024. Deep-signspeak: Deep learning based sign language recognition and regional language translation. In *2024 1st International Conference on Advanced Computing and Emerging Technologies (ACET)*, pages 1–6. IEEE.

Khan Rehan and Touhid Mullick. 2023. Real time sign language translator for video conferencing platforms.

Sherin Shanavas, Naila N N, and Harikrishnan S R. 2024. Gesture recognition and sign language detection using deep learning. *International Journal of Advanced Research in Science, Communication and Technology*, 4(1):117–124.

World Federation of the Deaf. World federation of the deaf.

Kai Zhao, Kejun Zhang, Yu Zhai, Daotong Wang, and Jianbo Su. 2021. Real-time sign language recognition based on video stream. *International Journal of Systems, Control and Communications*, 12(2):158–174.

Jingchen Zou, Jianqiang Li, Xi Xu, Yuning Huang, Jing Tang, Changwei Song, Linna Zhao, Wenxiu Cheng, Chujie Zhu, and Suqin Liu. 2024. Progressive sign language video translation model for real-world complex background environments. In *2024 IEEE 48th Annual Computers, Software, and Applications Conference (COMPSAC)*, pages 519–524. IEEE.

A Appendix

- Dataset: https://drive.google.com/file/d/1fNnwoOIQP1iE68PPHQnaPgF-DgoRCdPF/view?usp=sharing
- Code: https://github.com/Astha1asn/Sign-Language-Translation-To-Odia

Low-Resource Sign Language Glossing Profits From Data Augmentation

Vania Lara-Ortiz
Tecnológico de Monterrey
School of Engineering and Sciences
Zapopan, Jalisco, Mexico
A01798098@tec.mx

Sebastian Padó
Institute for Natural Language Processing
University of Stuttgart, Germany
pado@ims.uni-stuttgart.de

Abstract

Glossing is the task of translating from a written language into a sequence of *glosses*, i.e., textual representations of signs from some sign language. While glossing is in principle 'just' a machine translation (MT) task, sign languages still lack the large parallel corpora that exist for many written language pairs and underlie the development of dedicated MT systems. In this work, we demonstrate that glossing can be significantly improved through data augmentation. We fine-tune a Spanish transformer model both on a small dedicated corpus 3,000 Spanish–Mexican Sign Language (MSL) gloss sentence pairs, and on a corpus augmented with an English–American Sign Language (ASL) gloss corpus. We obtain the best results when we oversample from the ASL corpus by a factor of 4, achieving a BLEU increase from 62 to 85 and a TER reduction from 44 to 20. This demonstrates the usefulness of combining corpora in low-resource glossing situations.

1 Introduction

Sign languages (SLs) are visual-gestural languages and the primary means of communication for Deaf communities (Schönström, 2021). Although they serve as a crucial bridge between hearing and deaf people, they remain a understudied area in natural language processing (NLP), which represents an obstacle to diversity and equity (UNESCO General Conference, 2003).

SLs do not have a standardized form in the written modality. Researchers often represent signs through *glosses*: is a notation system used to translate sign language into written form. It is written in uppercase letters and aims to represent the syntactic structure and functioning of the sign language without the interference of spoken language grammar (see Table 1). Glossing helps to describe the semantic, syntatic, and morphological characteristics of SL (Burad, 2008).

Spanish ↔ MSL	Yo quiero comer manzana (I want to eat apple)	YO MANZANA COMER QUERER (I APPLE EAT WANT)
English ↔ ASL	She is studying at the library today	TODAY SHE STUDY LIBRARY

Table 1: Examples of utterances in written language (left) and glossed sign language (right). MSL: Mexican Sign Language, ASL: American Sign Language.

Sign Language Translation (SLT) is the task of translating between sign languages and written or spoken languages. Some approaches translate directly between the two modalities (Camgoz et al., 2018; Hamidullah et al., 2024) but many approaches use glosses as an intermediate representation that breaks up the difficult task into more manageable steps (Chen et al., 2022; Mesch and Wallin, 2015). Glosses, due to their textual nature, also fit naturally into the framework of machine translation for written languages (Müller et al., 2023).

SLT represents a major challenge in the NLP community due to the scarcity of high quality data, particularly parallel corpora. According to (Sennrich and Zhang, 2019), around 1 million parallel sentences are required to effectively train a typical Neural Machine Translation (NMT) model. Existing SL corpora fall far short of this scale: The widely used RWTH-PHOENIX-Weather 2014T (Camgoz et al., 2018) corpus, based on German Sign Language (DGS) weather forecast broadcasts, contains 8,257 parallel German to DGS glosses sentences. ASLG-PC12 (Othman and Jemni, 2012) is an English-American Sign Language (ASL) gloss parallel corpus, containing approximately 87,710 aligned sentence pairs.

In this work, we consider Mexican Sign Language (MSL), the main language used throughout

Proceedings of the Workshop on Sign Language Processing (WSLP), pages 18–23

Mexico among a large segment of the deaf population (Bickford, 1991). It has its own grammatical structure and lexicon, distinct from spoken Spanish. In the NLP community, it remains highly underrepresented due to the absence of corpora. To date, only one parallel corpus is publicly available (Lara-Ortiz et al., 2025). It contains 3000 aligned sentence pairs of Spanish and glossed MSL. Our research questions (RQs) in this situation are:

RQ1 What performance do we achieve for Spanish-to-MSL glossing with a standard NMT model in this low-resource situation?

RQ2. How does this compare to a knowledge-based baseline translation model?

RQ3. Can we improve this setup with data augmentation based on other language pairs?

RQ1 simply establishes the state of affairs for fine-tuning a Transformer-based NLM on the task, using what is currently considered the standard setup for dedicated Machine Translation. Under the assumption that this does not work well, we investigate two different directions. RQ2 asks whether NMT is the most reasonable approach and compares it a simple knowledge-based approach that observes that the use of symbols as glosses (cf. Table 1) can be approximated by lemmatization. RQ3 stays with the NMT paradigm and combines the Spanish-MSL corpus with a larger English-ASL corpus. In doing so, we build on a recent body of work which shows that data augmentation can be crucial in low-resource scenarios to improve the performance of NN models in general (Li et al., 2022) and of NMT models specifically (Haddow et al., 2022).

2 Background: Sign Languages

Sign languages are human languages that arise in Deaf communities and are transmitted primarily through the visual-gestural modality. They exhibit the full range of linguistic structure found in spoken languages, including phonology (e.g., handshape, location, movement, orientation), morphology (e.g., classifier constructions), and syntax (e.g., topicalization). Crucially, sign languages are not derived from or subordinate to the spoken languages of their surrounding communities; for example, American Sign Language (ASL) is genealogically unrelated to English. Cross-linguistic variation among sign languages provides rich data for typological and theoretical inquiry.

Due to the difficulty of reducing sign languages to a written representation, corpora for SLs are much sparser than for spoken/written languages: Most major corpora contain only thousands to tens of thousands of sentences (Kopf et al., 2022).

SLs used in this study. In our study, we focus on the languages already shown in Table 1, namely Mexican Sign Language (MSL) and American Sign Language (ASL). Although ASL and MSL and ASL are distinct SLs with different lexicons and grammars, they are both derived from French Sign Language (LSF) and share structural similarities, such as SOV word order – in contrast to Spanish and English SVO word order. For example, the sentence *Yo como manzana (I eat apple)* in MSL becomes *YO MANZANA COMER (I APPLE EAT)* and its counterpart in ASL gloss stays the same. Due to these similarities, an MSL glossing model should learn generalizable patterns when the training data is augmented with English-ASL glosses.

For MSL, we use the the first and only publicly available parallel corpus for Spanish and MSL glosses (Lara-Ortiz et al., 2025). The corpus consists of 3000 aligned sentence pairs and features simplified gloss annotations. The MSL side is characterized by very short and highly compressed gloss sequences. MSL utterances cluster strongly around 1–5 tokens, with a median close to 3 tokens. This contrasts with the Spanish side, which shows a broader and slightly longer distribution (median $\approx$ 4 tokens). For ASL, we use the ASLG-PC12 corpus (Othman and Jemni, 2012) due to its size (87,710 sentence pairs). It is also widely used for gloss translation tasks in NLP (Cao et al., 2022). Specifically, we used a subset of ASL-PCG12 considering sentences with less than 7 tokens per sentence in the ASL glosses part to reduce the distributional mismatch with our SPA–MSL data. This filtering serves as a normalization step that ensures that the augmented training data are more consistent with the linguistic properties of the MSL sequences present in our primary dataset. Under this condition, 16,900 pairs of English-ASL are left over. A manual inspection of the added ASL samples shows that they come from institutional proceedings (e.g., *"opening of the sitting"*, *"documents received"*, *"there were two further issues raised"*).

3 Methodology

3.1 Base Model for Spanish-Mexican Sign Language Glossing (RQ1)

Our base model for translation from Spanish to MSL glosses (**Base Model** below) is based on BARTO (Bidirectional and Auto-Regressive Transformer for Paraphrasing in Spanish) (Araujo et al., 2024), a Transformer pre-trained on large-scale Spanish dataset. BARTO uses the BART architecture (Lewis et al., 2020), an Encoder-Decoder sequence-to-sequence model trained for paraphrasing. In the absence of NMT models for sign languages, paraphrasing models like BART(O) represent a reasonable starting point for glossing, since they are trained to reformulate sentences while preserving their core meaning. BARTO in particular is well suited for our task, given the lexical similarity between Spanish and MSL glosses, since it is pre-trained on Spanish corpora.

However, Spanish and MSL differ significantly in syntax: while Spanish typically follows a Subject-Verb-Object (SVO) structure, MSL often adopts Subject-Object-Verb (SOV) patterns. Morphologically, MSL glosses do not encode verb conjugations, gender, or number. For these reasons, we fine-tune BARTO using a parallel corpus of 3,000 Spanish–MSL sentence pairs. During fine-tuning, BARTO learns to suppress inflectional morphology and produce outputs that conform to MSL syntax.

3.2 Knowledge-Based Baseline (RQ2)

We also consider a baseline (**Baseline (Lem)** below). It builds on the observation (cf. Table 1) that the symbols used for glossing are generally lemmas of the written language used in the same language community as the sign language. This suggests that simple lemmatization should represent an informed baseline for 'translating' the written language into glosses that can account for the change in lexical material but not the reordering that also takes place.

Concretely, we employ the Spanish lemmatizer provided by decision tree-based TreeTagger (Schmid, 1995) package which computes both part-of-speech tags and lemmas. We employ a small number of postprocessing steps to make the lemma sequence more like sign language glosses: (a), we remove all articles, auxiliaries, reflexive pronouns, and prepositions (which are generally omitted in the gloss sequences); (b) we replace feminine nouns with ending *a* by the masculine noun followed by *mujer (woman)*, again following the MSL conventions, such as *abuela (grandmother)* → *abuelo (grandfather)* + *mujer (woman)*; (c) plurals like *niños (boys)* → *niño ellos (they)*, following a similar convention for plurals; (d) we restore all adjectives, which the lemmatizer changes to masculine, to their original forms. As stated above, we do not adjust word order, and the output still has predominantly the Spanish default SVO structure. The lemmatizer can be improved with rule-based reordering, but this would re- quire a full handcrafted grammar for MSL (e.g., systematic SOV reordering) beyond the scope of this study

3.3 Data Augmentation (RQ3)

Finally, we experiment with a family of models that take the Base Model (fine-tuned on 3k sentence pairs in Spanish – MSL glosses) and incrementally incorporate parallel samples from the English-ASL dataset. As stated above, this experiment tests whether knowledge from a different SL can benefit the translation of another (cross-lingual transfer) despite syntactic differences.

Specifically, we add 3000, 6000 and 9000 ASL gloss sentence pairs – i.e., the same amount as for the original language pair and 2 and 3 times as much, respectively. This leads us to data-augmented models we designate as **DA Model (3+3k), (3+6k), (3+9k)**. The MSL and ASL datasets were simply concatenated.

4 Experimental Setup

To ensure robust evaluation, we partitioned each dataset into 10 equally sized subsets and carried out a variant of 10-fold cross validation, varying the combination of training (80%), validation (10%), and test (10%) subsets across five runs. For example, in the base model using 3000 Spanish-MSL gloss pairs, 2400 samples were used for training and 300 each for validation and testing in each run.

Before training, all data were preprocessed: text was lowercased, extra spaces were removed, punctuation was preserved, and input sequences were tokenized using Sentence Piece tokenization, provided by BARTO. For all experiments, we used the Hugging Face with the following training configuration: a learning rate of 10^{-4}, weight decay of 0.01, and a total of 30 training epochs. The batch size was set to 32 for training and 64 for evaluation. An evaluation was performed at the end of each epoch. We enabled half-precision training to reduce memory usage and speed up computation.

Generation was performed during the evaluation.

We evaluate our models with the standard MT metrics BLEU-1 through BLEU-4 (higher is better) and TER (lower is better). This enables us to measure both glossing quality at the level of individual tokens as well as overall quality.

Our experimental results are shown in Table 2, and example translations in Table 3. We now reconsider the research questions from Section 1.

5 Results

RQ1: Performance of the Base Model. The base model achieves a BLEU-1 score of 0.62, indicating a reasonable unigram performance. The BLEU-n scores however decline substantially for higher n (e.g., BLEU-4=0.35), indicating that the model struggles with longer phrases. This is also shown by the pretty high TER score of 44.2. The examples in Table 3 confirm that the parallel corpus that is available for Spanish–MSL glosses is not sufficient for the model to acquire the syntactic patterns of MSL, neither regarding function words nor word order – the output still looks largely Spanish.

RQ2: Performance of the Baseline. The lemmatizer-based baseline achieves a BLEU-1 score of 0.79, surpassing the base model considerably in unigram precision. This demonstrates again the proximity of glosses in MSL to Spanish lemmas – and in fact, the Baseline outperforms the Base model also substantially on the TER metric. However, the Baseline is basically unable to produce correct longer n-grams, which is expected, since it does not even attempt to capture the word order differences between Spanish and MSL glosses. Table 3 confirms that the Baseline does a fair job for very short sentences (such as pair 2) but not otherwise.

RQ3: Performance of the Data-Augmented Model. The DA model now also outperforms the Lemmatizer baseline on all metrics. However, we observe a clear behavior of diminishing returns: increasing the training corpus to 3+6=9k yields a smaller improvement, and a final increase to 3+9=12k sees essentially unchanged performance. The TER results mirror the behavior we find for BLEU, as do the example translations in Table 3: there is a clear improvement from the base model to the DA mode in terms of syntactic pattern, but then little further adaptation. This is expected, since mixing in more ASL data ultimately causes the model to optimize more towards ASL glossing. Indeed, we consider it a positive result that the AD model's performance on MSL glossing remains stable: Other studies on multi-lingual MT using a comparable setup found that results on a language pair can suffer when too much data for another language pair is added (Johnson et al., 2017).

6 Conclusions

In this paper, we have considered the translation from a written language into sign language glosses, a task that is both important from an equity point of view and difficult to capture with our current standard neural models due to the lack of large corpora. Indeed, our base model does worse than a lemmatizer baseline according to some metrics. Augmenting the training data with a gloss corpus for another (closely related) sign language yields a fair increase in glossing quality, but with diminishing returns for the addition of more data. These findings are largely in agreement with findings of data augmentation methods across a range of tasks (Li et al., 2022) but still do not yield a satisfactory answer to the question of how glossing can be further improved in such low-resource scenarios. Avenues for future research include the creation of synthetic data (see Perea-Trigo et al. (2024) for a rule-based approach) with the challenge of achieving a natural distribution, or alternatively the combination of a lemmatization-based approach—which is very good at generating the correct lexical material—with a reordering strategy to match the sign language's syntactic patterns, e.g., inspired by traditional statistical MT (Durrani et al., 2011). Exploring augmentation with an unrelated language pair, such as German–DGS, also represents a promising direction. Moreover, evaluating additional sequence-to-sequence architectures such as mT5, mBART, and other multilingual pretrained models remains an open line of research.

Acknowledgments. We express our sincere gratitude to the *Grupo Promotor de la LSM* for their guidance and support during the development of this project. We also acknowledge the financial support provided by the German Academic Exchange Service (DAAD) through the program *Research Grants – One-Year Grants for Doctoral Candidates, 2024/25*, funding number 57693452.

7 Limitations

In our study, we considered only two sign languages (with a focus on one of them, namely Mex-

Metric	Base Model (3k)	Baseline (Lem)	DA Model (3+3k)	DA Model (3+6k)	DA Model (3+9k)
BLEU-1	0.62 ± 0.072	0.79 ± 0.068	0.78 ± 0.076	0.84 ± 0.044	0.84 ± 0.045
BLEU-2	0.53 ± 0.106	0.39 ± 0.103	0.69 ± 0.104	0.76 ± 0.063	0.76 ± 0.061
BLEU-3	0.44 ± 0.132	0.17 ± 0.098	0.60 ± 0.105	0.67 ± 0.084	0.67 ± 0.080
BLEU-4	0.35 ± 0.1432	0.08 ± 0.140	0.48 ± 0.097	0.55 ± 0.098	0.55 ± 0.097
TER	44.2 ± 9.56	34.5 ± 10.19	28.7 ± 12.19	21.0 ± 6.76	21.0 ± 6.37

Table 2: Results for Mexican Sign language glossing with the Base, Baseline and Data-Augmented Models

Original (Spanish)	Ellas viven en México (They live in Mexico)	La niña está loca (The girl is crazy)	Tu amiga es distraída (Your friend (female) is distracted)
Original (MSL gloss)	MÉXICO ELLAS VIVIR (Mexico they live)	NIÑO MUJER LOCA (Boy woman crazy)	AMIGO MUJER TUYA DISTRAÍDA ASÍ (Friend woman yours distracted [PARTICLE])
Base Model	Ellas viven en México	La niña be loca	Tu amiga es distraída
Baseline (Lem)	Ellas vivir México	Niño mujer loca	Tuya amigo mujer distraída
DA Model (3+3k)	México ellas vivir	Niño mujer loca	Amiga tuya distraída así
DA Model (3+6k)	México ellas vivir	Niña loca	Amigo mujer tuya distraída
DA Model (3+9k)	México ellas vivir	Niño mujer loca	Amigo mujer tuya distraída así

Table 3: Three example sentence pairs with translations by the different models

ican Sign Language), and only a single neural language model. It remains to be tested to what extent these results generalize to other sign languages and to other NLMs.

8 Ethical Considerations

This project was carried out with the awareness and support of the *Grupo Promotor de la LSM*, a group of Mexican Deaf people and MSL interpreters, whose participation ensured alignment with community perspectives. However, the group cannot represent the full diversity of Mexican Sign Language (MSL), and the dataset may not capture all regional or sociolinguistic variations. Moreover, glossing inherently simplifies the grammatical richness of MSL. Finally, it is important to note that this dataset and any translation systems built from it should complement, but never replace, the work of professional interpreters, since misuse could negatively impact accessibility and the rights of the Deaf community.

References

Vladimir Araujo, Maria Mihaela Trusca, Rodrigo Tufiño, and Marie-Francine Moens. 2024. Sequence-to-sequence Spanish pre-trained language models. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 14729–14743, Torino, Italia. ELRA and ICCL.

J. Albert Bickford. 1991. Lexical variation in Mexican Sign Language. *Sign Language Studies*, 72:241–276.

Viviana Burad. 2008. La glosa: Un sistema de notación para la lengua de señas. *Cultura sorda*.

Necati Cihan Camgoz, Simon Hadfield, Oscar Koller, Hermann Ney, and Richard Bowden. 2018. Neural sign language translation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7784–7793, Salt Lake City, USA. IEEE.

Yong Cao, Wei Li, Xianzhi Li, Min Chen, Guangyong Chen, Long Hu, Zhengdao Li, and Kai Hwang. 2022. Explore more guidance: A task-aware instruction network for sign language translation enhanced with data augmentation. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 2679–2690, Seattle, United States. Association for Computational Linguistics.

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 (CVPR)*, pages 5120–5130.

Nadir Durrani, Helmut Schmid, and Alexander Fraser. 2011. A joint sequence translation model with integrated reordering. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 1045–1054, Portland, Oregon, USA. Association for Computational Linguistics.

Barry Haddow, Rachel Bawden, Antonio Valerio Miceli Barone, Jindřich Helcl, and Alexandra Birch. 2022. Survey of low-resource machine translation. *Computational Linguistics*, 48(3):673–732.

Yasser Hamidullah, Josef van Genabith, and Cristina España-Bonet. 2024. Sign language translation with sentence embedding supervision. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 425–434, Bangkok, Thailand. Association for Computational Linguistics.

Melvin Johnson, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2017. Google's multilingual neural machine translation system: Enabling zero-shot translation. *Transactions of the Association for Computational Linguistics*, 5:339–351.

Maria Kopf, Marc Schulder, and Thomas Hanke. 2022. The Sign Language Dataset Compendium: Creating an overview of digital linguistic resources. In *Proceedings of the LREC2022 10th Workshop on the Representation and Processing of Sign Languages: Multilingual Sign Language Resources*, pages 102–109, Marseille, France. European Language Resources Association (ELRA).

Vania Lara-Ortiz, Rita Fuentes-Aguilar, and Isaac Chairez. 2025. Spanish to Mexican Sign Language glosses corpus for Natural Language Processing tasks. *Scientific Data*, 12.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Bohan Li, Yutai Hou, and Wanxiang Che. 2022. Data augmentation approaches in natural language processing: A survey. *AI Open*, 3:71–90.

Johanna Mesch and Lars Wallin. 2015. Gloss annotations in the Swedish sign language corpus. *International Journal of Corpus Linguistics*, 20.

Mathias Müller, Zifan Jiang, Amit Moryossef, Annette Rios, and Sarah Ebling. 2023. Considerations for meaningful sign language machine translation based on glosses. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 682–693, Toronto, Canada. Association for Computational Linguistics.

Achraf Othman and Mohamed Jemni. 2012. English-ASL gloss parallel corpus 2012: ASLG-PC12. In *5th Workshop on the Representation and Processing of Sign Languages: Interactions between Corpus and Lexicon at LREC*.

Marina Perea-Trigo, Celia Botella-López, Miguel Ángel Martínez-del Amor, Juan Antonio Álvarez García, Luis Miguel Soria-Morillo, and Juan José Vegas-Olmos. 2024. Synthetic corpus generation for deep learning-based translation of Spanish Sign Language. *Sensors*, 24(5).

Helmut Schmid. 1995. Improvements in part-of-speech tagging with an application to german. In *Proceedings of the ACL SIGDAT-Workshop*, Dublin, Ireland.

Krister Schönström. 2021. Sign languages and second language acquisition research: An introduction. *Journal of the European Second Language Association*, 5:30–43.

Rico Sennrich and Biao Zhang. 2019. Revisiting low-resource neural machine translation: A case study. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 211–221, Florence, Italy. Association for Computational Linguistics.

UNESCO General Conference. 2003. Recommendation concerning the promotion and use of multilingualism and universal access to cyberspace. Technical report, UNESCO.

Augmenting Sign Language Translation Datasets with Large Language Models

Pedro Dal Bianco
III-LIDI
Universidad Nacional de La Plata
pdalbianco@lidi.info.unlp.edu.ar

Jean Paul Nunes Reinhold
CDTEC, Federal University of Pelotas
jean.pnr@inf.ufpel.edu.br

Facundo Quiroga
III-LIDI
Comisión de Investigaciones Científicas
Universidad Nacional de La Plata
fquiroga@lidi.info.unlp.edu.ar

Franco Ronchetti
III-LIDI
Comisión de Investigaciones Científicas
Universidad Nacional de La Plata
fronchetti@lidi.info.unlp.edu.ar

Abstract

Sign language translation (SLT) is a challenging task due to the scarcity of labeled data and the heavy-tailed distribution of sign language vocabularies. In this paper, we explore a novel data augmentation approach for SLT: using a large language model (LLM) to generate paraphrases of the target language sentences in the training data. We experiment with a Transformer-based SLT model (Signformer) on three datasets spanning German, Greek, and Argentinian Sign Languages. For models trained with augmentation, we adopt a two-stage regime: pre-train on the LLM-augmented corpus and then fine-tune on the original, non-augmented training set. Our augmented training sets, expanded with GPT-4-generated paraphrases, yield mixed results. On a medium-scale German SL corpus (PHOENIX14T), LLM augmentation improves BLEU-4 from 9.56 to 10.33. In contrast, a small-vocabulary Greek SL dataset with a near-perfect baseline (94.38 BLEU) sees a slight drop to 92.22 BLEU, and a complex Argentinian SL corpus with a long-tail vocabulary distribution remains around 1.2 BLEU despite augmentation. We analyze these outcomes in relation to each dataset's complexity and token frequency distribution, finding that LLM-based augmentation is more beneficial when the dataset contains a richer vocabulary and many infrequent tokens. To our knowledge, this work is the first to apply LLM paraphrasing to SLT, and we discuss these results with respect to prior data augmentation efforts in sign language translation.

1 Introduction

Sign Language Translation (SLT) aims to convert sign language video into spoken language text, bridging communication between deaf signers and hearing people. It is a multimodal task at the intersection of computer vision and natural language processing, and has seen steady progress in recent years (Camgoz et al., 2018, 2020). However, SLT remains extremely challenging due to the scarcity of large-scale parallel sign-video-to-text datasets (Bragg et al., 2019). Datasets that do exist tend to be limited in domain and have a heavy-tailed vocabulary distribution, with many words appearing only a few times (or even once) in the corpus. For example, the popular RWTH-PHOENIX-Weather 2014T (Phoenix14T) German SL dataset (Camgoz et al., 2018) has a relatively small vocabulary (under 3k words) and a high mean word frequency, making it easier for models to achieve relatively good BLEU scores compared to broader-domain corpora. In contrast, newer, more diverse SLT datasets feature much larger vocabularies and a majority of low-frequency tokens, resulting in very low baseline translation performance. The combination of sparsity and long-tail token distribution poses a major hurdle for training effective SLT models. A quantitative summary of these differences including vocabulary size and the proportion of singletons that drive long-tail effects is provided in Table 2.

Data augmentation is a common strategy to address low-resource settings. In spoken language machine translation, methods like back-translation

Proceedings of the Workshop on Sign Language Processing (WSLP), pages 24–30

and paraphrasing are commonly used to boost performance in low-resource scenarios (Sennrich et al., 2016; Hu et al., 2021). In the context of sign language, prior work has explored various augmentation techniques. Moryossef et al. (2021) generate synthetic gloss–text pairs from monolingual spoken text and report *relative* gains of +19.7% BLEU on NCSLGR (Neidle and Vogler, 2012) and +10.4% on PHOENIX14T (Camgoz et al., 2018). More recently, (Walsh et al., 2025) leveraged Sign Language Production models to generate new sign video samples (either via skeletal pose manipulation or video GANs), yielding up to 19% relative improvement in BLEU score. These approaches augment data on the sign language input (either at the gloss or video level). By contrast, our focus is on augmenting the *text output* of the training pairs using modern LLMs.

Large language models have demonstrated remarkable capabilities in producing paraphrases and diversifying text while preserving meaning. We investigate whether an LLM (GPT-4 in our case) can be used to automatically create multiple paraphrased translations for each sign video, thereby enlarging the effective training set and exposing the translation model to a richer variety of expressions. Our hypothesis is that this can alleviate the impact of rare words and rigid sentence patterns in SLT training data. To our knowledge, this idea has not been explored in prior SLT research, although LLMs have been integrated into SLT pipelines in other ways (e.g., using pretrained text models for the translation decoder (Wong et al., 2024)).

We conduct experiments on three datasets covering different sign languages and levels of complexity: (1) Phoenix14T (German Sign Language) (Camgoz et al., 2018), a weather forecast domain corpus; (2) a Greek Sign Language (GSL) corpus of educational video translations (Voskou et al., 2023); and (3) an Argentinian Sign Language (LSA) corpus derived from the LSA-T dataset (Bianco et al., 2022). We augment each training set by generating three paraphrases per original sentence using GPT-4 (with prompts instructing the model to preserve semantics and most words while varying word order). For augmented models, we first train on the augmented corpus and then fine-tune on the original sentences only. We employ a Transformer translation model based on the Signformer architecture (Yang, 2024). We compare our augmented models against baselines trained solely on the original data.

Our main contributions can be summarized as follows: (1) We introduce LLM-based target-output paraphrasing as a data augmentation technique for sign language translation and release four augmented versions of SLT datasets (covering DGS, GSL, LSA and ISL). (2) We present an empirical evaluation of this augmentation across datasets of varying vocabulary size and complexity, showing that its impact differs markedly: from a modest BLEU-4 improvement in one case to negligible or even slight negative effects in others. We analyze these outcomes and provide hypotheses linking them to dataset properties such as vocabulary breadth and frequency of singletons. All of our code and datasets are publicly available [1].

2 Related Work

Sign Language Translation. Early SLT systems followed a two-stage approach: first performing continuous sign language recognition to predict an intermediate gloss sequence, then translating glosses to text (Camgoz et al., 2018). Glosses are textual labels (often one per sign) that approximate the signed content. While glosses simplify the translation problem, creating gloss annotations is labor-intensive and glosses cannot capture all nuances (facial expressions, classifier constructions, etc.). To avoid these limitations, recent research has shifted toward *gloss-free* SLT, building end-to-end models that map video directly to spoken language text (Camgoz et al., 2020; Chen et al., 2022). Gloss-free SLT is considerably more challenging, typically yielding lower accuracy than gloss-based methods, but it is more scalable since it requires only video-text pairs. Modern gloss-free approaches often employ transformer architectures and have begun incorporating large pretrained models. For example, the Sign2GPT system (Wong et al., 2024) uses a pretrained CLIP visual encoder and a GPT-style language model for decoding, with lightweight adapters, achieving state-of-the-art results on Phoenix14T and CSL-Daily (Chinese Sign Language). (Yang, 2024) introduced *Signformer*, a transformer that eschews any pretrained components and is extremely lightweight (0.57M parameters for a smaller variant), yet it reached competitive performance (second place on Phoenix14T gloss-free leaderboard). Our work builds on a Signformer-like architecture, but using a sequence of body pose keypoints as input.

[1] Url anonimized for review purposes.

Data Augmentation in SLT. The scarcity of sign-to-text data has motivated various augmentation strategies. Aside from simple video augmentations (e.g., mirroring, spatial jitter) commonly used in sign recognition, researchers have proposed more complex methods for SLT. On the sign input level, one approach is to generate synthetic training examples using sign language production models. (Stoll et al., 2020) and others have developed techniques to create sign animations or videos from text; however, the visual quality and realism of generated signs can be limiting. Recent work by (Walsh et al., 2025) took a step forward by employing (i) skeleton-based motion synthesis and stitching, and (ii) generative adversarial models (SignGAN, SignSplat) to produce artificial sign video variations, yielding *relative* improvements in BLEU of up to ∼19% on benchmark SLT datasets. Complementarily, in *sign language recognition* (SLR), dynamic sign generation has also proven effective: works like Rios et al. (2025) introduce *HandCraft*, a lightweight generator that produces synthetic sign sequences and, through synthetic-data pretraining, establishes new state-of-the-art results on LSFB and DiSPLaY—further supporting the value of sign-level augmentation for recognition. On the text output, data augmentation is less explored in SLT. (Moryossef et al., 2021) augmented the text output of a gloss-to-text translator by creating paraphrase pairs from monolingual data with heuristic rules, effectively expanding data via pseudo-gloss generation. In broader NLP, LLMs like GPT-3/4 have been used to generate paraphrases or new training samples for low-resource tasks (Davoodi et al., 2022). In this work, we apply a similar idea specifically to SLT: using an LLM to rephrase ground-truth translations in order to introduce lexical and syntactic variety. This approach does not require any additional sign data and thus is complementary to sign-level augmentation methods. We compare our results with prior augmentation approaches and discuss scenarios where text augmentation might be preferable or vice versa.

3 Methodology

3.1 Model Architecture

Our baseline model is inspired by Signformer (Yang, 2024), a recent transformer-based SLT model designed for efficiency. We adopt a simplified version of Signformer in which, instead of feeding in spatio-temporal visual embeddings (e.g., CNN features from video frames), we use pose keypoints extracted from each video frame. Specifically, we utilize the MediaPipe Holistic (Maia et al., 2025) model to obtain 2D coordinates of the signer's body, hands, and face key landmarks for each frame. These pose landmarks (in total, we use 33 body pose points, 21 points for each hand, and a subset of facial landmarks relevant to mouth and eyebrows) are concatenated into a feature vector per frame, yielding a time-series of pose features. We then project this pose feature vector into the model's embedding space via a linear layer. This serves as the input to the encoder. By using skeleton data, we drastically reduce the input dimensionality and remove background noise, potentially enabling faster training and inference suitable for edge devices (Yang, 2024). However, this comes at the cost of losing some visual information (like detailed appearance, color, or subtle gestures not captured by keypoints). Prior findings suggest pose-based approaches may slightly lag behind image-based models in translation quality, especially on unconstrained content (Zelezný et al., 2025). We acknowledge this trade-off; indeed, our model's absolute BLEU scores are lower than state-of-the-art results that use full video frames (see Section 5). Nonetheless, the *relative* comparisons (with vs. without augmentation) remain meaningful within our setup.

3.2 LLM-Based Data Augmentation

To augment the training data, we employ GPT-4 as a paraphrase generator. For each video-sentence pair (V, T) in the original training set (where T is the ground-truth spoken language translation of the sign video V), we generate $N = 3$ additional sentences T'_1, T'_2, T'_3 that convey the same meaning as T. We design a prompt to guide GPT-4 to produce high-quality paraphrases that preserve semantics and key vocabulary. The prompt (shown in figure 1) attempts to generate paraphrases that are close to the original sentence in vocabulary and style, while introducing some variation (particularly in word order and occasional synonyms). The constraint to reuse 70% of words is intended to prevent GPT-4 from rephrasing too freely and possibly introducing unfamiliar vocabulary that might confuse the translation model. We adjusted the prompt for each target language (e.g., Spanish for LSA, Greek for GSL, etc.) accordingly.

For each original sign video V, we thus obtain 3

translations: the original T and three paraphrases T'_1, T'_2, T'_3. During training we materialize these as 4 separate examples (V, T), (V, T'_1), (V, T'_2), (V, T'_3) (i.e., V is repeated four times with each textual variant). Figure 1 summarizes the overall augmentation pipeline. As concrete illustrations of the augmentation, Table 1 shows three training instances from RWTH-Phoenix datasets and their LLM generated paraphrases.

3.3 Training Schedule

We compare two conditions:

- **Baseline:** train the model on the original (non-augmented) training set only.
- **+Augmentation:**
  - *Stage1:* pre-train on the augmented corpus (original targets + three GPT-4 paraphrases per instance).
  - *Stage2:* fine-tune on the original training set only, to realign the decoder distribution with the reference phrasing and reduce drift toward rare paraphrastic variants. Unless otherwise stated, all hyperparameters are kept identical across conditions; early stopping is performed on the same development set.

4 Datasets and Evaluation

We evaluate our approach on three sign language translation datasets that differ notably in linguistic diversity, recording conditions, and lexical structure—factors that strongly influence how data augmentation behaves.

The **PHOENIX14T** dataset (Camgoz et al., 2018) contains weather broadcast recordings in German Sign Language (DGS) with corresponding German text. It is a real-world corpus characterized by consistent domain-specific phrasing and limited topic variation. Although this repetitiveness simplifies translation, the naturally recorded conditions introduce visual variability across signers and sessions, maintaining a moderate level of linguistic and visual complexity.

In contrast, the **GSL** dataset (Adaloglou et al., 2020) is recorded under controlled laboratory conditions, featuring a small group of signers repeatedly performing a restricted set of predefined sentences. As a result, it exhibits low linguistic and visual variability, with high redundancy across samples and virtually no rare tokens. This simplicity allows models to easily memorize sentence structures and reach near-perfect BLEU scores, but at the cost of generalization.

Finally, the **LSA-T** dataset (Bianco et al., 2022) comprises real-world videos from diverse sources, with a wide range of signers, lighting, and signing styles. Its naturalistic, spontaneous signing and extensive Spanish vocabulary make it a far more challenging dataset. The high proportion of singletons and irregular phrasing create a long-tail distribution, resulting in sparse lexical coverage and low baseline translation accuracy. This makes LSA-T particularly valuable for testing augmentation strategies aimed at mitigating data scarcity and improving robustness under realistic conditions.

Together, these datasets span a spectrum from controlled and repetitive to unconstrained and diverse, providing an ideal testbed for assessing how LLM-based paraphrasing interacts with varying levels of linguistic and visual complexity. Table 2 quantitavely describes mentioned datasets.

For all datasets, we preprocessed the videos with MediaPipe to extract pose sequences, as described above. We then normalized coordinate values and frame rates for input to the model (following steps similar to (Železný et al., 2023)). The text was lowercased and tokenized; we built a separate vocabulary for each language (German, Greek, Spanish) with a size of 5,000 tokens, ensuring coverage of all training words. We evaluate translation quality using case-insensitive BLEU-4 (Papineni et al., 2002) on the test set.

5 Results and Analysis

Table 3 reports BLEU-4 on the test sets for the Baseline vs. the two-stage **+Augmentation** setup.

Overall trends. Phoenix14T shows a small but consistent gain (+0.77 BLEU). Given its moderately rich yet formulaic domain, exposing the decoder to paraphrastic re-orderings appears to improve generalization beyond memorized templates, and the subsequent fine-tuning on original references helps keep the output close to the evaluation style. In contrast, the GSL subset starts with an exceptionally high baseline (94.38 BLEU), indicating substantial overlap and low linguistic variability between training and test. In this near-saturated regime, even with our final fine-tuning stage, augmentation slightly hurts (92.22 BLEU): the decoder learns alternative, semantically valid phrasings that do not exactly match the single reference, and the

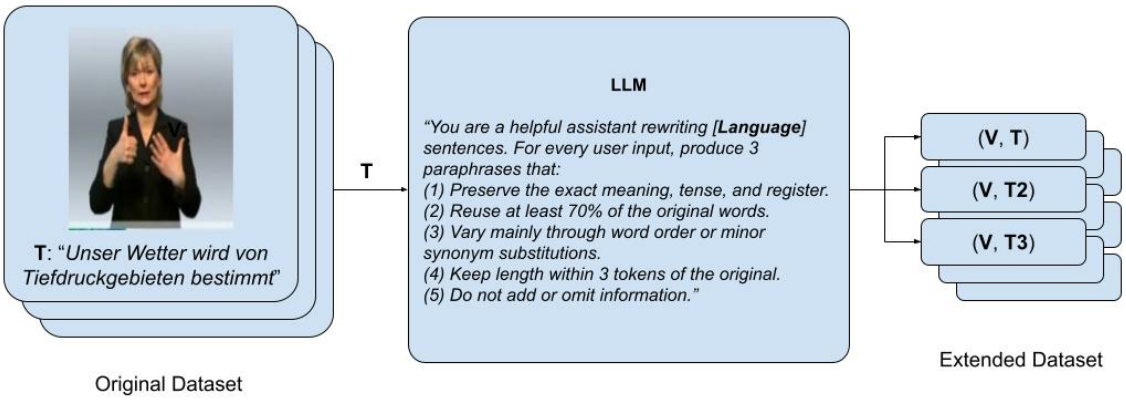

Figure 1: LLM-augmented SLT pipeline. For each video–text pair (V, T) we use an LLM to generate three paraphrases (T'_1, T'_2, T'_3) that preserve meaning while introducing limited lexical/syntactic variety. We then adopt a two-stage regime: (i) pre-train on the augmented corpus (all T and T'_i), (ii) fine-tune on the original targets T only. At test time, the model translates directly from the sign input to text.

Original (reference)	**LLM paraphrases**
tiefdruckgebiete bestimmen unser wetter *low-pressure areas determine our weather*	• Unser Wetter wird von Tiefdruckgebieten bestimmt. *Our weather is determined by low-pressure areas.* • Die Bestimmung unseres Wetters erfolgt durch Tiefdruckgebiete. *The determination of our weather is due to low-pressure areas.*
auch mit den temperaturen geht es aufwärts *the temperatures are also rising*	• Auch die Temperaturen steigen an. *The temperatures are also increasing.* • Die Temperaturen gehen ebenfalls nach oben. *The temperatures are also going up.*
eine gewitterfront überquert deutschland von west nach ost *a thunderstorm front crosses Germany from west to east*	• Eine Gewitterfront zieht von Westen nach Osten über Deutschland. *A thunderstorm front moves from west to east across Germany.* • Von Westen nach Osten überquert eine Gewitterfront Deutschland. *From west to east, a thunderstorm front crosses Germany.*

Table 1: Original training references paired with their GPT-4 paraphrases from the PHOENIX14T dataset.

fine-tune does not fully eliminate these variants. Finally, the reduced LSA-T subset remains extremely low (around 1.2 BLEU) in both settings; paraphrasing largely preserves the same rare content words (by design of our prompt) and thus does not mitigate the core issue: severe data sparsity on the sign inputs and a very heavy-tailed token distribution.

Data characteristics matter. The observed utility of LLM paraphrasing correlates with vocabulary breadth and the prevalence of infrequent tokens. When the dataset offers enough lexical variety (Phoenix14T), paraphrastic exposure helps the model handle word-order and light lexical alternations encountered at test time. When the task is artificially simple (our GSL subset), increased output variety degrades single-reference BLEU despite the final fine-tune. When the vocabulary is extremely sparse (our reduced LSA-T subset), paraphrasing the target alone does not address full coverage: many content signs/words are never learned well enough for the decoder to benefit from the text augmentation.

On pose inputs. Our absolute Phoenix14T scores (around 10 BLEU) are well below SOTA that use full video features and/or gloss supervision (22–24 BLEU). Likely contributors include our small model size, reliance on 2D pose keypoints (which may miss mouthing and subtle facial cues), and the absence of an intermediate gloss stage (Yang, 2024; Maia et al., 2025). Nevertheless, within this consistent pose-based setup, the two-stage augmentation policy yields the relative effects summarized above.

6 Conclusion

We presented a study on augmenting SLT training data by generating paraphrase variations of the target text using an LLM, combined with a two-stage training schedule that pre-trains on augmented text and then fine-tunes on the original data. Across multiple sign languages, this strategy yields a modest improvement on a medium-complexity dataset

Statistic	PHOENIX14T (DGS)	GSL	LSA-T (LSA)
Language (target)	German	Greek	Spanish
Sign language	DGS	GSL	LSA
Real-world footage	Yes	No	Yes
No. of signers	9	7	103
Duration [h]	10.71	9.51	21.78
Samples (clips)	7,096	10,295	8,459
Unique sentences	5,672	331	8,102
% unique sentences	79.93%	3.21%	95.79%
Vocabulary size (types)	2,887	N/A	14,239
Singletons (types with count=1)	1,077	0	7,150
% singletons	37.3%	0%	50.21%
Resolution	210×260	848×480	1920×1080
FPS	25	30	30

Table 2: Corpus statistics for the three datasets used in our experiments. The bottom block highlights lexical properties related to long-tail behavior (vocabulary size and proportion of singletons).

Dataset	Baseline (BLEU-4)	+Augmentation (BLEU-4)
PHOENIX14T (DGS)	9.56	10.33
GSL (Greek)	94.38	92.22
LSA (Spanish)	1.18	1.19

Table 3: Test BLEU-4 for baseline vs. LLM-augmented training on three datasets.

(Phoenix14T), but negligible or negative effects on extremely simple (GSL subset) or extremely sparse (reduced LSA-T subset) settings. These results suggest that LLM-based target output augmentation is not a one-size-fits-all solution; its usefulness depends on properties like vocabulary diversity and data sufficiency.

In addition, we demonstrated a pose-based SLT modeling approach that, while not achieving SOTA accuracy, allowed us to efficiently experiment with data augmentation. An interesting avenue for future work is to combine sign level and output text augmentation: e.g., use sign synthesis to generate new training signs for existing sentences, and simultaneously use text paraphrasing to generate new sentences for existing signs. Such a combination could address both the lack of visual-sign variations and the lack of linguistic variations. Another direction is to apply our augmentation in a scenario with multiple reference translations for evaluation; we hypothesize this would show clearer gains of the method, as single-reference BLEU can penalize legitimate paraphrases even after fine-tuning.

Finally, while we used a powerful proprietary LLM (GPT-4) to generate our paraphrases, it would be valuable to investigate if similar benefits can be obtained with open-source LLMs or simpler neural paraphrasers, and test different variations of the prompt, which would make this approach more accessible and reproducible for the research community.

References

Nikolas Adaloglou, Theocharis Chatzis, Ilias Papastratis, Andreas Stergioulas, Georgios Th Papadopoulos, Vassia Zacharopoulou, George J Xydopoulos, Klimnis Atzakas, Dimitris Papazachariou, and Petros Daras. 2020. A comprehensive study on sign language recognition methods. *arXiv preprint arXiv:2007.12530*.

Pedro Dal Bianco, Gast'on R'ıos, Franco Ronchetti, Facundo Quiroga, Oscar Stanchi, Waldo Hasperu'e, and Alejandro Rosete. 2022. Lsa-t: The first continuous argentinian sign language dataset for sign language translation. In *Advances in Artificial Intelligence – IBERAMIA 2022*, volume 13788 of *Lecture Notes in Computer Science*, page 293–304. Springer, Cham.

Danielle Bragg, Oscar Koller, Miriam Bellard, Larwan Berke, Naomi Caselli, and etc. 2019. Sign language recognition, generation, and translation: An interdisciplinary perspective. *ACM Transactions on Accessible Computing*, 12(2):5:1–5:44.

Necati Cihan Camgoz, Simon Hadfield, Oscar Koller, Hermann Ney, and Richard Bowden. 2018. Neural sign language translation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7784–7793.

Necati Cihan Camgoz, Oscar Koller, Simon Hadfield, and Richard Bowden. 2020. Sign language transformers: Joint end-to-end sign language recognition and translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10020–10030.

Shizhe Chen, Yuecong Wang, and etc. 2022. Two stream transformer networks for sign language trans-

lation. In *Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Ehsan Davoodi and 1 others. 2022. Improving low-resource classification via large language models for data augmentation. In *Proceedings of the 60th Annual Meeting of the ACL (Short Papers)*.

Xinyu Hu and 1 others. 2021. Text data augmentation made simple by leveraging llms: A case study on low-resource nlu tasks. In *Proceedings of the EMNLP 2021 (Findings)*.

Wesley Maia, António M. Lopes, and Sérgio A. David. 2025. Automatic sign language to text translation using mediapipe and transformer architectures. *Neurocomputing*, 642:130421.

Amit Moryossef, Kayo Yin, Graham Neubig, and Yoav Goldberg. 2021. Data augmentation for sign language gloss translation. In *Proceedings of the 1st International Workshop on Automatic Translation for Signed and Spoken Languages (AT4SSL) at MTSummit*, pages 1–11, Virtual. Association for Machine Translation in the Americas.

Carol Neidle and Christian Vogler. 2012. A new web interface to facilitate access to corpora: Development of the asllrp data access interface (dai). In *Proceedings of the 5th Workshop on the Representation and Processing of Sign Languages: Interactions between Corpus and Lexicon (LREC 2012)*, Istanbul, Turkey. European Language Resources Association (ELRA).

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318.

Gaston Gustavo Rios, Pedro Dal Bianco, Franco Ronchetti, Facundo Quiroga, Oscar Stanchi, Santiago Ponte Ahón, and Waldo Hasperué. 2025. Handcraft: Dynamic sign generation for synthetic data augmentation. *arXiv preprint arXiv:2508.14345*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the ACL*.

Stefanie Stoll and 1 others. 2020. Text2sign: Towards sign language production using neural machine translation and generative adversarial networks. In *Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*.

Andreas Voskou, Konstantinos P. Panousis, Harris Partaourides, Kyriakos Tolias, and Sotirios Chatzis. 2023. A new dataset for end-to-end sign language translation: The greek elementary school dataset. *arXiv preprint arXiv:2310.04753*.

Harry Walsh, Maksym Ivashechkin, and Richard Bowden. 2025. Using sign language production as data augmentation to enhance sign language translation. *arXiv preprint arXiv:2506.09643*.

Ryan Wong, Necati Cihan Camgoz, and Richard Bowden. 2024. Sign2gpt: Leveraging large language models for gloss-free sign language translation. In *International Conference on Learning Representations (ICLR)*.

Eta Yang. 2024. Signformer is all you need: Towards edge ai for sign language. *arXiv preprint arXiv:2411.12901*.

Tomáš Železný, Jakub Straka, Václav Javorek, Ondřej Valach, Marek Hrúz, and Ivan Gruber. 2023. Exploring pose-based sign language translation: Ablation studies and attention insights. *arXiv preprint arXiv:2507.01532*.

Tomáš Zelezný, Jakub Straka, Václav Javorek, Ondřej Valach, Marek Hrúz, and Ivan Gruber. 2025. Exploring pose-based sign language translation: Ablation studies and attention insights. *arXiv preprint arXiv:2507.01532*.

Multilingual Sign Language Translation with Unified Datasets and Pose-Based Transformers

Pedro Dal Bianco
III-LIDI
Universidad Nacional de La Plata
pdalbianco@lidi.info.unlp.edu.ar

Oscar Stanchi
CONICET
III-LIDI
ostanchi@lidi.info.unlp.edu.ar

Facundo Quiroga
III-LIDI
Comisión de Investigaciones Científicas
Universidad Nacional de La Plata
fquiroga@lidi.info.unlp.edu.ar

Franco Ronchetti
III-LIDI
Comisión de Investigaciones Científicas
Universidad Nacional de La Plata
fronchetti@lidi.info.unlp.edu.ar

Abstract

Sign languages are highly diverse across countries and regions, yet most Sign Language Translation (SLT) work remains monolingual. We explore a unified, *multi-target* SLT model trained jointly on four sign languages (German, Greek, Argentinian, Indian) using a standardized data layer. Our model operates on pose keypoints extracted with MediaPipe, yielding a lightweight and *dataset-agnostic* representation that is less sensitive to backgrounds, clothing, cameras, or signer identity while retaining motion and configuration cues. On RWTH-PHOENIX-Weather 2014T, Greek Sign Language Dataset, LSA-T, and ISLTranslate, naive joint training under a fully shared parameterization performs worse than monolingual baselines; however, a simple two-stage schedule: multilingual pre-training followed by a short language-specific fine-tuning, recovers and *surpasses* monolingual results on three datasets (PHOENIX14T: $+0.15$ BLEU-4; GSL: $+0.74$; ISL: $+0.10$) while narrowing the gap on the most challenging corpus (LSA-T: -0.24 vs. monolingual). Scores span from BLEU-4≈ 1 on open-domain news (LSA-T) to > 90 on constrained curricula (GSL), highlighting the role of dataset complexity. We release our code to facilitate training and evaluation of multilingual SLT models.

1 Introduction

Sign Language Translation (SLT) aims to convert sign language videos into spoken or written language text, helping bridge communication between deaf and hearing communities. SLT research has concentrated mostly on single-language benchmarks. Most notably, German Sign Language (DGS) with RWTH-PHOENIX-Weather 2014T has typically been used as baseline (Camgoz et al., 2018). Subsequently, transformer-based approaches demonstrated steady improvements (Camgoz et al., 2020), yet the diversity of sign languages and the scarcity of labeled data make it impractical to build and maintain one system per language. In contrast, multilingual modeling has transformed spoken/written machine translation (MT): a single shared model with target-language control tokens can learn to translate among many languages and even generalize in low-resource settings (Johnson et al., 2017). Bringing these ideas into SLT is promising but still relatively new. Recent work has shown the feasibility of multilingual SLT with architectural mechanisms to regulate parameter sharing across languages (Yin et al., 2022), and with clustering strategies to mitigate interference by grouping related languages (Zhang et al., 2025); in parallel, scaling data and directions is beginning to push SLT beyond narrow domains (Zhang et al., 2024). However, evaluation setups differ: some studies prefer many-to-one (many sign languages $\rightarrow$ one spoken language) for comparability, while others explore many-to-many configurations with multiple spoken targets, leaving open how far a *fully shared*, standard architecture can go when each sign language is translated into its *own* spoken language.

We address this question by training a single multilingual SLT model across four sign

Proceedings of the Workshop on Sign Language Processing (WSLP), pages 31–36

languages: DGS in RWTH-PHOENIX-Weather 2014T (DGS→German) (Camgoz et al., 2018), the Greek Sign Language Dataset (GSL→Greek) (Adaloglou et al., 2020), LSA-T (Argentinian Sign Language; LSA→Spanish) (Bianco et al., 2023), and ISLTranslate (Indian Sign Language; ISL→English) (Joshi et al., 2023). In this work we adapt *Signformer* (Yang, 2024) to operate on pose keypoints (hands, body, selected facial landmarks) extracted with MediaPipe (Lugaresi et al., 2019) instead of on CNN-derived visual embeddings. This choice yields a lightweight pipeline and can encourage cross-lingual transfer over motion patterns, albeit at the cost of some visual nuance in fine handshape/face details (for which robustness techniques continue to improve (Moryossef, 2024)). Practically, we unify data preparation across these corpora using an open-source library that standardizes formats and preprocessing, lowering barriers to multilingual experimentation.[1]

Our contributions can be listed as:

- **A multi-target multilingual SLT model** that translates each sign language into its *native spoken language* within a single, fully shared Transformer with no language-specific routing, complementing prior multilingual SLT designs that add sharing controls (Yin et al., 2022; Zhang et al., 2025).
- **A unified, open-source data layer** that harmonizes formats and preprocessing across RWTH-PHOENIX-Weather 2014T, Greek Elementary, LSA-T, and ISLTranslate, enabling streamlined multilingual training and evaluation (Bianco, 2025; Camgoz et al., 2018; Adaloglou et al., 2020; Bianco et al., 2023; Joshi et al., 2023).
- **A pose-keypoint adaptation of Signformer** (Yang, 2024) that replaces frame-based encoders with MediaPipe/BlazePose landmarks (Lugaresi et al., 2019; Bazarevsky et al., 2020), producing an efficient model suitable for cross-lingual sharing and deployment.
- **An empirical study of multilingual transfer** on four typologically and domain-diverse sign languages, showing that multilingual pre-training plus light language-specific fine-tuning *surpasses* monolingual baselines on PHOENIX14T, GSL, and ISL, and *narrows* (but does not close) the gap on LSA-T, consistent with trends observed as SLT scales (Zhang et al., 2024).

[1]Url anonimized for review purposes.

2 Related Work

Research on Sign Language Translation (SLT) began with the introduction of RWTH-PHOENIX-Weather 2014T and the first end-to-end baselines by Camgoz et al. (2018), which established the now-standard formulation of translating continuous sign video directly into spoken/written text. Subsequent transformer-based architectures advanced the state of the art by better modeling long-range temporal dependencies and jointly learning recognition and translation objectives (Camgoz et al., 2020). More recently, efforts to *scale* SLT in both data and directions highlighted that broader, multi-domain supervision can yield sizeable gains, especially when training setups move beyond a single sign language and a single target (Zhang et al., 2024). Nevertheless, the field has remained predominantly *monolingual*, in large part because sign corpora are scarce, heterogeneous, and difficult to align across languages, which complicates the construction of unified training pipelines and fair evaluation.

In contrast, multilingual modeling has been a defining trend in spoken/written neural machine translation (NMT). A single Transformer with a shared subword vocabulary and simple target-language control tokens can successfully learn many-to-many mappings, facilitate transfer for low-resource pairs, and even enable zero-shot generalization (Johnson et al., 2017). This paradigm naturally motivates multilingual SLT, where the model could amortize learning across sign languages that share articulatory patterns (e.g., hand trajectories, mouthings) or pragmatic structures, while still specializing to language-specific phenomena through conditioning.

Early steps toward multilingual SLT made this connection explicit. Yin et al. (2022) proposed and systematically explored many-to-one, one-to-many, and many-to-many setups, reporting that naive full sharing can cause interference, and that architectural controls (e.g., language-aware routing) help balance sharing versus specialization. Building on this line, Zhang et al. (2025) showed that automatically clustering sign languages into families and training family-specific models can further mitigate negative transfer while preserving the benefits of multilingual supervision. In parallel, work on

scaling SLT emphasized the importance of enlarging both data and translation directions, reinforcing that multilinguality, when properly managed, acts as both regularizer and data multiplier (Zhang et al., 2024). Against this backdrop, our study intentionally opts for a simpler design choice: a fully shared, standard Transformer without routing or family modules, paired with target-language tokens, to isolate how far basic parameter sharing can go in a *multi-target* configuration where each sign language maps to its native spoken language (akin to multilingual NMT) (Johnson et al., 2017).

Finally, the feasibility of multilingual SLT also hinges on the availability of diverse corpora beyond PHOENIX14T. Recent datasets such as the Greek Sign Language Dataset (Adaloglou et al., 2020), LSA-T for Argentinian Sign Language (Bianco et al., 2023), and ISLTranslate for Indian Sign Language (Joshi et al., 2023) broaden the linguistic and domain coverage for SLT research. Yet these resources differ in annotation conventions, domains, and difficulty, complicating joint training. This motivates standardized preprocessing layers and unified data schemas, which we leverage to train and evaluate a single pose-based model across multiple sign languages within one coherent framework.

3 Methodology

3.1 Datasets and Data Processing

Our study spans four SLT corpora with diverse languages, domains, and collection protocols: RWTH-PHOENIX-Weather 2014T (Camgoz et al., 2018), Greek Sign Language Dataset (GSL) (Adaloglou et al., 2020), LSA-T (Bianco et al., 2023), and ISLTranslate (Joshi et al., 2023). To make joint training feasible and comparable across languages, we standardize all datasets through a unified schema that normalizes splits, text preprocessing, and video-to-sequence conversion.

Concretely, videos are sampled at a consistent frame rate and processed with MediaPipe to extract 2D landmarks for hands, upper body, and selected facial regions (Lugaresi et al., 2019). We apply temporal smoothing and torso-based normalization to reduce jitter and scale variance, then select a subset of $\sim$150 features per frame (prioritizing hands/arms and a small set of facial cues) that best capture manual articulations and grammatical markers. Text targets are normalized and tokenized with a shared subword vocabulary. Figure 1 illustrates how the multilingual training set is formed by concatenating all corpora and converting each video to a pose-keypoint sequence, and, as a side benefit, using pose keypoints instead of raw frames also reduces sensitivity to dataset-specific nuisances (e.g., backgrounds, lighting, clothing, camera/viewpoint, signer appearance), promoting more invariant cross-corpus sharing while preserving motion/configuration cues.

3.2 Training Procedure

We adopt a two-stage schedule designed to leverage cross-lingual transfer while preserving language-specific nuances:

Stage 1 (Multilingual pre-training): we train a single fully shared model on the union of all datasets. To avoid overfitting to high-resource subsets, mini-batches are balanced by oversampling lower-resource languages, and early stopping is triggered on a macro-averaged validation BLEU across languages. The objective is standard cross-entropy over subword targets; we do not use gloss supervision.

Stage 2 (Language-specific fine-tuning): starting from the multilingual checkpoint, we fine-tune one model per language with a lower learning rate, which reliably recovers (and sometimes surpasses) the monolingual baselines. Throughout, the target-language token conditions the decoder so that the same parameters handle DGS$\rightarrow$German, GSL$\rightarrow$Greek, LSA$\rightarrow$Spanish, and ISL$\rightarrow$English within one architecture (Johnson et al., 2017). The full workflow is summarized in Figure 2.

3.3 Model Architecture

Our model builds on **Signformer** (Yang, 2024), a compact Transformer sequence-to-sequence architecture. We replace the original frame-based convolutional tokenization with a pose-based encoder: each frame's selected keypoints (hands, upper body, facial cues) are concatenated into a vector of dimension $d_{in} \approx 150$, normalized, and linearly projected to the model embedding space. Unlike multilingual SLT systems that introduce language-specific routing or adapters (Yin et al., 2022), we keep all parameters shared, emphasizing simplicity and parameter efficiency. Figure 3 illustrates the model's architecture.

Beyond efficiency, the pose-based encoder acts as an *inductive bias* toward signer and background invariant features, encouraging cross-lingual shar-

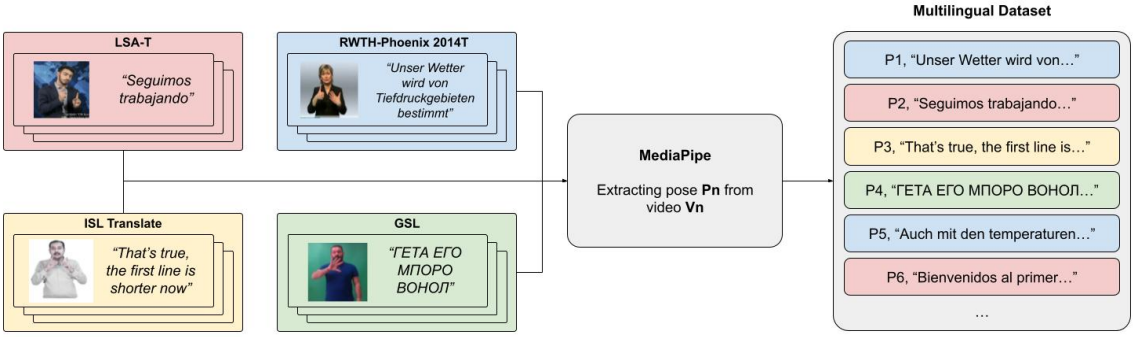

Figure 1: Multilingual dataset construction. Each corpus (PHOENIX14T, GSL, LSA-T, ISLTranslate) is standardized via a unified schema, then each video is converted into a sequence of MediaPipe keypoints (hands/body/face). The resulting pose sequences are concatenated into one multilingual training set with target-language tokens for multi-target decoding.

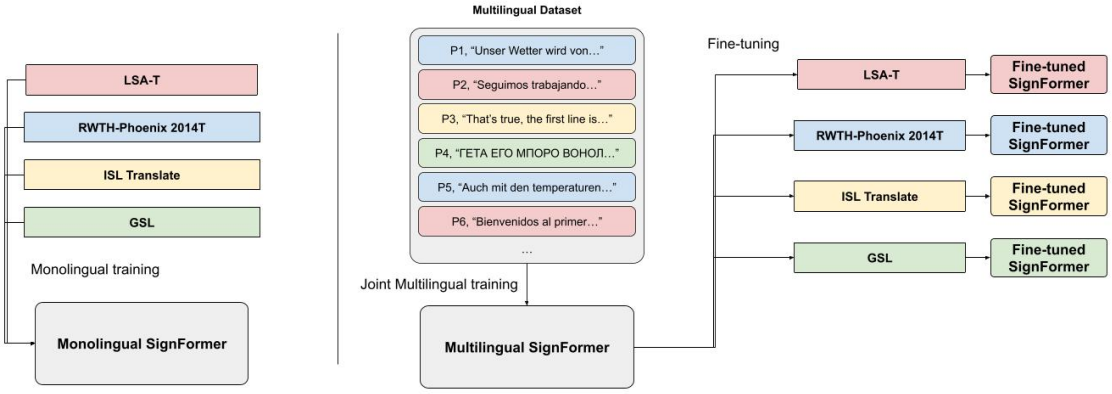

Figure 2: Monolingual training vs Two-stage multilingual training. *Stage 1:* joint pre-training of a fully shared Signformer on the concatenation of PHOENIX14T, GSL, LSA-T, and ISLTranslate with target-language tokens. *Stage 2:* light fine-tuning on each language's data starting from the multilingual checkpoint.

ing without overfitting to visual artefacts that differ across datasets.

4 Experiments and Results

We evaluate three training regimes: (i) **Monolingual baselines**—one pose-based *Signformer* per dataset; (ii) a **Multilingual joint** model trained naively on the concatenation of all corpora; and (iii) **Multilingual + fine-tuning**, where the joint model is lightly adapted to each language. We report case-insensitive BLEU-4 (Papineni et al., 2002), following standard SLT practice (Camgoz et al., 2018, 2020). Table 1 summarizes results for all four datasets.

Two clear trends emerge. First, *naive* joint training under a fully shared parameterization incurs sizeable drops relative to monolingual training (PHOENIX14T: -5.29; GSL: -31.31; LSA-T: -0.70; ISL: -2.02 BLEU), indicating capacity dilution and cross-language interference when mixing heterogeneous sign languages without stronger sharing controls. Second, the *two-stage* schedule is crucial: brief, low–learning-rate fine-tuning

Dataset	Monolingual	Joint	+Fine-tune
PHOENIX14T (DGS→De)	9.56	4.27	**9.71**
GSL Dataset (GSL→Gr)	94.38	63.07	**95.12**
LSA-T (LSA→Es)	**1.18**	0.48	0.94
ISL-Translate (ISL→En)	2.61	0.59	**2.71**

Table 1: BLEU-4 on test sets for monolingual baselines, a single multilingual joint model, and multilingual pre-training followed by language-specific fine-tuning. Best per row in **bold**.

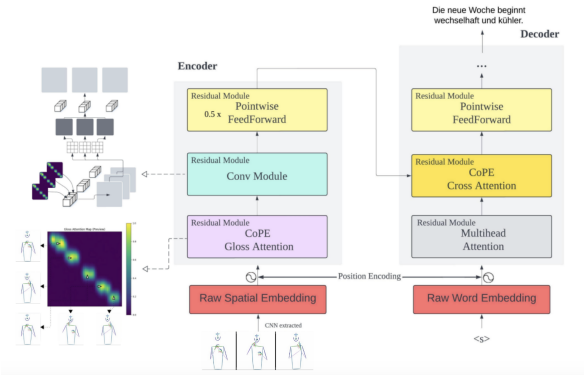

Figure 3: Overview of the adapted *Signformer* architecture (originally taken from (Yang, 2024)) for multilingual SLT using pose keypoints as input. Instead of frame-based visual tokens, each frame's concatenated hand, upper-body, and selected facial landmarks (after normalization and linear projection) feed the encoder. A shared decoder, conditioned on a target-language token, generates the translation.

largely restores and, on *three* datasets, *surpasses* monolingual performance (PHOENIX14T $+0.15$, GSL $+0.74$, ISL $+0.10$ vs. monolingual), while LSA-T remains challenging (joint $\rightarrow$ FT: $+0.46$, ending -0.24 below monolingual). These outcomes mirror multilingual MT and SLT scaling results—multilingual pre-training acts as a regularizer and data multiplier, but sensitive adaptation is required to realize gains across languages and domains (Johnson et al., 2017; Zhang et al., 2024).

Dataset complexity and representation effects. The spread in BLEU-4 reflects intrinsic differences across corpora. GSL's curriculum-oriented content and constrained phrasing may partly explain its very high scores, whereas LSA-T's news-style, open-domain content, signer variability, and potential annotation/pose-estimation noise make it considerably harder. Moreover, pose-based inputs—while enabling compact, deployable models—trade some fine-grained appearance cues (e.g., subtle handshapes, facial expression nuances) for efficiency, which can widen the gap to video-based SOTA on the most challenging settings (Yang, 2024). Still, the fact that PHOENIX14T and GSL not only recover but slightly surpass monolingual baselines after multilingual pre-training suggests that shared motion/configuration patterns are learnable with keypoints when paired with light language-specific adaptation.

5 Conclusion

We presented a multi-target multilingual SLT system that translates DGS$\rightarrow$German, GSL$\rightarrow$Greek, LSA$\rightarrow$Spanish, and ISL$\rightarrow$English within a single, fully shared Transformer, enabled by a unified data layer and pose-based inputs. Naive joint training alone is insufficient—performance drops on all four datasets—but a simple two-stage schedule (multilingual pre-training followed by brief language-specific fine-tuning) reliably recovers and *surpasses* monolingual baselines on PHOENIX14T, GSL, and ISL, while narrowing (though not closing) the gap on LSA-T. These findings echo multilingual MT and recent SLT scaling results: cross-lingual transfer is beneficial, but careful adaptation is necessary to mitigate interference (Johnson et al., 2017; Zhang et al., 2024).

Relative to prior multilingual SLT that commonly evaluates many-to-one into a single target language, our study emphasizes a *multi-target* configuration aligned with each dataset's native spoken language and demonstrates that a compact, pose-based *Signformer* can serve as an effective backbone for this setting. While pose inputs may underperform on unconstrained domains like LSA-T, they enable lightweight, privacy-friendly models.

References

Nikolas Adaloglou, Theocharis Chatzis, Ilias Papastratis, Andreas Stergioulas, Georgios Th Papadopoulos, Vassia Zacharopoulou, George J Xydopoulos, Klimnis Atzakas, Dimitris Papazachariou, and Petros Daras. 2020. A comprehensive study on sign language recognition methods. *arXiv preprint arXiv:2007.12530*.

Valentin Bazarevsky, Ivan Grishchenko, Karthik Raveendran, Tyler Zhu, Fan Zhang, and Matthias Grundmann. 2020. Blazepose: On-device real-time body pose tracking. *arXiv preprint arXiv:2006.10204*.

Pedro Dal Bianco. 2025. Slt datasets downloader. GitHub repository. Accessed 2025-09-23.

Pedro Dal Bianco, Gastón Ríos, Franco Ronchetti, Facundo Quiroga, Oscar Stanchi, Waldo Hasperué, and Alejandro Rosete. 2023. Lsa-t: The first continuous argentinian sign language dataset for sign language translation. In *Advances in Artificial Intelligence – IBERAMIA 2022*, volume 13788 of *Lecture Notes in Computer Science*, pages 293–304. Springer, Cham.

Necati Cihan Camgoz, Simon Hadfield, Oscar Koller, Hermann Ney, and Richard Bowden. 2018. Neural sign language translation. In *Proceedings of the*

IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 7784–7793.

Necati Cihan Camgoz, Oscar Koller, Simon Hadfield, and Richard Bowden. 2020. Sign language transformers: Joint end-to-end sign language recognition and translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Melvin Johnson, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2017. Google's multilingual neural machine translation system: Enabling zero-shot translation. *Transactions of the Association for Computational Linguistics*, 5:339–351.

Abhinav Joshi, Susmit Agrawal, and Ashutosh Modi. 2023. Isltranslate: Dataset for translating indian sign language. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 10466–10475.

Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Uboweja, Michael Hays, Fan Zhang, Chuo-Ling Chang, Ming Guang Yong, Juhyun Lee, Wan-Teh Chang, Wei Hua, Manfred Georg, and Matthias Grundmann. 2019. Mediapipe: A framework for building perception pipelines. *arXiv preprint arXiv:1906.08172*.

Amit Moryossef. 2024. Optimizing hand region detection in mediapipe holistic full-body pose estimation to improve accuracy and avoid downstream errors. *arXiv preprint arXiv:2405.03545*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318.

Eta Yang. 2024. Signformer is all you need: Towards edge AI for sign language. *arXiv preprint arXiv:2411.12901*.

Aoxiong Yin, Zhou Zhao, Weike Jin, Meng Zhang, Xingshan Zeng, and Xiaofei He. 2022. Mlslt: Towards multilingual sign language translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Biao Zhang, Garrett Tanzer, and Orhan Firat. 2024. Scaling sign language translation. In *Advances in Neural Information Processing Systems 37 (NeurIPS 2024)*.

Ruiquan Zhang, Cong Hu, Pei Yu, and Yidong Chen. 2025. Improving multilingual sign language translation with automatically clustered language family information. In *Proceedings of the 31st International Conference on Computational Linguistics (COLING 2025)*.

Continuous Fingerspelling Dataset for Indian Sign Language

Kirandevraj R[1] **Vinod K Kurmi**[2] **Vinay P Namboodiri**[3] **CV Jawahar**[1]
[1] IIIT Hyderabad, India.
[2] IISER Bhopal, India.
[3] University of Bath, UK.
kirandevraj.r@research.iiit.ac.in, vinodkk@iiserb.ac.in,
vpn22@bath.ac.uk, jawahar@iiit.ac.in

Abstract

Fingerspelling enables signers to represent proper nouns and technical terms letter-by-letter using manual alphabets, yet remains severely under-resourced for Indian Sign Language (ISL). We present the first continuous fingerspelling dataset for ISL, extracted from the ISH News YouTube channel, in which fingerspelling is accompanied by synchronized on-screen text cues. The dataset comprises 1,308 segments from 499 videos, totaling 70.85 minutes and 14,814 characters, with aligned video-text pairs capturing authentic coarticulation patterns. We validated the dataset quality through annotation using a proficient ISL interpreter, achieving a 90.67% exact match rate for 150 samples. We further established baseline recognition benchmarks using a ByT5-small encoder-decoder model, which attains 82.91% Character Error Rate after fine-tuning. This resource supports multiple downstream tasks, including fingerspelling transcription, temporal localization, and sign generation. The dataset is available at the following link: https://kirandevraj.github.io/ISL-Fingerspelling/.

1 Introduction

Sign languages serve as the primary communication medium for over 70 million deaf individuals worldwide, yet technological support for these languages remains vastly underrepresented compared to spoken languages. Fingerspelling serves as a critical bridging mechanism in sign languages, allowing signers to spell words from spoken languages letter-by-letter using a dedicated manual alphabet (Padden and Gunsauls, 2003). While American Sign Language (ASL) has benefited from substantial datasets, with recent collections encompassing millions of characters and hundreds of hours of data that enable significant advances in recognition accuracy (Georg et al., 2024), research on ISL fingerspelling recognition has been severely limited by the absence of comparable resources.

The structure of manual alphabets varies across sign languages; some employ one-handed configurations (such as the American Sign Language), while others utilize two-handed systems (such as the ISL). Despite being a subset of the broader sign language lexicon, fingerspelling plays a substantial role in communication. Recognition of fingerspelled sequences presents significant computational challenges owing to two primary factors: first, the movements are characterized by rapid, subtle articulations with extensive co-articulation between consecutive letters, making visual parsing difficult (Patrie and Johnson, 2011); second, fingerspelling predominantly encodes out-of-vocabulary items, including proper nouns, technical terminology, and domain-specific vocabulary, which lack established sign equivalents, limiting the applicability of lexicon-based recognition approaches (Padden and Gunsauls, 2003). This signifies a dedicated focus on fingerspelling.

The development of automated ISL fingerspelling recognition systems has been severely constrained by the absence of large-scale standardized benchmark datasets. Although ASL benefits from substantial resources such as FSboard (Georg et al., 2024) with over 3 million characters and ChicagoFSWild+ (Shi et al., 2019) with 55,232 sequences from 260 signers, existing ISL datasets primarily focus on isolated sign recognition or continuous sentence-level translation tasks(Joshi et al., 2023, 2024), with limited attention to fingerspelling as a distinct recognition challenge.

To address this critical gap in ISL processing resources, we present the first dedicated benchmark dataset for continuous Indian Sign Language fingerspelling recognition, comprising 1,308 fingerspelling segments extracted from 499 ISH News YouTube videos. The dataset totals 70.85 minutes of video data across 1,308 annotated segments containing 14,814 characters, capturing authentic coar-

Proceedings of the Workshop on Sign Language Processing (WSLP), pages 37–42

Figure 1: ISH News fingerspelling example showing eight frames of the word "formaldehyde." The side panel displays letters (F-O-R-M-A-L-D-E) that are sequentially synchronized with the signer's hand configurations. These visual cues serve as our annotation source.

ticulation patterns in naturalistic signing contexts. We establish baseline recognition results using a ByT5-small encoder-decoder transformer model, achieving 82.91% Character Error Rate after fine-tuning and providing reference performance metrics for future research. We have made our dataset and annotations publicly available to facilitate reproducible research and support the broader development of ISL processing technologies for deaf and hard-of-hearing communities.

2 Related Works

Fingerspelling recognition has been extensively studied for American Sign Language using datasets such as ChicagoFSVid (Kim et al., 2016), ChicagoFSWild (Shi et al., 2018), ChicagoFSWild+ (Shi et al., 2019) with 55,232 sequences from 260 signers, and FSboard (Georg et al., 2024) with over 3 million characters. Recent work has extended to fingerspelling span detection in longer videos (Shi et al., 2022; R et al., 2022), enabling automatic localization of fingerspelling segments. In contrast, Indian Sign Language fingerspelling research has primarily focused on image-based handshape classification (Suchithra et al., 2025; Langote et al., 2024), recognizing static handshapes from single frames rather than addressing temporal dynamics in continuous sequences.

Indian Sign Language research has witnessed significant growth with several dataset contributions. Large-scale translation datasets include iSign (Joshi et al., 2024) with 118k video-English pairs and ISLTranslate (Joshi et al., 2023) with 31k pairs from educational videos. Isolated sign recognition is supported by INCLUDE (Sridhar et al., 2020) (263 signs, 4,287 videos), ISL-CSLTR (Elakkiya and Natarajan, 2021) (700 sentence videos, 1,036-word vocabulary), and CISLR (Joshi et al., 2022) (7,050 videos, 4,765 words). However, fingerspelling has been severely underexplored. Existing fingerspelling datasets are exclusively image-based, focusing on isolated alphabet recognition: ISL Fingerspelling (Dongare et al., 2025) provides 14K images, ISL Skeletal (Johnson et al., 2023) contains 3.6K images per letter, ISL Hand Gesture (Biswas, 2024) offers 14.3K images, and Static Gestures of ISL (Singh et al., 2022) include 102K images. None of these captures the temporal dynamics, coarticulation patterns, or continuous sequences necessary for realistic fingerspelling transcription. We address this gap with the first continuous ISL fingerspelling dataset.

3 Fingerspelling Benchmark

3.1 Dataset Creation

We created a continuous fingerspelling dataset from the ISH News YouTube channel by leveraging naturally occurring fingerspelling instances in news videos. The channel employs a distinctive visual cue system where fingerspelling segments, typically proper nouns such as person names and place names, are accompanied by synchronized on-screen text that displays each letter sequentially below a contextual image, timed to match the signer's fingerspelling gestures (Figure 1). We identified 499 unique videos containing these visual cues from which we extracted 1,308 fingerspelling instances. Using the ELAN annotation tool (Brugman and Russel, 2004; Max Planck Institute for Psycholinguistics, 2023), we manually marked the temporal boundaries of each fingerspelling segment around the start and end of the text animation and associated them with the corresponding words or phrases from the visual cues.

Dataset	Type & Size
ISL Fingerspelling (Dongare et al., 2025)	14K images
ISL Skeletal (Johnson et al., 2023)	3.6K img/letter
ISL Hand Gesture (Biswas, 2024)	14.3K images
Static Gestures (Singh et al., 2022)	102K images
Continuous ISL Fingerspelling (Ours)	**1,308 seg. 14,814 chars**

Table 1: Comparison with existing ISL fingerspelling datasets. Prior work focuses on static images of isolated letters, while our dataset provides continuous video sequences.

Validation Outcome	Count
Exact match	136
Signer skipped space	7
Signer made error	5
Too fast to verify	2
Total	**150**

Table 2: Interpreter validation results on 150 randomly sampled segments after correcting validator transcription errors.

This approach enables annotation of continuous fingerspelling sequences from authentic YouTube content.

Video Processing: Following temporal boundary annotation, we preprocessed the video segments to isolate the signer region and remove extraneous visual elements such as the side panel containing text cues. For each annotated segment, we first extracted the corresponding video clips based on marked timestamps. We then employed YOLOv8 (Varghese and M, 2024) person detection on randomly sampled frames to identify the signer's bounding box and select the rightmost detected person (signers consistently appear on the right side of the frame in ISH News videos). To ensure robust cropping across varying camera angles and signer movements, we aggregated bounding boxes across multiple sampled frames using median coordinates. Finally, we applied these computed crop coordinates to extract signer-only video segments, producing 1,308 preprocessed clips containing the signer performing fingerspelling gestures without on-screen text overlays or background elements. This preprocessing ensures that models trained on our dataset focus on visual signing features rather than textual cues.

3.2 Dataset statistics

The dataset comprises 1,308 fingerspelling segments extracted from 499 videos, totaling 70.85 minutes of signing content from 3 unique signers. Among these videos, 408 video IDs overlapped with the iSign (Joshi et al., 2024) sentence-level translation dataset, whereas 91 were not previously included in iSign. The extracted segments contain 14,814 characters total. Alphabets constituted 92.64% (13,724 characters), reflecting the predominantly textual nature of fingerspelling in proper nouns. Spaces accounted for 6.82% (1,011 characters), separating multi-word names and phrases. Numbers appear minimally at 0.17% (25 characters), corresponding to occasional numeric references in names or titles. Other characters comprise 0.36% (54 characters), and primarily include periods used in abbreviations and initials, hyphens in compound names, and occasional parentheses. Table 1 compares our dataset with existing ISL fingerspelling resources, highlighting the shift from static image-based datasets to continuous video sequences.

3.3 Annotation Validation

To validate the reliability of the cue-based annotations, we conducted validation on 150 randomly sampled segments (totaling 8.20 minutes) with a proficient Indian Sign Language interpreter. The interpreter independently transcribed each segment by watching fingerspelling gestures without access to visual cues. In the first round, we identified discrepancies between the cue-based annotations and interpreter transcriptions in 27 cases. Upon closer examination in the second round, we determined that 13 discrepancies resulted from validator transcription errors, which we corrected. The remaining 14 cases reflected actual issues in the source videos or extraction process, as detailed in Table 2. After corrections, 136 of 150 segments (90.67%) achieved exact match with the interpreter validation, confirming the overall reliability of the cue-based annotation approach.

3.4 Fingerspelling Tasks

Our dataset supports three key tasks in sign language processing: **Transcription** converts continuous fingerspelling video segments into character sequences, handling coarticulation, signing speed variations, and ambiguous handshapes. In Section 4, we establish baseline benchmarks for this task. **Temporal Localization** identifies fingerspelling segment boundaries within longer videos. Our annotations provide temporal boundaries for

cue-accompanied fingerspelling instances. The total number of hours of these 499 videos is 20. **Generation** produces signing videos from text with realistic handshapes and transitions. Our dataset can serve as a reference for fingerspelling.

4 Models, Experiments and Results

4.1 Baseline Models

Experimental Setup We conduct two experiments to evaluate fingerspelling recognition performance. First, we evaluated a model pretrained on the iSign dataset (Joshi et al., 2024) in a zero-shot setting on our fingerspelling test set to assess transfer learning from general ISL to fingerspelling. During iSign pre-training, all video IDs overlapping with our fingerspelling dataset were excluded from the training data to prevent data leakage. Second, we fine-tuned the pretrained model on fingerspelling-specific data. We split our fingerspelling dataset based on video ID overlap with iSign: 1,104 segments from videos present in iSign served as the training set, while 204 segments from videos not in iSign formed the test set. The model performance was evaluated using the Character Error Rate (CER).

Model Architecture We adopt the modeling approach from FLEURS-ASL and FSboard (Georg et al., 2024), using a ByT5-small encoder-decoder Transformer. We extracted 75 keypoints (33 body pose, 21 per hand) from MediaPipe Holistic (Lugaresi et al., 2019; Grishchenko and Bazarevsky, 2020) at 15 Hz, yielding 225-dimensional vectors (75 keypoints × 3 coordinates). The iSign dataset provides poses in pose-format (Moryossef et al., 2021). Preprocessing included shoulder-distance normalization for scale invariance, downsampling to 15 Hz, zero-filling for missing keypoints, and padding/truncation to fixed sequence length. We selected ByT5 over subword models because of its character-level tokenization in fingerspelling (Tanzer, 2024). The landmarks were projected through a two-layer feedforward network with layer normalization and dropout into the 1472-dimensional input space of the transformer.

Training We employ a two-stage training strategy: Stage 1 freezes the ByT5 parameters while training only the pose embedding projection for 40 epochs with a learning rate of 1e-4 and batch size of 16, followed by Stage 2 which unfreezes all parameters for end-to-end fine-tuning for 20 epochs with a reduced learning rate of 1e-5 and batch size of 4. We used the AdamW optimizer with gradient clipping, gradient accumulation (steps=2), and 500-step warmup. Training was performed on two RTX 4090 GPUs, completing in approximately 18 hours.

Evaluation Set	**CER (%)**
Pretrained on iSign (zero-shot)	
Test (204 seg.)	432.44
Full dataset (1,308 seg.)	433.06
Fine-tuned on fingerspelling (1,104 train)	
Test (204 seg.)	82.91

Table 3: ByT5-small model performance on fingerspelling transcription. The model was evaluated in zero-shot (pretrained only on iSign) and fine-tuned settings. Test set contains segments from videos not in iSign.

4.1.1 Results

Table 3 presents our baseline results under two evaluation conditions. Without fine-tuning on fingerspelling-specific data, the model pretrained only on general ISL achieved a CER of 432.44% on the test set and 433.06% on the full dataset, demonstrating extremely limited zero-shot transfer capability. After fine-tuning on the fingerspelling training split, performance improved substantially to 82.91% CER, representing an 80.8% relative reduction in error rate. This large performance gap indicates that while learned visual representations from general ISL provide some foundation, fingerspelling recognition requires domain-specific adaptation because of its distinct character-level structure and rapid hand movements. The post-fine-tuning CER of 82.91% establishes a baseline for future work, although it remains substantially higher than the state-of-the-art ASL fingerspelling results (e.g., FSboard achieves 10% CER (Georg et al., 2024)), highlighting the unique challenges and data scarcity for ISL fingerspelling recognition.

5 Conclusion

We present the first continuous fingerspelling dataset for Indian Sign Language, comprising 1,308 video segments from 499 videos totaling 70.85 minutes and 14,814 characters. Our baseline ByT5-small model achieved 82.91% CER after fine-tuning, establishing initial benchmarks while revealing substantial room for improvement. Future work should prioritize expanding dataset scale and signer diversity, investigating transfer learning from larger fingerspelling datasets, and developing improved methods to handle coarticulation patterns in ISL fingerspelling.

Limitations and Ethical Considerations

5.1 Limitations

Our cue-based extraction achieves 90.67% exact match with expert validation after correcting validator errors, with remaining discrepancies from signer errors in videos (5 cases), missing spaces (7 cases), and overly rapid signing (2 cases). The dataset's reliance on ISH News videos with a limited number of professional signers constrains demographic diversity and may reduce generalization to casual or regional signing styles. Temporal boundaries were manually annotated by the first author based on observed correspondence between visual cues and fingerspelling gestures, introducing potential subjectivity in boundary placement. The predominance of proper nouns and news-related terminology may limit model performance in technical jargon or conversational fingerspelling. The relatively small scale (1,308 segments, 70.85 minutes) limits the training of large-scale models and the comprehensive evaluation across diverse fingerspelling scenarios.

5.2 Ethical Considerations

We used publicly available ISH News YoutTube videos, with 407 of 499 videos already in the iSign dataset (Joshi et al., 2024) (which obtained ISH News permission for research use) and the remaining 92 videos featuring identical signers and settings. Sign language videos capture facial expressions and body postures, enabling signer identification and raising privacy concerns despite publicly available nature and institutional permissions. Professional broadcast signers do not represent full ISL community diversity, including regional variations and casual signing styles. Models trained on these broadcast-quality data should not be deployed in accessibility applications without extensive community validation, as generalization gaps could harm deaf users.

Acknowledgements

We are grateful to ISH News for making their sign language news videos publicly available on YouTube. We acknowledge the iSign dataset team for obtaining usage permissions from ISH News, which facilitated our research. We thank the ISL interpreter who assisted with the annotation validation process.

References

Sougatamoy Biswas. 2024. ISL Hand Gesture Dataset.

Hennie Brugman and Albert Russel. 2004. Annotating multi-media/multi-modal resources with elan. In *International Conference on Language Resources and Evaluation*.

Tanvi Dongare, Gaurika Nawani, Aditya Deshpande, Ayaan Shaikh, and Dr. Deepali Javale. 2025. Isl fingerspelling image dataset.

R Elakkiya and B Natarajan. 2021. Isl-csltr: Indian sign language dataset for continuous sign language translation and recognition. *Mendeley Data*, 1.

Manfred Georg, Garrett Tanzer, Saad Hassan, Max Shengelia, Esha Uboweja, Sam S. Sepah, Sean Forbes, and Thad Starner. 2024. Fsboard: Over 3 million characters of asl fingerspelling collected via smartphones. *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13897–13906.

Ivan Grishchenko and Valentin Bazarevsky. 2020. Mediapipe holistic - simultaneous face, hand and pose prediction, on device.

Jans Johnson, Jisha Joseph, Maris Reji, and Megha George. 2023. Indian sign language skeletal-point numpy array using mediapipe.

Abhinav Joshi, Susmit Agrawal, and Ashutosh Modi. 2023. Isltranslate: Dataset for translating indian sign language. In *Annual Meeting of the Association for Computational Linguistics*.

Abhinav Joshi, Ashwani Bhat, Pradeep Raj M S, Priya Gole, Shashwat Gupta, Shreyansh Agarwal, and Ashutosh Modi. 2022. Cislr: Corpus for indian sign language recognition. In *Conference on Empirical Methods in Natural Language Processing*.

Abhinav Joshi, Romit Mohanty, Mounika Kanakanti, Andesha Mangla, Sudeep Choudhary, Monali Barbate, and Ashutosh Modi. 2024. isign: A benchmark for indian sign language processing. *ArXiv*, abs/2407.05404.

Taehwan Kim, Jonathan Keane, Weiran Wang, Hao Tang, Jason Riggle, Gregory Shakhnarovich, Diane Brentari, and Karen Livescu. 2016. Lexicon-free fingerspelling recognition from video: Data, models, and signer adaptation. *Comput. Speech Lang.*, 46:209–232.

Vaishali Langote, Aditya Deshpande, Tanvi Dongare, Gaurika Nawani, Ayaan Shaikh, and Arhaan Mulani. 2024. Bridging the gap: Isl fingerspelling to text, sentiment analysis and language conversion. In *2024 8th International Conference on Computing, Communication, Control and Automation (ICCUBEA)*, pages 1–6. IEEE.

Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Uboweja, Michael Hays, Fan Zhang, Chuo-Ling Chang, Ming Guang Yong, Juhyun Lee, Wan-Teh Chang, Wei Hua, Manfred Georg, and Matthias Grundmann. 2019. Mediapipe: A framework for building perception pipelines. *ArXiv*, abs/1906.08172.

Max Planck Institute for Psycholinguistics. 2023. Elan.

Amit Moryossef, Mathias Müller, and Rebecka Fahrni. 2021. pose-format: Library for viewing, augmenting, and handling .pose files. `https://github.com/sign-language-processing/pose`.

Carol Padden and Darline Clark Gunsauls. 2003. How the alphabet came to be used in a sign language. *Sign Language Studies*, 4:10 – 33.

Carol J Patrie and Robert E Johnson. 2011. *RSVP: Fingerspelled word recognition through rapid serial visual presentation*. DawnSignPress.

Prajwal K R, Hannah Bull, Liliane Momeni, Samuel Albanie, Gül Varol, and Andrew Zisserman. 2022. Weakly-supervised fingerspelling recognition in british sign language videos. *ArXiv*, abs/2211.08954.

Bowen Shi, Diane Brentari, Greg Shakhnarovich, and Karen Livescu. 2022. Searching for fingerspelled content in american sign language. In *Annual Meeting of the Association for Computational Linguistics*.

Bowen Shi, Aurora Martinez Del Rio, Jonathan Keane, Diane Brentari, Greg Shakhnarovich, and Karen Livescu. 2019. Fingerspelling recognition in the wild with iterative visual attention. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 5399–5408.

Bowen Shi, Aurora Martinez Del Rio, Jonathan Keane, Jonathan Michaux, Diane Brentari, Gregory Shakhnarovich, and Karen Livescu. 2018. American sign language fingerspelling recognition in the wild. *2018 IEEE Spoken Language Technology Workshop (SLT)*, pages 145–152.

Animesh Singh, Sunil K. Singh, Ajay Mittal, and Brij B. Gupta. 2022. Static gestures of Indian Sign Language (ISL) for English Alphabet, Hindi Vowels and Numerals.

Advaith Sridhar, Rohith Gandhi Ganesan, Pratyush Kumar, and Mitesh M. Khapra. 2020. Include: A large scale dataset for indian sign language recognition. *Proceedings of the 28th ACM International Conference on Multimedia*.

M. Suchithra, Ayushi Gupta, and Abhilasha Kasaraneni. 2025. Fingerspelling for indian sign language using swin transformer. *2025 6th International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV)*, pages 947–952.

Garrett Tanzer. 2024. Fingerspelling within sign language translation. *ArXiv*, abs/2408.07065.

Rejin Varghese and Sambath. M. 2024. Yolov8: A novel object detection algorithm with enhanced performance and robustness. *2024 International Conference on Advances in Data Engineering and Intelligent Computing Systems (ADICS)*, pages 1–6.

Enhancing Indian Sign Language Translation via Motion-Aware Modeling

Anal Roy Chowdhury and Debarshi Kumar Sanyal
Indian Association for the Cultivation of Science, Kolkata, India
analroychowdhury084@gmail.com, debarshi.sanyal@iacs.res.in

Abstract

Sign language translation (SLT) has witnessed rapid progress in the deep learning community across several sign languages, including German, American, British, and Italian. However, Indian Sign Language (ISL) remains relatively underexplored. Motivated by recent efforts to develop large-scale ISL resources, we investigate how existing SLT models perform on ISL data. Specifically, we evaluate three approaches: (i) training a transformer-based model, (ii) leveraging visual-language pretraining, and (iii) tuning a language model with pre-trained visual and motion representations. Unlike existing methods that primarily use raw video frames, we augment the model with optical flow maps to explicitly capture motion primitives, combined with a multi-scale feature extraction method for encoding spatial features (SpaMo-OF). Our approach achieves promising results, obtaining a BLEU-4 score of 8.58 on the iSign test set, establishing a strong baseline for future ISL translation research.

1 Introduction

Sign languages bridge the communication gap between deaf and hearing communities. The World Health Organization[1] predicts that by 2050, over 700 million people will experience disabling hearing loss. This growing prevalence highlights the urgent need for assistive technologies that can support inclusion and accessibility. Sign language translation (SLT) has emerged as a promising research area, with extensive studies on German, American, and British Sign Languages, largely enabled by the availability of large-scale datasets such as PHOENIX-2014T (German) (Camgoz et al., 2018), How2Sign (Duarte et al., 2021) and OpenASL (Shi et al., 2022) (American), and BOBSL (British) (Albanie et al., 2021). SLT methods in the literature typically fall into two categories: gloss-based and gloss-free. Gloss supervision (Camgoz et al., 2020; Chen et al., 2022) has been shown to improve translation quality, but many datasets lack gloss annotations due to their high cost. The shortage of trained sign language experts and the expense of annotation have therefore pushed the community toward gloss-free approaches (Lin et al., 2023; Gong et al., 2024; Wong et al., 2024; Jang et al., 2025; Hwang et al., 2025).

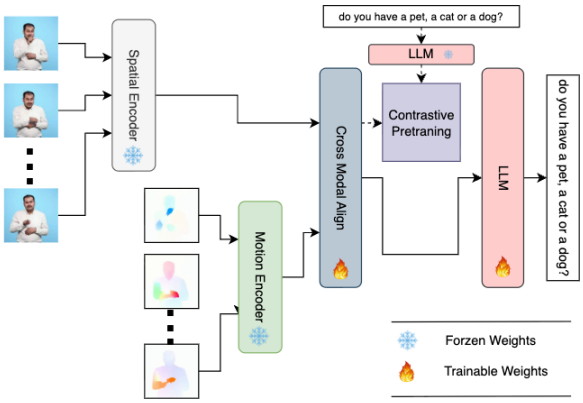

Figure 1: Overall architecture of our SLT framework.

All existing studies have primarily focused on developing translation systems for well-resourced sign languages such as German, Chinese, American, and British. In contrast, low-resource sign languages like Indian Sign Language (ISL) remain largely overlooked. The recent release of large-scale ISL datasets such as ISLTranslate (Joshi et al., 2023) and iSign (Joshi et al., 2024), together with the underperformance of ISL when using existing models, motivates us to examine whether recent architectures that perform well on other sign languages can also generalize to ISL. To this end, we evaluate three representative approaches for SLT: (i) training transformers from scratch (Camgoz et al., 2020), (ii) visual-language pretraining using contrastive learning (Zhou et al., 2023), and (iii) finetuning Large Language Models (LLMs) with multi-scale

[1] https://www.who.int/news-room/fact-sheets/detail/deafness-and-hearing-loss

Proceedings of the Workshop on Sign Language Processing (WSLP), pages 43–50

spatial features and motion features extracted using a pre-trained backbone. We further adopt the recently proposed SpaMO model (Hwang et al., 2025), which achieves state-of-the-art results on several SLT benchmarks but has not yet been applied to ISL. SpaMO leverages multi-scale feature extraction from input video frames to improve downstream translation performance. We extend it by incorporating optical flow features, which significantly improve results on the iSign dataset by better capturing motion cues in signed gestures. To ensure reliable evaluation, we curated a noise-free subset of iSign and conducted experiments on this data.

Our main contributions are as follows: i) We conduct the first systematic evaluation of three representative SLT approaches on Indian Sign Language. ii) We enhance translation performance by augmenting SpaMO with optical flow features to capture motion primitives alongside multi-scale spatial features. iii) We curate and release a carefully selected subset of iSign to enable robust evaluation for future ISL translation research. The dataset can be accessed using the following link `https://github.com/Analroy/SpaMo-OF.git`.

2 Related Work

The SLT framework has evolved from RNN-based to Transformer-based architectures (Camgoz et al., 2018, 2020), where sequential models take CNN-based features as input. More recent approaches to building better translation systems focus on capturing richer representations, such as pose features or a combination of pose and RGB features (Chen et al., 2022), as well as sign-aware representations (Hu et al., 2021, 2023). To learn stronger sign-specific representations, (Zhou et al., 2023) proposed pretraining the visual encoder, while (Lin et al., 2023) employed contrastive pretraining of the visual encoder using pseudo-gloss supervision. Recent advances in LLMs have also attracted attention (Gong et al., 2024; Wong et al., 2024; Chen et al., 2024), as researchers explore leveraging large-scale pretrained models and adapting them to domain-specific data using parameter-efficient methods such as LoRA (Hu et al., 2022). In contrast to these resource-intensive pretraining approaches, (Hwang et al., 2025) demonstrated that multi-scale features and motion features extracted from a frozen model, when aligned to the LLM space, can achieve improved translation performance by applying LoRA tuning only to the language model.

3 Method

We aim to translate a sign language video $V = [f_1, f_2, \ldots, f_T]$ into a spoken language sentence $Y = [w_1, w_2, \ldots, w_S]$ by leveraging complementary spatial and motion features, aligning them with textual representations, and decoding them with an LLM. The overall pipeline is illustrated in Fig.1.

3.1 Feature Extraction

Spatial Features: A Vision Transformer captures multiscale spatial representations S^2 (Shi et al., 2025) from input frames, following (Hwang et al., 2025). These features encode detailed hand shapes and body postures across scales.

Motion Features: To explicitly capture temporal dynamics, we combine:

Optical Flow: Computed using Global Motion Aggregation (GMA) (Jiang et al., 2021), which robustly handles occluded hand movements (Fig. 2).

VideoMAE Primitives: VideoMAE (Tong et al., 2022) processes 16-frame segments to learn higher-level motion patterns.

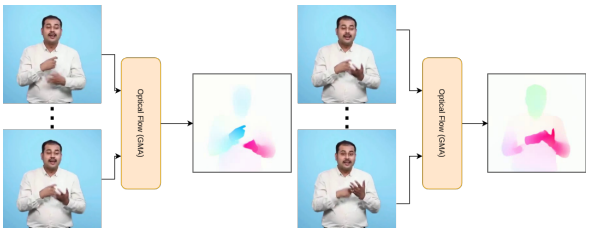

Figure 2: Optical flow map extraction using the GMA.

3.2 Cross-Modal Alignment

Spatial and motion features are projected into the textual embedding space via an alignment module (Hwang et al., 2025), consisting of a linear layer, 1D TCN, and MLP. To bridge the modality gap, we warm up this module with softmax-based contrastive learning (Radford et al., 2021; Jia et al., 2021), aligning embeddings of matching sign-text pairs while pushing apart mismatched ones. Only the alignment module is updated, preserving the LLM's language capabilities and providing well-initialized features for SLT training.

3.3 Language Modeling

Aligned visual and motion features are fed into a multilingual LLM fine-tuned with LoRA to generate the target sentence Y. To focus the LLM on

the SLT task, we employ a task-specific prompt (Hwang et al., 2025) that provides a clear instruction, e.g., "Translate the given sentence into Indian," along with multilingual reference translations (e.g., Hindi, French, Spanish) sampled from the training set. The prompt template is shown in Appendix A. Each reference is formatted as $[SRC] = [TRG]$, enabling in-context learning while preventing direct exposure to the target sentence by shuffling pairs during training. At test time, a translation pair from the training set serves as the reference.

4 Experiments

4.1 Datasets

We used the following datasets in our experiments.

German Sign Language (DGS): The RWTH-PHOENIX-2014T (Phoenix-14T) dataset (Camgoz et al., 2018) is the standard benchmark for DGS translation. It contains 7,096 training, 519 validation, and 642 test samples, each aligned with German sentences. The dataset covers weather forecast scenarios interpreted by professional sign language interpreters on television and includes a vocabulary of 2,887 words.

Indian Sign Language (ISL): iSign (Joshi et al., 2024) is a recently introduced large-scale dataset for ISL, comprising over 127k sentence-aligned signing videos collected from diverse real-world contexts.

4.2 Balanced Subset Construction from iSign (ISL)

We retain only sentences containing 5–15 words, reducing the dataset from 127k to 76k samples. Sentences shorter than 4 words are excluded, as very short translations primarily produce low BLEU-1/BLEU-2 scores. When such samples constitute a large portion of the data, the averaged BLEU-4 score no longer reflects meaningful translation quality. Similarly, extremely long sentences, i.e., those with more than 15 words (which have 310 frames on average and up to 2370 frames) are also removed, as their excessive frame counts introduce variability and noise, making model training unstable and inefficient.

From this pool, we construct a balanced subset of 10K samples (avg. 200 frames/sample) as follows: (i) **Word frequency grouping:** vocabulary is split into *rare* (<5 occurrences), *mid-frequency* (5–50), and *common* (>50). (ii) **Sample prioritization:** sentences with rare or mid-frequency words are preferentially selected to ensure coverage of underrepresented words. (iii) **Subset construction:** samples are chosen until the 10K quota is met, filling any gap with random draws. (iv) **Vocabulary coverage:** this guarantees diversity and balance, supporting more effective training.

4.3 Evaluation Metrics

To assess the quality of sign language translations, we employ standard evaluation metrics commonly used in the machine translation literature: BLEU (Papineni et al., 2002) and ROUGE-L (Lin and Och, 2004). BLEU measures n-gram precision by comparing predicted translations with ground-truth references, and we report scores from BLEU-1 through BLEU-4 using the SacreBLEU[2].

4.4 Contending Methods

We evaluate the following SLT models:

SLT (GF) (Camgoz et al., 2020): It is a transformer-based model that jointly learns sign recognition and translation in an end-to-end manner, using CTC loss for alignment. We use the GF framework of this model.

GFSLT-VLP (Zhou et al., 2023): It is a gloss-free framework that combines CLIP-based contrastive learning with masked self-supervised objectives, enabling robust cross-modal representations and strong translation without gloss annotations.

SpaMo (Hwang et al., 2025): It uses off-the-shelf visual encoders for spatial and motion features, combined with language prompts and a lightweight visual-text alignment stage before SLT supervision.

4.5 Implementation Details

We follow the architecture and training setup of SpaMo (Hwang et al., 2025) for spatial–motion feature extraction, cross-modal fusion, and language modeling. Spatial features are obtained from CLIP ViT-L/14 (Radford et al., 2021), while motion representations are enhanced with optical flow maps estimated using global motion averaging (GMA) (Jiang et al., 2021), followed by VideoMAE-L/16 (Tong et al., 2022) over 16-frame clips with a stride of 8. For the language model, we employ Flan-T5-XL (Chung et al., 2024) with LoRA adaptation, using a 1K-step warm-up on both Phoenix-14T and iSign. All experiments are conducted on a single NVIDIA A100 GPU.

[2] https://github.com/mjpost/sacrebleu

4.6 Results

We present our experimental results in Table 1, comparing three existing models and our approach on the Phoenix-14T and iSign-10k datasets.

Method	BLEU-1	BLEU-2	BLEU-3	BLEU-4	ROUGH-L
Phoenix-14T					
SLT (GF)	44.99	32.14	24.62	20.00	**45.32**
GFSLT-VLP	39.52	29.15	22.54	18.23	38.60
SpaMo	**46.41**	**33.15**	**25.22**	**20.18**	42.21
SpaMo-OF (ours)	38.06	25.26	18.58	14.72	33.52
iSign-10k					
SLT (GF)	9.84	2.87	1.33	0.73	9.08
GFSLT-VLP	9.24	4.10	2.56	1.93	9.24
SpaMo	25.42	12.90	9.13	7.35	16.23
SpaMo-OF (ours)	**27.91**	**15.00**	**10.67**	**8.58**	**18.98**

Table 1: Performance of SLT methods on the **Phoenix-14T** and **iSign-10k test sets**, reported in BLEU and ROUGE-L.

Results on Phoenix-14T: We used the preprocessed data from (Camgoz et al., 2020) to conduct experiments with SLT(GF). For the other models, we had to prepare the dataset in the appropriate format. The findings, shown in Table 1, indicate that (Hwang et al., 2025) achieves the best performance, with a BLEU-4 score of 20.18. Incorporating optical flow maps (SpaMo-OF) results in a performance drop.

Results on iSign-Full: We conducted experiments only with SLT (GF) (Camgoz et al., 2020) for this dataset. We preprocessed the dataset in a manner consistent with Phoenix-14T. Features were extracted from the signing videos using EfficientNet (Tan and Le, 2019). But this yielded poor performance (BLEU-4: 0.32), similar to the trend reported by (Joshi et al., 2024). Given that the dataset contains over 127k samples, further experimentation with other models proved impractical within our resources due to the substantial computational resources required.

Results on iSign-10k: Our experiments on the iSign-10k subset suggest that incorporating optical flow maps enables the model to leverage occluded motion cues more effectively, as captured by GMA (Jiang et al., 2021). As shown in Table 1, our method achieves a BLEU-4 score of 8.58, the highest among all evaluated approaches.

As part of our ablation study, we examined the effect of in-context examples on both datasets and observed marginal performance gains, with three in-context examples yielding the best BLEU-4 scores. We further evaluated the impact of in-context examples during test time. Detailed results are provided in Appendix B.

4.7 Qualitative Evaluation of Translation Results

The translation system achieves mixed performance with near-perfect accuracy on simple sentences with common vocabulary but struggles with sentences containing numbers and technical references. A few examples of correct and incorrect translations produced by SpaMo-OF on iSign-10k test set are shown in Table 2. More qualitative examples and detailed error cases are provided in Appendix C.

Ground Truth:	when he began to sing, the air became warm.
Generated:	when he began to sing, the air became warm.
Ground Truth:	once a farmer and his wife lived in a village with their small son.
Generated:	once a farmer and his wife lived in a village with their small son.
Ground Truth:	soldiers were paid regular salaries and maintained by the king throughout the year.
Generated:	soldiers were paid regular salaries and maintained by the king throughout the year.
Ground Truth:	do you have a pet, a cat or a dog?
Generated:	do you have a pet, a cat or a dog?
Ground Truth:	she was also a certified flight instructor. after qualifying as a pilot,
Generated:	kalpana was born in karnal, haryana.
Ground Truth:	story, mittu and the yellow mango.
Generated:	the peacock is blue and green.
Ground Truth:	many people feel bewildered by the speed of technological innovation.
Generated:	the company is aiming to become a global player in the industry.
Ground Truth:	look at figure 7.25a and b carefully.
Generated:	identify the parts of the pistol with the help of figure 7.24.

Table 2: Translation examples from the iSign-10k test set using SpaMo-OF. Blue indicates partial matches; top rows show correct outputs, while bottom rows illustrate common errors.

5 Conclusion

This work presented the first systematic evaluation of representative SLT models on Indian Sign Language, highlighting the challenges of extending methods successful in well-resourced languages to a low-resource setting. We have shown that curating a clean, balanced subset of iSign is critical for reliable evaluation and that augmenting SpaMo with optical flow features yields notable improvements, achieving a BLEU-4 score of 8.58. Our results suggest that dataset quality, rather than scale alone, is key to translation performance, and that motion-aware representations play an essential role in modeling signed communication. Future efforts should focus on constructing cleaner benchmarks and designing models that more effectively integrate spatial and motion primitives to advance robust ISL translation.

Limitations

Our work is limited by computational resources, which prevented training on the full iSign dataset (127k+ samples). We relied on a curated 10k subset for feasibility. Optical flow computation also adds overhead, limiting scalability. Additionally, the curated subset may not capture the full linguistic diversity of ISL. Future work should explore more efficient architectures and larger, more diverse datasets to improve performance and generalization.

References

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

Necati Cihan Camgoz, Simon Hadfield, Oscar Koller, Hermann Ney, and Richard Bowden. 2018. Neural Sign Language Translation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Necati Cihan Camgoz, Oscar Koller, Simon Hadfield, and Richard Bowden. 2020. Sign Language Transformers: Joint End-to-End Sign Language Recognition and Translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Yutong Chen, Ronglai Zuo, Fangyun Wei, Yu Wu, Shujie LIU, and Brian Mak. 2022. Two-Stream Network for Sign Language Recognition and Translation. In *Advances in Neural Information Processing Systems*, volume 35, pages 17043–17056. Curran Associates, Inc.

Zhigang Chen, Benjia Zhou, Jun Li, Jun Wan, Zhen Lei, Ning Jiang, Quan Lu, and Guoqing Zhao. 2024. Factorized Learning Assisted with Large Language Model for Gloss-free Sign Language Translation. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 7071–7081. ELRA and ICCL.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, and 16 others. 2024. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(70):1–53.

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 *Conference on Computer Vision and Pattern Recognition (CVPR)*.

Jia Gong, Lin Geng Foo, Yixuan He, Hossein Rahmani, and Jun Liu. 2024. LLMs are Good Sign Language Translators. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18362–18372.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Hezhen Hu, Weichao Zhao, Wengang Zhou, and Houqiang Li. 2023. SignBERT+: Hand-Model-Aware Self-Supervised Pre-Training for Sign Language Understanding. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(9):11221–11239.

Hezhen Hu, Weichao Zhao, Wengang Zhou, Yuechen Wang, and Houqiang Li. 2021. SignBERT: Pre-Training of Hand-Model-Aware Representation for Sign Language Recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 11087–11096.

Eui Jun Hwang, Sukmin Cho, Junmyeong Lee, and Jong C. Park. 2025. An Efficient Gloss-Free Sign Language Translation Using Spatial Configurations and Motion Dynamics with LLMs. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 3901–3920. Association for Computational Linguistics.

Youngjoon Jang, Haran Raajesh, Liliane Momeni, Gül Varol, and Andrew Zisserman. 2025. Lost in Translation, Found in Context: Sign Language Translation with Contextual Cues. In *Proceedings of the Computer Vision and Pattern Recognition Conference (CVPR)*, pages 8742–8752.

Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. 2021. Scaling up visual and vision-language representation learning with noisy text supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 4904–4916. PMLR.

Shihao Jiang, Dylan Campbell, Yao Lu, Hongdong Li, and Richard Hartley. 2021. Learning to estimate hidden motions with global motion aggregation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9772–9781.

Abhinav Joshi, Susmit Agrawal, and Ashutosh Modi. 2023. ISLTranslate: Dataset for translating Indian Sign Language. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages

10466–10475. Association for Computational Linguistics.

Abhinav Joshi, Romit Mohanty, Mounika Kanakanti, Andesha Mangla, Sudeep Choudhary, Monali Barbate, and Ashutosh Modi. 2024. iSign: A Benchmark for Indian Sign Language Processing. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 10827–10844. Association for Computational Linguistics.

Chin-Yew Lin and Franz Josef Och. 2004. Automatic Evaluation of Machine Translation Quality Using Longest Common Subsequence and Skip-Bigram Statistics. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-04)*, pages 605–612.

Kezhou Lin, Xiaohan Wang, Linchao Zhu, Ke Sun, Bang Zhang, and Yi Yang. 2023. Gloss-Free End-to-End Sign Language Translation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12904–12916. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR.

Baifeng Shi, Ziyang Wu, Maolin Mao, Xin Wang, and Trevor Darrell. 2025. When do we not need larger vision models? In *Computer Vision – ECCV 2024*, pages 444–462. Springer Nature Switzerland.

Bowen Shi, Diane Brentari, Gregory Shakhnarovich, and Karen Livescu. 2022. Open-Domain Sign Language Translation Learned from Online Video. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 6365–6379, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Mingxing Tan and Quoc Le. 2019. EfficientNet: Rethinking model scaling for convolutional neural networks. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 6105–6114. PMLR.

Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. 2022. Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training. In *Advances in Neural Information Processing Systems*, volume 35, pages 10078–10093. Curran Associates, Inc.

Ryan Wong, Necati Cihan Camgoz, and Richard Bowden. 2024. Sign2GPT: Leveraging Large Language Models for Gloss-Free Sign Language Translation. In *The Twelfth International Conference on Learning Representations*.

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 (ICCV)*, pages 20871–20881.

Appendix

A Prompt Template

In this section, we describe the prompt template used in our experiments for sign language translation. To facilitate multilingual in-context learning, we leverage the Google Translate API[3] to translate Indian Sign Language (ISL) sentences into multiple target languages (Hindi, Spanish, and French), enabling the model to benefit from cross-lingual cues.

Sign Video Input:	[Extracted Sign Feature]
Instruction:	Translate the given sentence into Indian.
In-context Examples:	पेड़ का रंग क्या है?
	¿Cuál es el color del árbol?
	Quelle est la couleur de l'arbre?

Table 3: Example of the prompt format used in our experiment.

B Ablation Study

Table 4 shows the impact of varying the number of in-context examples during training on the Phoenix-14T and iSign-10k datasets. We observe that increasing the number of examples leads to consistent, albeit modest, gains across all BLEU and ROUGE metrics. Interestingly, the best performance—reflected in the highest BLEU-4 and ROUGE-L scores—is achieved with three in-context examples, indicating that a small amount of contextual guidance can effectively enhance the model's ability to align signs with corresponding text.

[3] https://cloud.google.com/translate?hl=en

No. of in-context examples	BLEU-1	BLEU-2	BLEU-3	BLEU-4	ROUGE-L
Phoenix-14T					
0	35.14	22.03	15.65	12.20	29.38
1	33.52	20.68	14.41	11.02	28.93
2	35.79	22.99	16.51	12.91	31.31
3	**37.45**	**24.37**	**18.01**	**14.41**	**32.63**
iSign-10k					
0	26.61	14.67	10.67	8.73	17.75
1	26.22	13.79	9.89	8.08	16.93
2	26.26	13.98	10.06	8.23	16.94
3	**27.95**	**15.37**	**11.03**	**8.92**	**18.99**

Table 4: Performance with varying numbers of in-context examples during training (all models tested with zero in-context examples).

Table 5 examines the effect of in-context examples during testing, with all models trained using three examples. For Phoenix-14T (German), the in-context examples are in English, Spanish, and French, while for iSign-10k (ISL), they are in Hindi, Spanish, and French. The results indicate that providing a small number of in-context examples can slightly improve performance. Notably, for iSign-10k, including Hindi examples at test time appears to enhance translation quality, suggesting that using a language closely related to the target output can help the model better generalize, whereas adding more examples beyond two does not consistently yield further gains.

No. of in-context examples	BLEU-1	BLEU-2	BLEU-3	BLEU-4	ROUGE-L
Phoenix-14T					
0	35.70	22.87	16.63	13.15	31.34
1	**37.63**	**24.60**	**18.10**	14.37	**32.82**
2	37.30	24.22	17.77	14.18	32.37
3	37.45	24.37	18.01	**14.41**	32.63
iSign-10k					
0	27.91	15.00	10.67	8.58	18.98
1	28.11	15.51	11.22	**9.12**	**19.07**
2	**28.14**	**15.62**	**11.25**	9.09	18.93
3	27.95	15.37	11.03	8.92	18.99

Table 5: Performance with varying numbers of in-context examples during testing (all models trained with 3 in-context examples).

C Qualitative Evaluation of Translation Results

Tables 6, 7, and 8 provide additional translation examples drawn from iSign-10k test set, with generated sentences produced using SpaMo-OF. The tables contain both high-quality and poor-quality system outputs. Blue highlighting indicates exact word matches between ground truth and generated text, revealing the system's binary performance pattern with either extensive lexical preservation or complete semantic divergence.

Ground Truth:	why did the farmer's wife strike the mongoose with her basket?
Generated:	why did the farmer's wife strike the mongos with her basket?
Ground Truth:	saida's mother had been ailing for a long time. fever, cough.
Generated:	saida's mother had been ailing for a long time. fever, cough.
Ground Truth:	what did raja nanda do to anger the court of gautamiputra?
Generated:	prashastis and what they tell us?
Ground Truth:	in the city of madurai, there was a craftsperson named chandragupta.
Generated:	between 2,200 and 1900 years ago. between 2,200 and 1900 years ago.
Ground Truth:	taxes on crops were the most important. this was because most people were farmers.
Generated:	usually the tax was fixed at 1/6th of what was produced.
Ground Truth:	connect the dots to write a circle.
Generated:	say which one will float and which one will sink.
Ground Truth:	the mauryan rulers did not try to control these changes.
Generated:	9 years after becoming king, I conquered.
Ground Truth:	when people began writing on cloth.
Generated:	trace the river indus and its tributaries in the map.
Ground Truth:	including craftspersons and merchants. in addition,
Generated:	for instance, inscriptions from matura mention goldsmiths.
Ground Truth:	a poor shepherd once lived in iran. though uneducated,
Generated:	a poor shepherd once lived in iran. he became famous for his piousness.

Table 6: Additional translation examples showing mixed quality patterns including near-matches, entity disambiguation failures, instruction drift, and complete semantic divergence. Blue highlights exact word matches between ground truth and generated text.

Ground Truth:	he was ordered to explain why he always carried an iron chest.
Generated:	he was ordered to explain why he always carried an iron chest.
Ground Truth:	but an old blanket was all that came out.
Generated:	but an old blanket was all that came out.
Ground Truth:	now the synopsis of the next part of the story.
Generated:	now the synopsis of the next part of the story.
Ground Truth:	which had to flee from its post. the rays got through.
Generated:	which had to flee from its post. the rays got through.
Ground Truth:	day by day, i float my paper boats.
Generated:	day by day, i float my paper boats.
Ground Truth:	we apologize for any inconvenience caused during the repairs.
Generated:	we apologize for any inconvenience caused by the late arrival of train.
Ground Truth:	the little group of travellers scattered in fear but swami haridas called them together.
Generated:	the little group of travellers scattered in fear but swami haridas called them together.
Ground Truth:	there once lived a bird and two newborn babies blue in a forest.
Generated:	there once lived a bird and two newborn babies in a forest.
Ground Truth:	question number five. did she repent her hasty action?
Generated:	question number five. did she repent her hasty action?
Ground Truth:	frankly, i don't like to leave the child alone with the mongoose.
Generated:	i don't like to leave the child alone with the mongoose.

Table 7: Translation examples demonstrating high-quality outputs with perfect or near-perfect semantic preservation, minor lexical variations, and contextual consistency. Blue indicates exact word-level matches between ground truth and generated text.

Content Type	Example	Ground Truth	Generated
		High Performance: Narrative & Simple Sentences	
Narrative	Story introduction	once a farmer and his wife lived in a village with their small son.	once a farmer and his wife lived in a village with their small son.
Narrative	Story continuity	there once lived a bird and two newborn babies in a forest.	there once lived a bird and two newborn babies in a forest.
Historical	Factual statement	soldiers were paid regular salaries and maintained by the king throughout the year.	soldiers were paid regular salaries and maintained by the king throughout the year.
Question	Direct question	do you have a pet, a cat or a dog?	do you have a pet, a cat or a dog?
		Low Performance: Educational & Instructional Content	
Instruction	Drawing activity	connect the dots to write a circle.	say which one will float and which one will sink.
Technical Ref.	Figure reference	look at figure 7.25a and b carefully.	identify the parts of the pistol with the help of figure 7.24.
Historical Ed.	Context instruction	when people began writing on cloth.	trace the river indus and its tributaries in the map.
Entity Ref.	Historical query	what did raja nanda do to anger the court of gautamiputra?	prashastis and what they tell us?
Educational	Historical context	in the city of madurai, there was a craftsperson named chandragupta.	between 2,200 and 1900 years ago. between 2,200 and 1900 years ago.
Biographical	Career context	she was also a certified flight instructor. after qualifying as a pilot,	kalpana was born in karnal, haryana.

Table 8: Translation performance across content types in iSign-10k test set using SpaMo-OF. Blue indicates exact matches. Top section shows perfect translations on narrative and simple content while bottom section reveals failures on relatively more complex content.

Pose-Based Temporal Convolutional Networks for Isolated Indian Sign Language Word Recognition

Tatigunta Bhavi Teja Reddy , Vidhya Kamakshi
Department of Computer Science & Engineering
National Institute of Technology Calicut
Kozhikode - 673601, Kerala, India
tatigunta_m240602cs@nitc.ac.in, vidhyakamakshi@nitc.ac.in

Abstract

This paper presents a lightweight and efficient baseline for isolated Indian Sign Language (ISL) word recognition developed for the WSLP-AACL-2025 Shared Task.We propose a two-stage framework combining skeletal landmark extraction via MediaPipe Holistic with a Temporal Convolutional Network (TCN) for temporal sequence classification. The system processes pose-based input sequences instead of raw video, significantly reducing computation and memory costs. Trained on the WSLP-AACL-2025 dataset containing 4,398 isolated sign videos across 4,361 word classes, our model achieves 54% top-1 and 78% top-5 accuracy.

1 Introduction

Sign Language serves as a primary means of communication for millions of deaf and hard-of-hearing individuals across the world. However, the lack of mutual intelligibility between signers and non-signers continues to pose substantial barriers in education, healthcare, employment, and daily communication. While human interpreters provide an effective bridge, their limited availability and high cost restrict widespread accessibility. Automated Sign Language Recognition (SLR) systems thus hold significant potential to enhance social inclusion by enabling real-time, scalable translation between sign and spoken languages.

Recent progress in computer vision and deep learning has revitalized research in automatic sign language understanding. Unlike spoken languages, which rely on one-dimensional acoustic signals, sign languages are inherently multimodal—integrating hand configurations, body posture, facial expressions, and spatial-temporal dynamics to convey meaning. This multidimensional structure makes SLR a particularly challenging problem in visual sequence modeling. Conventional frame-based models often struggle to capture the fine-grained temporal dependencies and spatial variations inherent to signing. Consequently, developing models that effectively learn temporal patterns, remain robust to inter-signer variability, and generalize across diverse signing conditions is a key research objective.

Despite these advances, Sign Language Recognition remains a challenging task due to its inherently temporal and highly variable nature. Each sign involves dynamic motion sequences that differ across signers in speed, articulation, and regional style, while transitions between signs often blur semantic boundaries. Moreover, annotated datasets for Indian Sign Language (ISL) are limited in size and diversity, constraining the training of data-intensive deep models. These challenges call for lightweight architectures capable of capturing long-range temporal dependencies using compact representations.

Motivated by these challenges this study, we address the problem of isolated Indian Sign Language (ISL) word recognition as part of the WSLP-AACL-2025 Shared Task. The goal is to design and train an efficient recognition pipeline that performs reliably despite the limited availability of labeled samples per class. Our work explores a lightweight, pose-based approach using MediaPipe Holistic for landmark extraction and Temporal Convolutional Networks (TCNs) for temporal modeling, aiming to balance recognition accuracy, computational efficiency, and real-time deployability on assistive devices.

2 Related Work

Research in Sign Language Recognition (SLR) has evolved from handcrafted visual features to deep neural architectures capable of modeling complex spatio-temporal dynamics. Early vision-based approaches (Tamura and Kawasaki, 1988) relied on geometric and motion descriptors, often

Proceedings of the Workshop on Sign Language Processing (WSLP), pages 51–54

combined with Hidden Markov Models (HMMs) (Starner and Pentland, 1995; Starner et al., 1998) for real-time American Sign Language (ASL) recognition. Subsequent works enhanced temporal modeling through parallel HMMs to mitigate co-articulation effects (Vogler and Metaxas, 1999), and through hybrid CNN–HMM architectures for continuous signing (Koller et al., 2015).

With the rise of deep learning, pose-based representations have gained prominence for their robustness and computational efficiency. MediaPipe Holistic (Lugaresi et al., 2019) enabled real-time extraction of body and hand landmarks, facilitating lightweight recognition pipelines. Leveraging such pose data, transformer-based models have demonstrated strong performance for isolated sign recognition (Alyami et al., 2024), while recurrent GRU-based architectures have been successfully applied to Indian Sign Language (ISL) recognition (Subramanian et al., 2022). More recent studies explore Temporal Convolutional Networks (TCNs) with dilated causal convolutions for efficient temporal reasoning (Xu et al., 2023), and correlation networks enhanced with spatial-temporal attention for continuous SLR (Hu et al., 2023).

Reviewing the literature reflects a paradigm shift toward pose-based and temporally aware architectures that balance recognition accuracy with real-time deployability, forming the foundation for the approach adopted in this work.

3 Methodology

The proposed Indian Sign Language (ISL) word recognition system adopts a two-stage framework integrating pose-based feature extraction with temporal modeling. In the first stage, skeletal landmarks are extracted from each video frame using MediaPipe Holistic (Lugaresi et al., 2019). This pipeline provides 33 pose landmarks and 21 landmarks per hand, resulting in 75 keypoints per frame, each with (x, y, z) coordinates, yielding a 225-dimensional feature vector. This representation retains essential kinematic information while substantially reducing input dimensionality compared to raw RGB frames.

In the second stage, the extracted pose sequences are processed by a Temporal Convolutional Network (TCN) designed to capture temporal dependencies across sign sequences. Unlike recurrent networks, TCNs leverage 1D convolutions

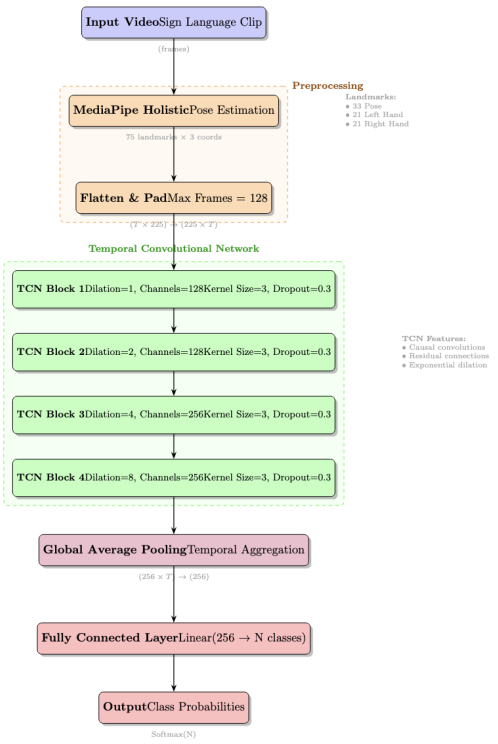

Figure 1: Overview of the proposed Pose-based TCN pipeline for ISL Word Recognition.

along the temporal axis, enabling full parallelization during training and inference. The network maps normalized and padded pose sequences directly to sign word labels in an end-to-end manner, achieving a balance between recognition accuracy and computational efficiency suitable for real-time applications.

The overall model design emphasizes three principles: (i) robustness to intra-class and inter-signer variations, (ii) effective temporal modeling through dilated causal convolutions, and (iii) lightweight representation enabling deployment on resource-limited assistive devices. The model architecture is illustrated in Figure 1. Each temporal block consists of two dilated Conv1D layers followed by causal chomp, ReLU activation, and dropout, with residual connections to stabilize gradient flow. The network input is a tensor of shape $(T \times 225)$, where $T = 128$ is the temporal length. Four temporal blocks with exponentially increasing dilation rates are stacked, followed by global average pooling along the temporal dimension and a fully connected layer for classification across N sign classes.

3.1 Temporal Convolutional Network Architecture

Temporal Convolutional Networks (TCNs) serve as a parallelizable alternative to recurrent architectures for sequence modeling. A TCN operates using 1D convolutions over time, where causal convolutions ensure that each timestep prediction depends only on the current and past frames. Dilated convolutions enlarge the receptive field exponentially with minimal parameter overhead, allowing efficient long-range temporal modeling. Residual connections are incorporated to mitigate vanishing gradient problems and facilitate deeper network training. This architecture preserves temporal causality while providing high throughput suitable for real-time recognition.

3.2 Dataset and Preprocessing Pipeline

We employ the WSLP-AACL-2025 Shared Task Word Recognition dataset (Lab, 2025), consisting of 4,398 short video clips of isolated sign language words performed by a single signer in controlled and semi-controlled settings. The dataset spans 4,361 unique word classes, forming an extreme few-shot learning scenario: 80% of classes contain two or fewer samples, the median sample count per class is one, and the maximum is five. Videos range from 2–5 seconds at 30 FPS, with resolutions between 320p and 1080p. Following integrity checks, the dataset is divided into 3,517 training and 879 validation samples using a fixed random seed for reproducibility.

Pose extraction is performed using MediaPipe Holistic configured in non-static mode with detection and tracking confidences set to 0.3. For each frame, 75 landmarks with (x, y, z) coordinates are extracted and normalized relative to the frame dimensions and depth. Missing landmarks are replaced with zeros. Frames are resized to a width of 320 pixels, every third frame is skipped to reduce redundancy, and each sequence is truncated or padded to 128 frames. The resulting pose tensors of shape $(128 \times 75 \times 3)$ are stored in NPZ format for training.

3.3 Implementation Details

The TCN comprises four temporal blocks with hidden channel sizes [128, 128, 256, 256], kernel size 3, and dilation rates [1, 2, 4, 8]. A dropout rate of 0.3 is applied within each block. Training uses the AdamW optimizer with learning rate 10^{-3}, weight decay 0.01, $\beta = (0.9, 0.999)$, and $\epsilon = 10^{-8}$. The learning rate is adaptively reduced using a plateau scheduler (factor 0.5, patience 3, minimum learning rate 10^{-6}). Early stopping based on validation accuracy prevents overfitting.

The objective function is the categorical cross-entropy loss, defined as:

$$\mathcal{L} = -\sum_{k=1}^{N} y_k \cdot \log(\hat{y_k}) \tag{1}$$

where y_k denotes the one-hot encoded ground truth and $\hat{y_k}$ represents the predicted probability corresponding to the k^{th} sign.

This configuration achieves a balance between temporal modeling capacity, generalization on few-shot classes, and computational efficiency suitable for shared-task benchmarking and real-time deployment.

4 Results

The proposed pose-based Temporal Convolutional Network achieved a top-1 classification accuracy of 54% on the validation set. Corresponding precision, recall, and F1-scores were observed to lie consistently within the 52–54% range, indicating balanced performance across most sign classes despite the highly imbalanced few-shot nature of the dataset.

A Top-5 accuracy of approximately 78% further demonstrates that the correct class frequently appeared among the top predicted candidates, highlighting the model's capacity to capture semantically relevant temporal patterns even when the top prediction was incorrect.

An examination of prediction confidence distributions revealed that correctly classified samples exhibited moderate confidence levels, whereas lower confidence was typically associated with visually or temporally ambiguous gestures, signer variation, or partial landmark occlusions. These findings suggest that while the model effectively learns generalizable temporal representations from pose trajectories, performance remains constrained by limited per-class data and subtle intra-class motion variations.

5 Summary

This work presents an efficient baseline for isolated Indian Sign Language (ISL) recognition by integrating MediaPipe-based pose estimation with

Temporal Convolutional Networks (TCNs). The proposed system achieved 54% validation accuracy on a challenging few-shot multi-class dataset, highlighting the effectiveness of skeleton-based representations in capturing essential gesture dynamics. The TCN architecture, leveraging dilated causal convolutions, successfully modeled long-range temporal dependencies while retaining computational efficiency through its fully parallelizable design. Preprocessing strategies such as frame skipping and resolution reduction reduced computational cost by nearly 70% with minimal performance degradation, demonstrating the approach's suitability for real-time deployment. Representing each frame through 75 key landmarks achieved an input size reduction of approximately 4,000× relative to raw video frames, significantly enhancing inference speed without compromising discriminative power. The proposed pipeline establishes a compact and practical foundation for the recognition of ISL in low-resource and assistive environments. Future work can extend this baseline by exploring richer temporal attention mechanisms, synthetic data augmentation, and integration of facial and contextual cues to further enhance recognition accuracy and system robustness.

References

Sultan Alyami, Hamzah Luqman, and Mohammad Hammoudeh. 2024. Isolated arabic sign language recognition using a transformer-based model and landmark keypoints. *ACM Transactions on Asian and Low-Resource Language Information Processing*, 23(1):1–19.

Lianyu Hu, Liqing Gao, Zekang Liu, and Wei Feng. 2023. Continuous sign language recognition with correlation network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2529–2539.

Oscar Koller, Hermann Ney, and Richard Bowden. 2015. Deep sign: Hybrid cnn-hmm for continuous sign language recognition. *British Machine Vision Conference (BMVC)*.

Exploration Lab. 2025. Wslp-aacl-2025 shared task word recognition dataset. https://huggingface.co/datasets/Exploration-Lab/WSLP-AACL-2025/tree/main/Shared_task_WR.

Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, and 1 others. 2019. Mediapipe: A framework for building perception pipelines. In *arXiv preprint arXiv:1906.08172*.

Thad Starner and Alex Pentland. 1995. Visual recognition of american sign language using hidden markov models. In *International Workshop on Automatic Face and Gesture Recognition*, pages 189–194. IEEE.

Thad Starner, Joshua Weaver, and Alex Pentland. 1998. Real-time american sign language recognition using desk and wearable computer based video. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(12):1371–1375.

Balamurali Subramanian, Bekhzod Olimov, Sushil M Naik, and 1 others. 2022. An integrated mediapipe-optimized gru model for indian sign language recognition. *Scientific Reports*, 12(1):11964.

Shingo Tamura and Satoru Kawasaki. 1988. Recognition of sign language motion images. *Pattern Recognition*, 21(4):343–353.

Christian Vogler and Dimitris Metaxas. 1999. Parallel hidden markov models for american sign language recognition. In *Proceedings of the Seventh IEEE International Conference on Computer Vision*, volume 1, pages 116–122. IEEE.

Xiujuan Xu, Jian Wang, and Lei Zhang. 2023. Isolated word sign language recognition based on improved skresnet-tcn network. *Journal of Sensors*, 2023:9503961.

Cross-Linguistic Phonological Similarity Analysis in Sign Languages Using HamNoSys

Abhishek Bharadwaj Varanasi, Manjira Sinha, Tirthankar Dasgupta
TCS Research
[1]{varanasi.abhishek,sinha.manjira,dasgupta.tirthankar}@tcs.com

Abstract

This paper presents a cross-linguistic analysis of phonological similarity in sign languages using symbolic representations from the Hamburg Notation System (HamNoSys). We construct a dataset of 1000 signs each from British Sign Language (BSL), German Sign Language (DGS), French Sign Language (LSF), and Greek Sign Language (GSL), and compute pairwise phonological similarity using normalized edit distance over HamNoSys strings. Our analysis reveals both universal and language-specific patterns in handshape usage, movement dynamics, non-manual features, and spatial articulation. We explore intra and inter-language similarity distributions, phonological clustering, and co-occurrence structures across feature types. The findings offer insights into the structural organization of sign language phonology and highlight typological variation shaped by linguistic and cultural factors.

1 Introduction

Sign languages (SLs) are complex visual-gestural languages that convey meaning through a combination of hand configurations, movements, orientations, and spatial locations (Sinha, 2009). Unlike spoken languages, sign languages lack a standardized written form (Langer et al., 2014), making computational analysis and cross-linguistic comparison particularly challenging. One of the foundational aspects of sign language linguistics is phonology-the study of minimal visual units that distinguish signs. Phonological modeling in sign languages has been a growing area of interest in computational linguistics and sign language processing. Early work focused on rule-based systems and handcrafted features to capture phonological components such as handshape, location, and movement (Stokoe, 1960; Brentari, 1998). These approaches laid the foundation for formal linguistic analysis but lacked scalability and cross-linguistic generalization.

This paper addresses the problem of identifying signs that are phonologically similar within and across multiple sign languages. Specifically, we focus on four major sign languages: British Sign Language (BSL), German Sign Language (DGS), French Sign Language (LSF), and Greek Sign Language (GSL). For each language, we construct a dataset of 1000 signs, each annotated with its phonological structure using the Hamburg Notation System (HamNoSys) (Prillwitz and für Deutsche Gebärdensprache und Kommunikation Gehörloser, 1989). We compute pairwise phonological similarity between all sign pairs using a normalized edit distance over their HamNoSys representations, resulting in a 1000×1000 similarity matrix per language.

Unlike prior work that focuses on building computational models for sign recognition or translation (Cihan Camgoz et al., 2017; Camgoz et al., 2018; Stoll et al., 2020; Saunders et al., 2020; Chen et al., 2022), our objective is to perform a detailed analytical study of phonological similarity patterns. We explore intra-language and inter-language similarity distributions, identify phonological clusters, and investigate the structural properties of the resulting similarity matrices. Our findings offer insights into the phonological organization of signs and provide a foundation for future work in multilingual sign language processing.

1.1 Overview of HamNoSys

The Hamburg Notation System (HamNoSys) (Prillwitz and für Deutsche Gebärdensprache und Kommunikation Gehörloser,

Proceedings of the Workshop on Sign Language Processing (WSLP), pages 55–70

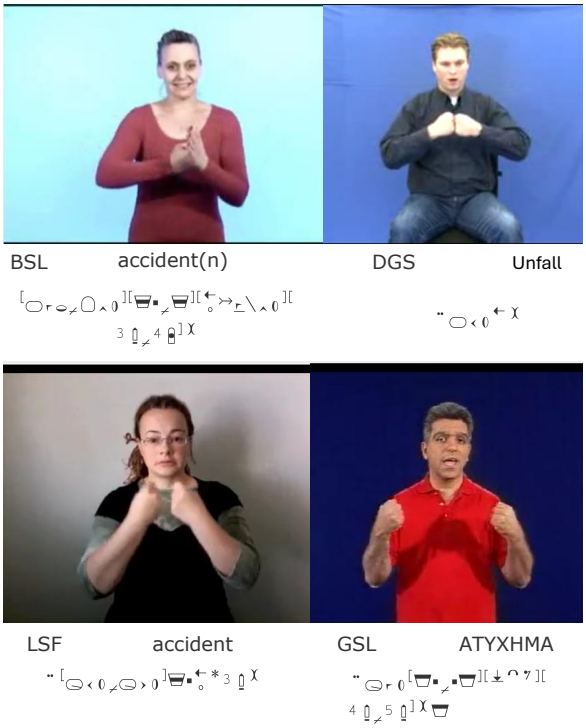

Figure 1: Phonological representation (using HamNoSys) of the word "ACCEIDENT" across different languages.

1989) has emerged as a powerful tool for representing sign language phonology in a language-independent manner. It has been used in various applications, including sign synthesis (Hanke, 2004), avatar animation (Efthimiou et al., 2009), and sign language corpora annotation (Crasborn and Zwitserlood, 2008). It encodes the phonological structure of signs using a linear sequence of symbols that describe the following key features (See figure 2 for sample Hamnosys based phonological features):

Handshape: The configuration of the fingers and palm (e.g., `FlatOpen`, `Fist`, `Claw`).

Location: The spatial region of the body where the sign is articulated (e.g., `Chest`, `Forehead`, `NeutralSpace`).

Orientation: The direction the palm and fingers face during the sign (e.g., `Inward`, `Outward`, `Downward`).

Movement: The trajectory, type, and repetition of motion (e.g., `UpDown`, `Circle`, `Sideways`).

Apart from these, there are non-manual features representing facial expressions, head and body posture, and eye gaze. Each sign is represented as a structured string of HamNoSys symbols, allowing for symbolic comparison and computational processing. For example figure 1 depicts the sign representation for the word "Accident". Note that every language has its own phonological patterns of representing the same concept. Also, see Appendix A for explanation of each HamNoSys symbols.

Although these signs differ only in the movement component, such a variation can lead to a different meaning. HamNoSys enables the isolation and comparison of these phonological components, making it a powerful tool for cross-linguistic phonological analysis.

In this study, we leverage HamNoSys to compute phonological similarity between signs using a normalized edit distance metric. This approach allows us to quantify how similar two signs are based on their symbolic phonological structure, independent of signer-specific or visual noise.

2 Related Work

Recent studies have explored the use of HamNoSys for computational tasks. For example, Morrissey (2008) used HamNoSys and its SiGML encoding as the intermediate representation in a spoken-to-sign language MT pipeline, while Efthimiou et al. (2010) leveraged it for multilingual sign language resources. Sugandhi et al. (2020) proposed a HamNoSys-based avatar generation approach for text-to-ISL translation. Several other efforts have continued this line of research: Neves et al. (2020) developed a conversion toolkit from HamNoSys to SiGML to support avatar animation; Walsh et al. (2022) introduced transformer baselines for directly translating spoken language text to HamNoSys sequences, demonstrating advantages over gloss-only representations; and Bhagwat et al. (2024) presented a Marathi↔ISL translation pipeline adopting HamNoSys as an intermediate phonetic layer for synthesis. Foundational descriptions such as Hanke (2004) further highlight HamNoSys as a machine-readable phonetic notation beneficial for MT and sign avatar generation.

In the domain of sign similarity, Ormel et al. (2010) proposed methods for measur-

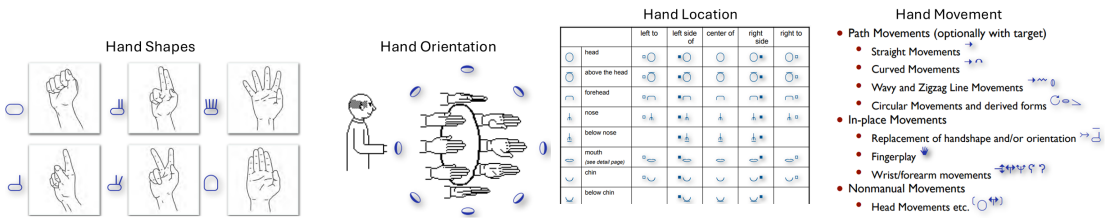

Figure 2: Examples of HamNoSys for hand shape, orientation, location and movement (Hanke, 2010)

ing phonological distance using feature-based representations, but their work was limited to small datasets and single-language settings. More recent work has explored neural models for sign similarity (Camgoz et al., 2020), though these approaches often rely primarily on visual features without explicit phonological grounding. To address this gap, Williams et al. (2017) operationalized phonological similarity by quantifying shared manual parameters, demonstrating psycholinguistic correlates of such similarity measures. Further advances have integrated phonological structure into neural models: Tavella et al. (2022) introduced the WLASL-LEX dataset annotated with phonological properties and showed that graph-based neural networks can recognize phonological features at scale; Rodriguez et al. (2023) proposed a phonological distance metric ("phdist") over fourteen phonological specifications in NGT and used it to analyze deep sign embeddings; and Kezar et al. (2023) demonstrated that incorporating phonological representations improves isolated sign recognition performance on the Sem-Lex benchmark. These works highlight increased attention toward phonologically grounded representations in computational modelling of sign similarity.

Our work differs in that it focuses on symbolic phonological similarity across multiple sign languages using HamNoSys. By constructing large-scale similarity matrices and performing analytical studies, we aim to uncover structural patterns in sign language phonology that are both linguistically meaningful and computationally tractable.

3 Dataset Construction

The dataset used in this study is derived from the publicly available **Dicta-Sign Language Resources** (Efthimiou et al., 2012), a multilingual repository of sign language data developed as part of the Dicta-Sign project. The resource provides a curated list of over 1000 concepts, each annotated with corresponding signs and phonological representations in four European sign languages: British Sign Language (BSL), German Sign Language (DGS), French Sign Language (LSF), and Greek Sign Language (GSL).

For each of the four languages, we selected 1000 signs corresponding to a shared set of concepts. Each sign is associated with a HamNoSys transcription that encodes its phonological structure, including handshape, location, orientation, and movement. These symbolic representations serve as the foundation for computing phonological similarity.

To quantify phonological similarity, we compute the normalized Levenshtein distance (Yujian and Bo, 2007) between HamNoSys strings. Given two signs i and j with HamNoSys representations H_i and H_j, the similarity score S_{ij} is defined as:

$$S_{ij} = 1 - \frac{d_{\text{lev}}(H_i, H_j)}{\max(|H_i|, |H_j|)} \quad (1)$$

where d_{lev} denotes the Levenshtein edit distance between the two strings, and $|H_i|$ is the length of the string. This results in a similarity score in the range $[0, 1]$, where 1 indicates identical phonological structure.

For each language, we construct a 1000×1000 similarity matrix $\mathbf{S}^{(l)}$ capturing all pairwise phonological similarities. These matrices form the basis for the analytical tasks described in the next section.

4 Analysis and Results

We present a comprehensive analysis of phonological similarity patterns within and across four sign languages: British Sign Language (BSL), German Sign Language (DGS), French Sign Language (LSF), and Greek Sign Language (GSL). Each language's dataset consists

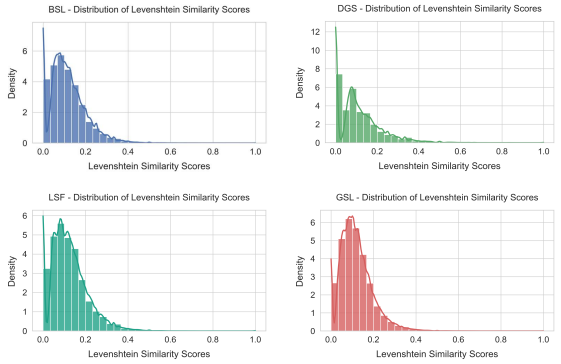

Figure 3: Intra-language phonological similarity distributions for BSL, DGS, LSF, and GSL.

of 1000 signs, and a 1000×1000 similarity matrix was computed using normalized edit distance over HamNoSys representations.

4.1 Intra-Language Similarity Distributions

Figure 3 shows the distribution of similarity scores within each language. All distributions are left-skewed, indicating that most sign pairs are moderately dissimilar, with a smaller proportion of highly similar signs. Notably, DGS and LSF exhibit slightly higher concentrations of high-similarity pairs, suggesting more phonologically compact lexicons.

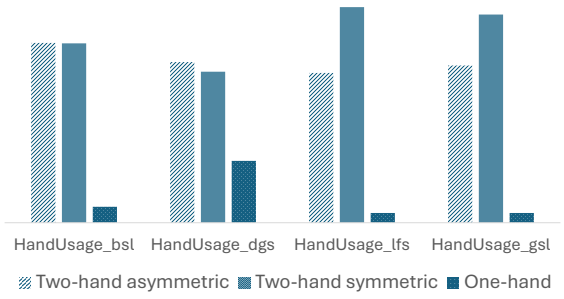

Figure 4: Intra-language hand-usage frequency for BSL, DGS, LSF, and GSL.

4.2 Phonological Clustering

To explore the internal structure of each language's phonological space, we applied hierarchical clustering on each similarity matrix. Figure 5 show the resulting clusters (only for the sake of clear visualization, we show the clustering results on a 100×100 subset). Clear block structures emerge, indicating the presence of phonological families—groups of signs that share similar handshapes, locations, or movements.

Language	Mean	Std Dev	Min	Max
BSL	0.115	0.084	0.000	1.000
DGS	0.112	0.106	0.000	1.000
LSF	0.122	0.085	0.000	1.000
GSL	0.118	0.075	0.000	1.000

Table 1: Summary statistics of phonological similarity scores

Table 1 present the mean, standard deviation, and range of similarity scores for each language. LSF and GSL show the highest average similarity, while DGS exhibits the widest spread, indicating greater phonological diversity.

4.3 One-hand vs Two-hand Sign Analysis

The Figure 4 presents the distribution of signs based on hand usage—categorized into one-handed signs, two-handed symmetric signs, and two-handed asymmetric signs—across British Sign Language (BSL), German Sign Language (DGS), French Sign Language (LSF), and Greek Sign Language (GSL). A clear trend emerges: all four languages predominantly use two-handed signs, with symmetric and asymmetric configurations being nearly equally represented. For instance, BSL shows a near-even split between symmetric (461) and asymmetric (462) two-handed signs, while LSF and GSL lean slightly toward symmetric usage. In contrast, one-handed signs are significantly less frequent, especially in LSF and GSL (only 25 each), whereas DGS shows a relatively higher count (159), suggesting a greater preference or flexibility for one-handed articulation in German Sign Language. This distribution highlights both universal tendencies and language-specific variations in sign formation, which may reflect linguistic, cultural, or ergonomic factors influencing sign language structure.

4.4 Phonological analysis across language

Table 2 showing the top 5 handshapes, movements, non-manual signs, and sign locations across British Sign Language (BSL), German Sign Language (DGS), French Sign Language

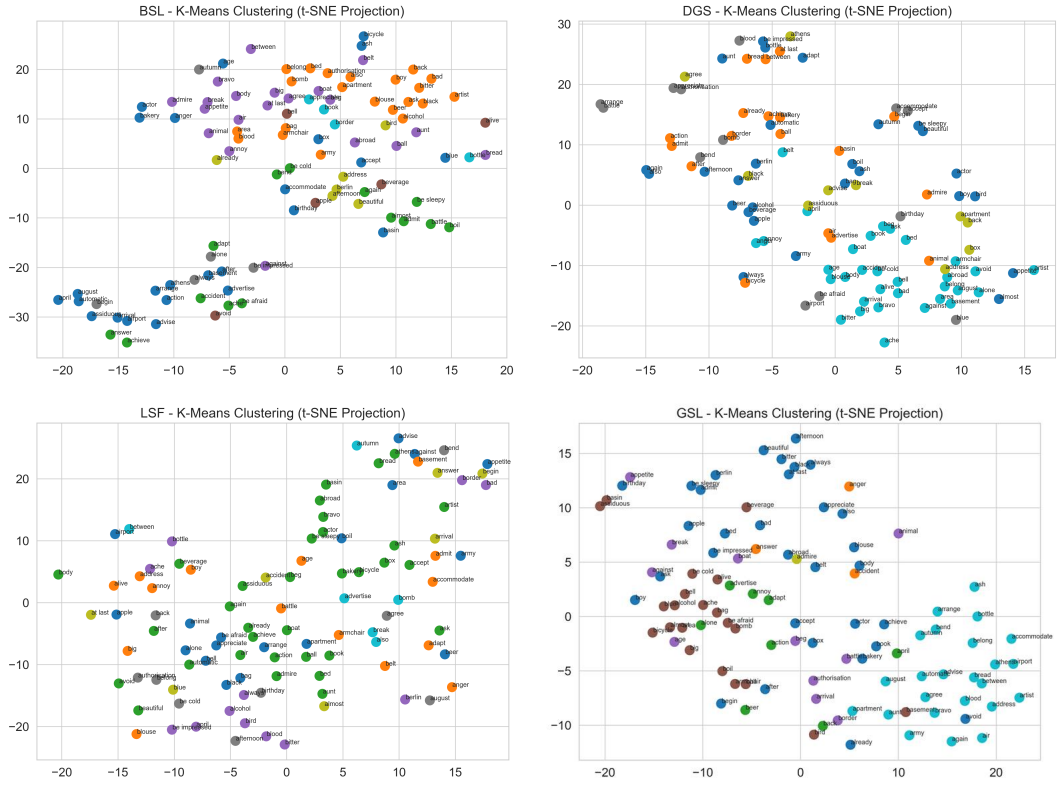

Figure 5: Phonological similarity using k-means clustering across languages. Same colored points belong to same cluster.

(LSF), and Greek Sign Language (GSL). The data reveals both shared and language-specific preferences in phonological features across the four sign languages.

Handshapes (HS): The handshapes hamthumboutmod, hamflathand, and hamfinger2 appear consistently across all languages with hamthumbacrossmod appearing in BSL, LSF, and GSL, indicating a core set of frequently used configurations. BSL and LSF favor hamthumboutmod most prominently. DGS and GSL show a high preference for hamflathand. DGS uniquely includes hamfist in its top 5, suggesting a more frequent use of closed hand configurations.

Movements (MOV): Universal dominance: hamrepeatfromstart is the most frequent movement across all four languages, highlighting repetition as a common linguistic strategy. hammoved and hammoveo are consistently present, but their ranks vary. GSL shows a higher frequency of hamfast, possibly reflecting a faster signing tempo or stylistic variation. LSF and GSL include hamrepeatfromstartseveral, suggesting more complex repetition patterns.

Non-Manual Features (NMA): BSL, LSF, and GSL emphasize hamshoulders and hamchest, indicating upper torso involvement. Moreover, GSL shows the highest counts for hamchest and hamshoulders, suggesting strong reliance on torso-based non-manual cues. DGS has lower counts overall and includes hamchin and hamhead, pointing to more facial involvement.

Sign Locations (LOC): hambetween (likely referring to the space between hands or between signer and viewer) is dominant in BSL and GSL, suggesting spatial articulation is central. DGS and LSF favor hampalml and hampalmd, indicating signs are often articulated near the palm or lower body. hamsymmlr and hamextfingeru appear across multiple languages, reflecting symmetrical and extended finger placements.

These patterns suggest that while there is a shared phonological core across sign languages—especially in handshapes and movements suggesting inter-language phonological similarity—each language exhibits

unique tendencies in non-manual features and spatial articulation. This supports the idea that sign languages, though visually grounded, are shaped by distinct linguistic and cultural norms.

Category	Features	BSL	DGS	LSF	GSL
HS	hamthumboutmod	666	216	543	646
	hamthumbacrossmod	464	0	490	631
	hamflathand	448	343	350	656
	hamfinger2	345	219	326	445
	hamfinger2345	298	0	0	401
	hamfist	0	208	0	0
	hamfingerstraightmod	0	269	289	0
MOV	hamrepeatfromstart	232	298	260	232
	hammoved	210	211	187	216
	hammoveo	134	105	115	165
	hammover	134	103	115	0
	hammoveu	109	0	0	0
	hammoveor	0	81	0	0
	repeatfromstartseveral	0	0	125	160
NMA	hamshoulders	341	91	376	456
	hamchest	259	81	377	495
	hamshouldertop	99	0	115	135
	hamneck	75	0	0	0
	hamstomach	66	0	48	75
	hamchin	0	58	0	0
	hamhead	0	27	0	0
	hamlips	0	25	56	49
LOC	hambetween	903	0	318	937
	hampalml	543	401	494	543
	hampalmd	509	317	498	491
	hamextfingeru	462	0	448	413
	hamsymmlr	375	330	320	552
	hamextfingerol	0	336	0	0

Table 2: Frequency distribution of the most frequent sign language phonological features across languages.

4.5 Intra-Phonological Co-occurrences

Insights derived from the co-occurrence (pointwise mutual information, PMI) table across four sign languages—BSL, DGS, LSF, GSL—focusing on phonological feature interactions revels that Across all languages, high PMI values are observed between compound or modified handshapes (e.g., hamthumboutmod, hamceeopen, hamfingerside), indicating that these handshapes frequently co-occur in signs with complex articulatory configurations. DGS shows strong co-occurrence between hamfingerpad and hamthumbball (PMI = 5.59±0.035), suggesting a preference for precision grip-like configurations. LSF and GSL both show high PMI between hamceeopen and hamfingerside, indicating a shared structural tendency toward open, lateral hand articulations. GSL also exhibits strong co-occurrence between hamceeopen and hamfingernail, hinting at a visual emphasis on finger extension and orientation.

The highest PMI values in hand-location category are found in LSF (hamextfingerdi-hamextfingeri, PMI = 6.91±0.043) and DGS (hamarmextended-hamextfingerdi, PMI = 5.91±0.037), suggesting frequent use of extended arm and finger configurations in spatial articulation. GSL shows strong co-occurrence between hamarmextended and hamextfingerir (PMI = 5.59±0.035), indicating a preference for distal articulation zones. Across all languages, combinations involving hamhandback, hamwristback, and hamextfinger variants suggest a consistent use of backward or lateral orientations in sign production.

Movement features show the highest PMI values overall, with DGS (hamclockdr-hamclocku, PMI = 9.91±0.061) and GSL (hamcircleil-hamstirccw, PMI = 8.91±0.055) demonstrating highly structured temporal motion patterns. Circular and clock-like movements (hamcircle, hamclock, hamstir) dominate across all languages, indicating a shared visual rhythm in sign articulation. These patterns suggest that cyclic and directional movements are central to sign semantics and may serve as phonological markers for verb or action-related signs.

Non-manual features show lower PMI values overall, indicating more diffuse or context-dependent usage. BSL shows the strongest co-occurrence (hameyes-hamnose, PMI = 4.30±0.027), suggesting facial articulation plays a significant role in sign contrast. LSF and GSL show moderate co-occurrence between hamchin, hamhead, and hamneck, pointing to a layered use of facial and neck gestures. DGS shows relatively low PMI values, possibly reflecting a more manual-centric phonological system or less reliance on facial features.

4.6 Inter-Phonological Co-occurrence

As depicted in Table 3 We also analyze how different category of phonological features interact among themselves. For example, how handshapes interact with movements, or locations in a particular language's signing space. We found BSL and GSL show higher co-occurrence between hamthumboutmod and

	BSL		DGS		LSF		GSL	
Type	Pairs	Freq	Pairs	Freq	Pairs	Freq	Pairs	Freq
HS + LOC	hamthumboutmod-hampalml	259	hamflathand-hamextfingerol	122	hamthumboutmod-hampalml	211	hamthumboutmod-hambetween	259
	hamthumboutmod-hambetween	249	hamflathand-hampalml	107	hamthumbacrossmod-hampalmd	180	hamthumboutmod-hamsymmlr	248
	hamthumboutmod-hamsymmlr	227	hamflathand-hampalmd	102	hamthumboutmod-hampalmd	172	hamflathand-hambetween	229
	hamthumboutmod-hampalmd	208	hamfist-hampalml	97	hamthumbacrossmod-hamextfingeru	165	hamflathand-hamsymmlr	229
	hamthumbacrossmod-hamextfingeru	190	fingerstraightmod-hamsymmlr	97	hamthumbacrossmod-hampalml	143	hamthumboutmod-hampalml	225
HS+ NMA	hamthumboutmod-hamrepeatfromstart	111	hamfingerstraightmod-hamrepeatfromstart	89	hamthumboutmod-hamrepeatfromstart	104	hamthumboutmod-hamrepeatfromstart	90
	hamthumboutmod-hammoved	99	hamfist-hamrepeatfromstart	75	hamthumbacrossmod-hamrepeatfromstart	91	hamthumbacrossmod-hamrepeatfromstart	87
	hamthumbacrossmod-hamrepeatfromstart	95	hamflathand-hamrepeatfromstart	65	hamfingerstraightmod-hamrepeatfromstart	79	hamflathand-hamrepeatfromstart	86
	hamflathand-hamrepeatfromstart	75	fingerstraightmod-hammoved	62	hamthumboutmod-hammoved	79	hamthumboutmod-hamfast	84
	hamfinger2-hamrepeatfromstart	67	hamfinger2-hamrepeatfromstart	62	hamthumbacrossmod-hammoved	75	hamthumboutmod-hammoved	79
HS+ NMA	hamthumboutmod-hamshoulders	166	hamthumboutmod-hamchest	29	hamthumboutmod-hamchest	163	hamthumbacrossmod-hamshoulders	181
	hamthumboutmod-hamchest	139	hamfinger2345-hamchest	25	hamthumboutmod-hamshoulders	137	hamthumboutmod-hamchest	173
	hamthumbacrossmod-hamshoulders	101	fingerstraightmod-hamshoulders	22	hamthumbacrossmod-hamshoulders	133	hamflathand-hamchest	166
	hamflathand-hamshoulders	99	hamflathand-hamshoulders	22	hamflathand-hamchest	113	hamthumboutmod-hamshoulders	159
	hamflathand-hamchest	90	hamthumboutmod-hamshoulders	21	hamthumbacrossmod-hamchest	105	hamthumbacrossmod-hamchest	155
MOV+ LOC	hamrepeatfromstart-hampalml	119	hamrepeatfromstart-hampalml	124	hamrepeatfromstart-hampalml	108	hamrepeatfromstart-hambetween	135
	hamrepeatfromstart-hambetween	113	hamrepeatfromstart-hamsymmlr	101	hamrepeatfromstart-hamextfingeru	100	hamrepeatfromstart-hamsymmlr	124
	hammoved-hamsymmlr	90	hammoved-hampalml	92	hamrepeatfromstart-hampalmd	92	hamrepeatfromstart-hampalml	113
	hamrepeatfromstart-hamextfingeru	88	hamrepeatfromstart-hamextfingerol	90	hammoved-hampalml	83	hamfast-hambetween	113
	hammoved-hampalml	88	hammoved-hamextfingerol	79	hammoved-hampalmd	78	hammoved-hambetween	112
MOV+ NMA	hamrepeatfromstart-hamshoulders	76	hammoved-hamshoulders	35	hammoved-hamshoulders	81	hammoved-hamchest	93
	hammoved-hamshoulders	56	hamrepeatfromstart-hamshoulders	20	hamrepeatfromstart-hamshoulders	78	hamrepeatfromstart-hamchest	88
	hammoved-hamchest	53	hamrepeatfromstart-hamchin	20	hammoved-hamchest	73	hammoved-hamshoulders	81
	hamrepeatfromstart-hamchest	53	hamrepeatfromstart-hamchest	20	hamrepeatfromstart-hamchest	65	hamhalt-hamshoulders	80
	hammover-hamshoulders	34	hammoved-hamchest	13	repeatfromstartseveral hamshoulders	53	hamrepeatfromstart-hamshoulders	77
NMA+ LOC	hamshoulders-hambetween	168	hamshoulders-hamsymmlr	46	hamshoulders-hampalml	148	hamchest-hambetween	260
	hamshoulders-hampalml	151	hamchin-hampalml	42	hamshoulders-hampalmd	146	hamchest-hamsymmlr	236
	hamshoulders-hamsymmlr	142	hamchin-hamextfingerul	37	hamchest-hampalmd	140	hamshoulders-hamsymmlr	235
	hamchest-hambetween	137	hamchest-hampalml	35	hamchest-hampalml	129	hamshoulders-hambetween	229
	hamshoulders-hampalmd	121	hamshoulders-hamextfingeruo	34	hamchest-hamextfingero	122	hamchest-hampalml	206

Table 3: Frequency distributions of co-occurrences of phonological features across different sign languages.

spatial locations like hampalml, hambetween, and hamsymmlr, suggesting that this handshape is highly versatile and frequently used in central signing space. DGS favors combinations like hamflathand-hamextfingerol and hamflathand-hampalml, indicating a preference for flat hand configurations in extended or lateral orientations. LSF shows similar patterns to BSL, with hamthumboutmod and hamthumbacrossmod frequently paired with hampalml and hampalmd, reflecting a balanced use of thumb-based handshapes in mid-body locations.

Across all languages, hamrepeatfromstart is the most frequent movement paired with dominant handshapes (hamthumboutmod, hamthumbacrossmod, hamflathand), reinforcing its role as a core phonological motion. BSL and LSF show strong pairings of hamthumboutmod with both hamrepeatfromstart and hammoved, suggesting a dynamic use of thumb-based signs. GSL includes hamfast in its top co-occurrences, indicating a tendency toward rapid articulation in certain handshape-movement combinations.

BSL, LSF, and GSL show strong co-occurrence between hamthumboutmod and upper-torso cues, while GSL uniquely favors hamthumbacrossmod-hamshoulders and hamflathand-hamchest, reflecting rich manual–non-manual integration. DGS, with lower overall frequencies and modest pairings like hamfinger2345-hamchest, suggests a more manual-centric system.

hamrepeatfromstart usually co-occurs with hampalml, hambetween, and hamsymmlr across all languages, confirming its central role in spatially anchored sign articulation. GSL shows strong pairings of hamfast with hambetween, suggesting a preference for fast, centrally located signs. BSL, DGS and LSF include extended finger orientations (like hamextfingeru, hamextfingerol) in frequent pairings, indicating a nuanced use of directional movement.

Co-occurrence between hamrepeatfromstart and hamshoulders or hamchest is common in BSL, LSF, and GSL, reinforcing the idea that repetitive movements are often accompanied by expressive non-manual cues. GSL shows the highest integration, with hammoved-hamchest and hamrepeatfromstart-hamchest appearing frequently, suggesting a significant coupling of motion and torso-based expression. DGS shows lower frequencies and more facial-centric pairings (e.g., hamrepeatfromstart-hamchin), indicating a different balance of articulatory features.

BSL and GSL show high co-occurrence between hamshoulders and hambetween, suggesting that upper-body non-manual features are often used in central signing space. LSF shows high frequencies for hamchest-hampalml and hamshoulders-hampalmd, indicating a preference for mid-body articulation zones. DGS includes more facial and lateral pairings (e.g., hamchin-hampalml, hamchin-hamextfingerul), reflecting a more distributed use of non-manual features.

In summary we observe that BSL and GSL exhibit strong centralization in signing space, with frequent use of hambetween and upper torso non-manuals. DGS shows a more distributed and facially oriented phonological structure, with lower integration of non-manuals and more lateral articulations. LSF balances manual and non-manual features with a preference for mid-body locations and thumb-based handshapes. These patterns reveal universal tendencies and regional variations in phonological feature co-occurrence, offering insights into the structural and cultural shaping of sign languages.

5 Conclusion

We analyzed phonological similarity across four sign languages using HamNoSys-based symbolic representations. By comparing 1000 signs per language, we identified consistent patterns in handshape, movement, and spatial usage, along with notable differences in non-manual features and articulation styles. Co-occurrence analysis revealed strong intra- and inter-feature dependencies, suggesting both universal phonological structures and geo-linguistic variation. LSF and DGS show higher internal consistency, and sign clustering reveals phonological families—laying the groundwork for multilingual sign language modeling and cross-linguistic phonological transfer. All these observations are based on raw frequency counts; formal statistical testing will be included in future work.

References

Suvarna Rajesh Bhagwat, RP Bhavsar, and BV Pawar. 2024. Marathi to indian sign language machine translation. *ACM Transactions on Asian and Low-Resource Language Information Processing*.

Diane Brentari. 1998. *A prosodic model of sign language phonology*. Mit Press.

Necati Cihan Camgoz, Simon Hadfield, Oscar Koller, Hermann Ney, and Richard Bowden. 2018. Neural sign language translation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7784–7793.

Necati Cihan Camgoz, Oscar Koller, Simon Hadfield, and Richard Bowden. 2020. Sign language transformers: Joint end-to-end sign language recognition and translation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10023–10033.

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*, 35:17043–17056.

Necati Cihan Camgoz, Simon Hadfield, Oscar Koller, and Richard Bowden. 2017. Subunets: End-to-end hand shape and continuous sign language recognition. In *Proceedings of the IEEE international conference on computer vision*, pages 3056–3065.

OA Crasborn and IEP Zwitserlood. 2008. Annotation of video data in the corpus ngt.

Eleni Efthimiou, Stavroula-Evita Fontinea, Thomas Hanke, John Glauert, Rihard Bowden, Annelies Braffort, Christophe Collet, Petros Maragos, and François Goudenove. 2010. Dicta-sign–sign language recognition, generation and modelling: a research effort with applications in deaf communication. In *Proceedings of the 4th Workshop on the Representation and Processing of Sign Languages: Corpora and Sign Language Technologies*, pages 80–83.

Eleni Efthimiou, Stavroula-Evita Fotinea, Thomas Hanke, John Glauert, Richard Bowden, Annelies Braffort, Christophe Collet, Petros Maragos, and François Lefebvre-Albaret. 2012. The dicta-sign wiki: Enabling web communication for the deaf. In *International conference on computers for handicapped persons*, pages 205–212. Springer.

Eleni Efthimiou, Stavroula-Evita Fotinea, Christian Vogler, Thomas Hanke, John Glauert, Richard Bowden, Annelies Braffort, Christophe Collet, Petros Maragos, and Jérémie Segouat. 2009. Sign language recognition, generation, and modelling: A research effort with applications in deaf communication. In *International Conference on Universal Access in Human-Computer Interaction*, pages 21–30. Springer.

Thomas Hanke. 2004. Hamnosys–representing sign language data in language resources and language processing contexts. In *sign-lang@ LREC 2004*, pages 1–6. European Language Resources Association (ELRA).

Thomas Hanke. 2010. Hamnosys–hamburg notation system for sign languages. *Institute of German Sign Language, Accessed in*, 7.

Lee Kezar, Jesse Thomason, Naomi Caselli, Zed Sehyr, and Elana Pontecorvo. 2023. The semlex benchmark: Modeling asl signs and their phonemes. In *Proceedings of the 25th International ACM SIGACCESS Conference on Computers and Accessibility*, pages 1–10.

Gabriele Langer, Susanne König, and Silke Matthes. 2014. Compiling a basic vocabulary for german sign language (dgs)–lexicographic issues with a focus on word senses. In *Proceedings of the XVI EURALEX International Congress: The User in Focus*, pages 767–786.

Sara Morrissey. 2008. *Data-driven machine translation for sign languages*. Ph.D. thesis, Dublin City University.

Carolina Neves, Luísa Coheur, and Hugo Nicolau. 2020. Hamnosys2sigml: translating hamnosys into sigml. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 6035–6039.

Ellen Ormel, Daan Hermans, Harry Knoors, Angelique Hendriks, and Ludo Verhoeven. 2010. Phonological activation during visual word recognition in deaf and hearing children. *Journal of Speech, Language, and Hearing Research*, 53(4):801–820.

Siegmund Prillwitz and Hamburg Zentrum für Deutsche Gebärdensprache und Kommunikation Gehörloser. 1989. *Hamnosys: Version 2.0; hamburg notation system for sign languages; an introductory guide*. Signum-Verlag.

J Martinez Rodriguez, Martha Larson, and L ten Bosch. 2023. Exploring the importance of sign language phonology for a deep neural network.

Ben Saunders, Necati Cihan Camgoz, and Richard Bowden. 2020. Adversarial training for multi-channel sign language production. *arXiv preprint arXiv:2008.12405*.

Samar Sinha. 2009. *A grammar of Indian sign language*. Ph.D. thesis, PhD dissertation, Jawaharlal Nehru University, New Delhi, India.

William C Stokoe. 1960. Sign language structure (studies in linguistics. *Occasional paper*, 8.

Stephanie Stoll, Necati Cihan Camgoz, Simon Hadfield, and Richard Bowden. 2020. Text2sign: towards sign language production using neural machine translation and generative adversarial networks. *International Journal of Computer Vision*, 128(4):891–908.

Sugandhi, Parteek Kumar, and Sanmeet Kaur. 2020. Sign language generation system based on indian sign language grammar. *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)*, 19(4):1–26.

Federico Tavella, Viktor Schlegel, Marta Romeo, Aphrodite Galata, and Angelo Cangelosi. 2022. Wlasl-lex: a dataset for recognising phonological properties in american sign language. *arXiv preprint arXiv:2203.06096*.

Harry Walsh, Ben Saunders, and Richard Bowden. 2022. Changing the representation: Examining language representation for neural sign language production. *arXiv preprint arXiv:2210.06312*.

Joshua T Williams, Adam Stone, and Sharlene D Newman. 2017. Operationalization of sign language phonological similarity and its effects on lexical access. *The Journal of Deaf Studies and Deaf Education*, 22(3):303–315.

Li Yujian and Liu Bo. 2007. A normalized levenshtein distance metric. *IEEE transactions on pattern analysis and machine intelligence*, 29(6):1091–1095.

A HamNoSys Explanations

Table 4 shows the detailed explanation and category of what each HamNoSys symbol means along with its unicode (Neves et al., 2020).

HamNoSys	Unicode	Explanation	Category
hamspace	0020	Space separator (used to separate symbols or words).	Other
hamexclaim	0021	Punctuation marker (e.g., exclamation, comma, full stop, question) for transcriptions.	Other
hamcomma	002C	Punctuation marker (e.g., exclamation, comma, full stop, question) for transcriptions.	Other
hamfullstop	002E	Punctuation marker (e.g., exclamation, comma, full stop, question) for transcriptions.	Other
hamquery	003F	Punctuation marker (e.g., exclamation, comma, full stop, question) for transcriptions.	Other
hamaltbegin	007B	Alternative/parenthetical markers used to bracket alternate transcriptions or metadata.	Other
hammetaalt	007C	Alternative/parenthetical markers used to bracket alternate transcriptions or metadata.	Other
hamaltend	007D	Alternative/parenthetical markers used to bracket alternate transcriptions or metadata.	Other
hamfist	E000	Fist handshape (closed hand).	Hand Shapes
hamflathand	E001	Flat handshape (palm and fingers extended and close together, like a flat hand).	Hand Shapes
hamfinger2	E002	Two-finger configuration (usually index+middle extended).	Hand Shapes
hamfinger23	E003	Two adjacent fingers extended (index+middle) in non-spread configuration.	Hand Shapes
hamfinger23spread	E004	Two adjacent fingers extended and spread apart (index+middle spread).	Hand Shapes
hamfinger2345	E005	Fingers 2 5 extended (index through little finger), excluding thumb.	Hand Shapes
hampinch12	E006	Pinch-like handshape (thumb and one or more fingers pinching together).	Hand Shapes
hampinchall	E007	Pinch-like handshape (thumb and one or more fingers pinching together).	Hand Shapes
hampinch12open	E008	Pinch-like handshape (thumb and one or more fingers pinching together).	Hand Shapes
hamcee12	E009	C-shaped hand configuration (curved hand like letter 'C').	Hand Shapes
hamceeall	E00A	C-shaped hand configuration (curved hand like letter 'C').	Hand Shapes
hamceeopen	E00B	C-shaped hand configuration (curved hand like letter 'C').	Hand Shapes
hamthumboutmod	E00C	Thumb pointed outwards (thumb extended away from palm) a thumb position modifier.	Hand Shapes
hamthumbacrossmod	E00D	Thumb lying across the palm or fingers a thumb position modifier.	Hand Shapes
hamthumbopenmod	E00E	Thumb held open (not tucked in) modifier for thumb openness.	Hand Shapes
hamfingerstraightmod	E010	Handshape specifying particular fingers extended.	Hand Shapes
hamfingerbendmod	E011	Handshape specifying particular fingers extended.	Hand Shapes
hamfingerhookmod	E012	Handshape specifying particular fingers extended.	Hand Shapes
hamdoublebent	E013	Modifier for double-bent or double-hooked finger shapes (complex finger bend).	Hand Shapes
hamdoublehooked	E014	Modifier for double-bent or double-hooked finger shapes (complex finger bend).	Hand Shapes
hamextfingeru	E020	Finger direction marker extended finger points up (used to show finger orientation).	Location/Orientation
hamextfingerur	E021	Finger direction marker extended finger points up-right (used to show finger orientation).	Location/Orientation

HamNoSys	Unicode	Explanation	Category
hamextfingerr	E022	Finger direction marker extended finger points right (used to show finger orientation).	Location/Orientation
hamextfingerdr	E023	Finger direction marker extended finger points down-right (used to show finger orientation).	Location/Orientation
hamextfingerd	E024	Finger direction marker extended finger points down (used to show finger orientation).	Location/Orientation
hamextfingerdl	E025	Finger direction marker extended finger points down-left (used to show finger orientation).	Location/Orientation
hamextfingerl	E026	Finger direction marker extended finger points left (used to show finger orientation).	Location/Orientation
hamextfingerul	E027	Finger direction marker extended finger points up-left (used to show finger orientation).	Location/Orientation
hamextfingerol	E028	Finger direction marker extended finger points out-left (used to show finger orientation).	Location/Orientation
hamextfingero	E029	Finger direction marker extended finger points out/away (used to show finger orientation).	Location/Orientation
hamextfingeror	E02A	Finger direction marker extended finger points out-right (used to show finger orientation).	Location/Orientation
hamextfingeril	E02B	Finger direction marker extended finger points in-left (used to show finger orientation).	Location/Orientation
hamextfingeri	E02C	Finger direction marker extended finger points in/toward (used to show finger orientation).	Location/Orientation
hamextfingerir	E02D	Finger direction marker extended finger points in-right (used to show finger orientation).	Location/Orientation
hamextfingerui	E02E	Finger direction marker extended finger points up-in (used to show finger orientation).	Location/Orientation
hamextfingerdi	E02F	Finger direction marker extended finger points down-in (used to show finger orientation).	Location/Orientation
hamextfingerdo	E030	Finger direction marker extended finger points down-out (used to show finger orientation).	Location/Orientation
hamextfingeruo	E031	Finger direction marker extended finger points up-out (used to show finger orientation).	Location/Orientation
hampalmu	E038	Palm orientation indicator (which way the palm faces: up/down/left/right or variants).	Location/Orientation
hampalmur	E039	Palm orientation indicator (which way the palm faces: up/down/left/right or variants).	Location/Orientation
hampalmr	E03A	Palm orientation indicator (which way the palm faces: up/down/left/right or variants).	Location/Orientation
hampalmdr	E03B	Palm orientation indicator (which way the palm faces: up/down/left/right or variants).	Location/Orientation
hampalmd	E03C	Palm orientation indicator (which way the palm faces: up/down/left/right or variants).	Location/Orientation
hampalmdl	E03D	Palm orientation indicator (which way the palm faces: up/down/left/right or variants).	Location/Orientation
hampalml	E03E	Palm orientation indicator (which way the palm faces: up/down/left/right or variants).	Location/Orientation
hampalmul	E03F	Palm orientation indicator (which way the palm faces: up/down/left/right or variants).	Location/Orientation
hamhead	E040	Head (general) indicates head as location or non-manual articulator.	Non-Manual Features
hamheadtop	E041	Top of the head (specific location).	Non-Manual Features
hamforehead	E042	Forehead (location; often for non-manuals like eyebrow movement).	Other
hameyebrows	E043	Eyebrows (non-manual feature raise/lower etc).	Non-Manual Features
hameyes	E044	Eyes (gaze direction or eye activity).	Non-Manual Features
hamnose	E045	Nose (facial location).	Non-Manual Features
hamnostrils	E046	Nostrils (specific part of nose).	Other
hamear	E047	Ear (location).	Other
hamearlobe	E048	Earlobe (location).	Other
hamcheek	E049	Cheek (facial location).	Other
hamlips	E04A	Lips / mouth area (non-manual/mouthings).	Non-Manual Features
hamtongue	E04B	Tongue (mouth articulation reference).	Other
hamteeth	E04C	Teeth (mouth reference).	Other

HamNoSys	Unicode	Explanation	Category
hamchin	E04D	Chin (location reference).	Non-Manual Features
hamunderchin	E04E	Under-chin (location).	Other
hamneck	E04F	Neck (location / non-manual).	Non-Manual Features
hamshouldertop	E050	Top of the shoulder (location).	Non-Manual Features
hamshoulders	E051	Shoulders (body reference).	Non-Manual Features
hamchest	E052	Chest (body location).	Non-Manual Features
hamstomach	E053	Stomach/abdomen area (location).	Non-Manual Features
hambelowstomach	E054	Lower stomach/abdomen (location).	Other
hamlrbeside	E058	Location: left/right beside (side position next to body).	Other
hamlrat	E059	Location: left/right at (side location marker) indicates side-relative placement.	Other
hamcoreftag	E05A	Coreference tag (used for referencing another element or anchor in notation).	Location/Orientation
hamcorefref	E05B	Coreference reference (points to a previously defined anchor or location).	Location/Orientation
hamneutralspace	E05F	Neutral signing space in front of the signer (space away from body).	Location/Orientation
hamupperarm	E060	Upper arm (location reference).	Other
hamelbow	E061	Elbow (location).	Other
hamelbowinside	E062	Inner side of the elbow (specific location).	Other
hamlowerarm	E063	Lower arm / forearm (location).	Other
hamwristback	E064	Back of the wrist (location).	Location/Orientation
hamwristpulse	E065	Wrist pulse area (location).	Location/Orientation
hamthumbball	E066	Bulbous part of thumb (thumb pad/ball) used as a location reference.	Hand Shapes
hampalm	E067	Palm orientation indicator (which way the palm faces: up/down/left/right or variants).	Location/Orientation
hamhandback	E068	Back of hand (dorsal side).	Location/Orientation
hamthumbside	E069	Thumb-related handshape or modifier.	Hand Shapes
hampinkyside	E06A	Pinky-side (ulnar side) of hand.	Location/Orientation
hamthumb	E070	Thumb-related handshape or modifier.	Hand Shapes
hamindexfinger	E071	Index finger (reference) used as location/orientation reference.	Other
hammiddlefinger	E072	Middle finger used as location/orientation reference.	Other
hamringfinger	E073	Ring finger used as location/orientation reference.	Other
hampinky	E074	Little finger / pinky used as location/orientation reference.	Location/Orientation
hamfingertip	E075	Handshape specifying particular fingers extended.	Hand Shapes
hamfingernail	E076	Handshape specifying particular fingers extended.	Hand Shapes
hamfingerpad	E077	Handshape specifying particular fingers extended.	Hand Shapes
hamfingermidjoint	E078	Handshape specifying particular fingers extended.	Hand Shapes
hamfingerbase	E079	Handshape specifying particular fingers extended.	Hand Shapes
hamfingerside	E07A	Handshape specifying particular fingers extended.	Hand Shapes
hamwristtopulse	E07C	Top/inner wrist near the pulse location reference.	Location/Orientation
hamwristtoback	E07D	From wrist top toward back of wrist orientation reference.	Location/Orientation
hamwristtothumb	E07E	Thumb-related handshape or modifier.	Location/Orientation
hamwristtopinky	E07F	Orientation/position from wrist toward pinky side.	Location/Orientation
hammoveu	E080	Hand movement direction: up (linear path in that direction).	Movements
hammoveur	E081	Hand movement direction: up-right (linear path in that direction).	Movements
hammover	E082	Hand movement direction: right (linear path in that direction).	Movements
hammovedr	E083	Hand movement direction: down-right (linear path in that direction).	Movements
hammoved	E084	Hand movement direction: down (linear path in that direction).	Movements
hammovedl	E085	Hand movement direction: down-left (linear path in that direction).	Movements
hammovel	E086	Hand movement direction: left (linear path in that direction).	Movements
hammoveul	E087	Hand movement direction: up-left (linear path in that direction).	Movements
hammoveol	E088	Hand movement direction: out-left (linear path in that direction).	Movements
hammoveo	E089	Hand movement direction: out/away (linear path in that direction).	Movements
hammoveor	E08A	Hand movement direction: out-right (linear path in that direction).	Movements
hammoveil	E08B	Hand movement direction: in-left (linear path in that direction).	Movements
hammovei	E08C	Hand movement direction: in/toward (linear path in that direction).	Movements

HamNoSys	Unicode	Explanation	Category
hammoveir	E08D	Hand movement direction: in-right (linear path in that direction).	Movements
hammoveui	E08E	Hand movement direction: up-in (linear path in that direction).	Movements
hammovedi	E08F	Hand movement direction: down-in (linear path in that direction).	Movements
hammovedo	E090	Hand movement direction: down-out (linear path in that direction).	Movements
hammoveuo	E091	Hand movement direction: up-out (linear path in that direction).	Movements
hamcircleo	E092	Circular movement path around out/away (circle in that orientation).	Movements
hamcirclei	E093	Circular movement path around in/toward (circle in that orientation).	Movements
hamcircled	E094	Circular movement path around down (circle in that orientation).	Movements
hamcircleu	E095	Circular movement path around up (circle in that orientation).	Movements
hamcirclel	E096	Circular movement path around left (circle in that orientation).	Movements
hamcircler	E097	Circular movement path around right (circle in that orientation).	Movements
hamcircleul	E098	Circular movement path around up-left (circle in that orientation).	Movements
hamcircledr	E099	Circular movement path around down-right (circle in that orientation).	Movements
hamcircleur	E09A	Circular movement path around up-right (circle in that orientation).	Movements
hamcircledl	E09B	Circular movement path around down-left (circle in that orientation).	Movements
hamcircleol	E09C	Circular movement path around out-left (circle in that orientation).	Movements
hamcircleir	E09D	Circular movement path around in-right (circle in that orientation).	Movements
hamcircleor	E09E	Circular movement path around out-right (circle in that orientation).	Movements
hamcircleil	E09F	Circular movement path around in-left (circle in that orientation).	Movements
hamcircleui	E0A0	Circular movement path around up-in (circle in that orientation).	Movements
hamcircledo	E0A1	Circular movement path around down-out (circle in that orientation).	Movements
hamcircleuo	E0A2	Circular movement path around up-out (circle in that orientation).	Movements
hamcircledi	E0A3	Circular movement path around down-in (circle in that orientation).	Movements
hamfingerplay	E0A4	Handshape specifying particular fingers extended.	Hand Shapes
hamnodding	E0A5	General HamNoSys element (specific meaning depends on context).	Other
hamswinging	E0A6	General HamNoSys element (specific meaning depends on context).	Movements
hamtwisting	E0A7	General HamNoSys element (specific meaning depends on context).	Movements
hamstircw	E0A8	General HamNoSys element (specific meaning depends on context).	Movements
hamstirccw	E0A9	General HamNoSys element (specific meaning depends on context).	Movements
hamreplace	E0AA	General HamNoSys element (specific meaning depends on context).	Other

HamNoSys	Unicode	Explanation	Category
hammovecross	E0AD	Hand movement direction: directional movement (linear path in that direction).	Movements
hammoveX	E0AE	Hand movement direction: directional movement (linear path in that direction).	Movements
hamnomotion	E0AF	General HamNoSys element (specific meaning depends on context).	Other
hamclocku	E0B0	Clockwise/counterclockwise circular motion indicated by clock position 'u'.	Movements
hamclockul	E0B1	Clockwise/counterclockwise circular motion indicated by clock position 'ul'.	Movements
hamclockl	E0B2	Clockwise/counterclockwise circular motion indicated by clock position 'l'.	Movements
hamclockdl	E0B3	Clockwise/counterclockwise circular motion indicated by clock position 'dl'.	Movements
hamclockd	E0B4	Clockwise/counterclockwise circular motion indicated by clock position 'd'.	Movements
hamclockdr	E0B5	Clockwise/counterclockwise circular motion indicated by clock position 'dr'.	Movements
hamclockr	E0B6	Clockwise/counterclockwise circular motion indicated by clock position 'r'.	Movements
hamclockur	E0B7	Clockwise/counterclockwise circular motion indicated by clock position 'ur'.	Movements
hamclockfull	E0B8	Full circular clockwise motion (full rotation).	Movements
hamarcl	E0B9	Short arced movement (a small curved path).	Movements
hamarcu	E0BA	Short arced movement (a small curved path).	Movements
hamarcr	E0BB	Short arced movement (a small curved path).	Movements
hamarcd	E0BC	Short arced movement (a small curved path).	Movements
hamwavy	E0BD	Wavy oscillating movement (smooth wave-like motion).	Movements
hamzigzag	E0BE	Zig-zag oscillating movement (sharp alternating motion).	Other
hamellipseh	E0C0	Elliptical (oval) movement path, specifying orientation of ellipse.	Movements
hamellipseur	E0C1	Elliptical (oval) movement path, specifying orientation of ellipse.	Movements
hamellipsev	E0C2	Elliptical (oval) movement path, specifying orientation of ellipse.	Movements
hamellipseul	E0C3	Elliptical (oval) movement path, specifying orientation of ellipse.	Movements
hamincreasing	E0C4	Movement or parameter increasing (e.g., amplitude growing).	Other
hamdecreasing	E0C5	Movement or parameter decreasing (e.g., amplitude shrinking).	Other
hamsmallmod	E0C6	Modifier: small (subtle / small-amplitude) movement.	Other
hamlargemod	E0C7	Modifier: large (wide / large-amplitude) movement.	Other
hamfast	E0C8	Modifier: fast speed.	Movements
hamslow	E0C9	Modifier: slow speed.	Movements
hamtense	E0CA	Modifier: tense or stiff quality of movement/hand.	Movements
hamrest	E0CB	Rest position (hold without motion).	Movements
hamhalt	E0CC	Abrupt stop / halt in motion.	Movements
hamclose	E0D0	Hand closing or coming together (close action).	Other
hamtouch	E0D1	Touch/contact action (hand touches another part).	Other
haminterlock	E0D2	Hands interlocking (fingers interlaced) action.	Other
hamcross	E0D3	Crossing hands or crossing motion/placement.	Other
hamarmextended	E0D4	Arm is extended away from body (extended-arm posture).	Location/Orientation
hambehind	E0D5	Placed or moved behind body or another body-part.	Other
hambrushing	E0D6	Brushing motion (light stroke across surface).	Other
hamrepeatfromstart	E0D8	Repetition operator indicates repeating the movement or sequence.	Movements
hamrepeatfromstartseveral	E0D9	Repetition operator indicates repeating the movement or sequence.	Movements
hamrepeatcontinue	E0DA	Repetition operator indicates repeating the movement or sequence.	Movements
hamrepeatcontinueseveral	E0DB	Repetition operator indicates repeating the movement or sequence.	Movements

HamNoSys	Unicode	Explanation	Category
hamrepeatreverse	E0DC	Repetition operator indicates repeating the movement or sequence.	Movements
hamalternatingmotion	E0DD	Alternating motion (hands or fingers alternate in action).	Movements
hamseqbegin	E0E0	Sequence/grouping marker: begins/ends a sequence, parallel group, or fusion of actions.	Other
hamseqend	E0E1	Sequence/grouping marker: begins/ends a sequence, parallel group, or fusion of actions.	Other
hamparbegin	E0E2	Sequence/grouping marker: begins/ends a sequence, parallel group, or fusion of actions.	Other
hamparend	E0E3	Sequence/grouping marker: begins/ends a sequence, parallel group, or fusion of actions.	Other
hamfusionbegin	E0E4	Sequence/grouping marker: begins/ends a sequence, parallel group, or fusion of actions.	Other
hamfusionend	E0E5	Sequence/grouping marker: begins/ends a sequence, parallel group, or fusion of actions.	Other
hambetween	E0E6	Spatial relation: between (e.g., movement or placement between hands or body parts).	Location/Orientation
hamplus	E0E7	Plus symbol: combines or adds elements (used in composite descriptions).	Other
hamsymmpar	E0E8	Symmetry operator: indicates two-handed symmetry (how attributes mirror across hands).	Location/Orientation
hamsymmlr	E0E9	Symmetry operator: indicates two-handed symmetry (how attributes mirror across hands).	Location/Orientation
hamnondominant	E0EA	Marker referring to the non-dominant hand (used to describe NDH behaviour).	Location/Orientation
hamnonipsi	E0EB	Marker meaning non-ipsilateral / opposite-side reference (side-related indicator).	Location/Orientation
hametc	E0EC	Placeholder: 'etc.' or miscellaneous/other elements not explicitly listed.	Other
hamorirelative	E0ED	Orientation/relative reference marker (indicates orientation relative to something else).	Location/Orientation
hammime	E0F0	Mime or pantomime marker indicates mimed action rather than lexical sign.	Non-Manual Features

Table 4: Explanations of HamNoSys symbols

Pose-Based Sign Language Spotting via an End-to-End Encoder Architecture

Samuel Ebimobowei Johnny[1] **Blessed Guda**[1,2]
Emmanuel Enejo Aaron[1] Assane Gueye[1,2]
{sjohnny, blessedg, eaaron, assaneg}@andrew.cmu.edu
[1]Carnegie Mellon University Africa, Kigali, Rwanda
[2]Carnegie Mellon University, Pittsburgh, USA

Abstract

Automatic Sign Language Recognition (ASLR) has emerged as a vital field for bridging the gap between deaf and hearing communities. However, the problem of sign-to-sign retrieval or detecting a specific sign within a sequence of continuous signs remains largely unexplored. We define this novel task as Sign Language Spotting. In this paper, we present a first step toward sign language retrieval by addressing the challenge of detecting the presence or absence of a query sign video within a sentence-level gloss or sign video. Unlike conventional approaches that rely on intermediate gloss recognition or text-based matching, we propose an end-to-end model that directly operates on pose keypoints extracted from sign videos. Our architecture employs an encoder-only backbone with a binary classification head to determine whether the query sign appears within the target sequence. By focusing on pose representations instead of raw RGB frames, our method significantly reduces computational cost and mitigates visual noise. We evaluate our approach on the Word Presence Prediction dataset from the WSLP 2025 shared task, achieving 61.88% accuracy and 60.00% F1-score. These results demonstrate the effectiveness of our pose-based framework for Sign Language Spotting, establishing a strong foundation for future research in automatic sign language retrieval and verification. Code is available at this repository.

1 Introduction

Sign language, which globally consists of more than 300 different sign languages (United Nations, 2023), was developed to address the need for effective communication for the deaf and hearing-impaired population (Tunga et al., 2021). Each sign language comprises a complex combination of hand gestures, facial expressions, and body movements that collectively encode the semantics and grammatical structures of spoken languages (Tang et al., 2025; Rastgoo et al., 2024). However, there is still a challenge and a communication gap between the deaf and hearing community (Das et al., 2024), (Venugopalan and Reghunadhan, 2021). Previous works have focused on sign language translation(SLT) where researchers have attempted to translate sign language either as RGB or poses to either text(that is word word-level semantically meaningful) (Yin and Read, 2020; Kan et al., 2022) or glosses (Zhou et al., 2023; Low, 2025).

Sign language recognition(SLR) could be isolated and continuous SLR. Isolated sign language (ISLR) (Kumari and Anand, 2024; Baihan et al., 2024; Ren et al., 2025) translation involves word-level focuses on recognizing individual signs in isolation, treating each sign as an independent classification problem. In contrast, Continuous Sign Language Recognition (CSLR) (Wang et al., 2025; Jian He et al., 2025; Zheng et al., 2023; Low, 2025) involves sentence-level SL, which addresses a more challenging task of translating continuous signing sequences into semantically correct sentences or gloss annotations, requiring models to handle temporal dependencies, co-articulation effects, and variable-length sequences.

While significant progress has been made in recognition and translation, the ability to search, retrieve, or verify specific signs within continuous signing videos remains underexplored. This capability- known as sign spotting - is critical for applications such as SL retrieval, dictionary lookup, and educational tools. This requires robust sign spotting capabilities, that is, the ability to locate and identify specific signs within continuous signing videos. Traditional approaches to this problem have relied on text-based intermediate representations.

For word spotting for CLSR, researchers have attempted to spot words using Large Language Models (LLMs). (Walsh et al., 2023) proposed using LLMs such as BERT and Word2Vec to leverage alignment to improve isolated signs from continu-

Proceedings of the Workshop on Sign Language Processing (WSLP), pages 71–76

ous signs. Their approach solves text gloss mapping using LLMs; their model provides an effective method, which was evaluated on MeieneDGS (Konrad et al., 2020) and BOBSL(Albanie et al., 2021).

Recent work on sign spotting addresses the challenge of SLT by decomposing it into modular stages. Spotter+GPT (Jian He et al., 2025) proposes an approach to eliminate the need for SLT-specific end-to-end training, significantly reducing computational costs. Their approach extracts I3D motion and ResNeXt-101 handshape features, matches them to a sign dictionary using DTW and cosine similarity, and passes spotted signs, which are the top-k glosses, to GPT for sentence generation. While Spotter+GPT demonstrates the effectiveness of modular SLT, our work addresses a fundamentally different task: word presence verification.

In this paper, we introduce a novel end-to-end video-to-video sign spotting framework that eliminates the need for textual or gloss-based intermediates. Given a query sign video and a sentence-level sign video, our model determines whether the query sign is present within the sentence. We adopt an encoder-only architecture with a binary classification head, operating directly on pose keypoints rather than RGB frames. This design reduces computational complexity and suppresses visual noise while maintaining discriminative spatial-temporal information. We evaluate our approach on the Word Presence Prediction dataset from the WSLP 2025 shared task[1]. To the best of our knowledge, this work represents the first study to address sign language spotting purely through video-to-video matching, establishing a foundation for future research in automatic sign language retrieval, verification, and search.

2 Methodology

We propose a video-to-video sign spotting architecture that jointly models visual–semantic alignment and binary word presence prediction. The framework learns robust cross-modal representations that generalize across signers and sentence contexts. Our approach consists of three main components: pose extraction, feature encoding, and presence prediction, as illustrated in Figure 1.

[1]https://exploration-lab.github.io/WSLP/task/

2.1 Pose Extraction

We use MediaPipe (Lugaresi et al., 2019) to convert RGB video sequences to pose-based representations, allowing for a more generalized, efficient, and resilient architecture. For each frame, MediaPipe estimates the pose keypoints of the signer in the video. Following (Johnny et al., 2025), we extract holistic pose features containing 42 hand keypoints (21 per hand), 8 body keypoints, and 19 facial landmarks. As suggested by (Johnny et al., 2025), we used only the hand and body features in this study.

2.2 Problem formulation

Given a sentence sequence $\mathbf{X}_s \in \mathbb{R}^{T_s \times F}$ and a query sequence $\mathbf{X}_q \in \mathbb{R}^{T_q \times F}$, where T_s and T_q denote the temporal lengths of the sentence and query sequences respectively, and F represents the dimensionality of the pose features, the objective is to determine whether the query sign appears within the sentence.

Let $f(\mathbf{X}_s, \mathbf{X}_q; \theta)$ be a parameterized model, where θ denotes the set of learnable parameters. The model outputs a probability score $\hat{y} = f(\mathbf{X}_s, \mathbf{X}_q; \theta)$, representing the likelihood that the query sign occurs in the given sentence. The binary prediction is made as: The training objective is to optimize the model parameters θ by minimizing a loss function $\mathcal{L}(\theta)$ over the training data:

$$\theta^* = \arg\min_{\theta} \mathcal{L}(\theta).$$

2.3 Pose CNN Encoder

Each pose frame is represented as a vector $\mathbf{x}_t \in \mathbb{R}^{100}$, corresponding to 50 keypoints with 2D coordinates (x, y). To preserve the spatial topology of the human skeleton, each vector is reshaped into a 2D array of size $\mathbb{R}^{50\times2}$, considering only the hand and body keypoints as described in Section 2.1.

A 2D CNN is applied independently to each frame to extract local spatial dependencies among keypoints. Specifically, each pose frame passes through three Conv2D blocks, each consisting of a convolutional layer, Batch Normalization, and ReLU activation. These blocks progressively capture hierarchical geometric patterns while maintaining spatial coherence across keypoints.

Following the convolutional layers, an **adaptive average pooling** layer reduces the spatial dimensions to a fixed-size representation, which is then

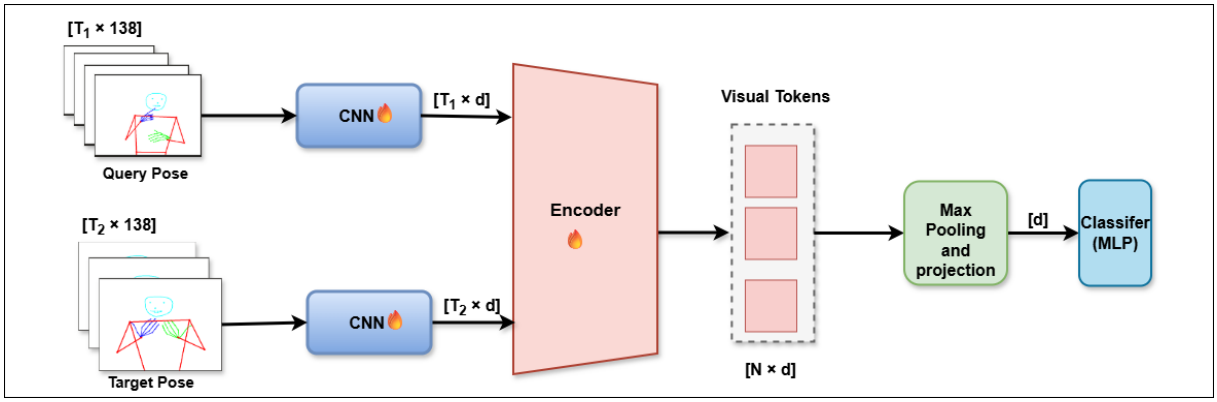

Figure 1: Architecture overview. Pose sequences are encoded using 2D CNNs and then processed by a Transformer encoder, which produces visual tokens. The **[CLS]** token is max-pooled to predict query presence using binary cross-entropy loss ($\mathcal{L}_{BCE}$).

linearly projected to a feature vector of dimension $d = 128$.

This process yields a sequence of per-frame embeddings:

$$\mathbf{H} = \{\mathbf{h}_t\}_{t=1}^{T} \in \mathbb{R}^{T\times 128}, \tag{1}$$

where T denotes the total frames in the video. Each embedding $\mathbf{h}_t$ encodes the spatial structure of the signer's body and hand poses at time step t. The resulting feature sequence is passed to the transformer encoder for temporal modeling.

2.4 Visual Transformer Encoder model

To enable temporal dependencies and cross-sequence interactions between the query and sentence embeddings, we adopt a BERT-style sequence modeling approach. Specifically, a [CLS] token is prepended for global sequence-level classification, while a [SEP] token is inserted to explicitly separate the *query pose tokens* from the *candidate pose tokens*. This design enables the model to attend across the boundary between the two sequences, allowing direct interaction between corresponding temporal segments.

Learnable positional encodings and token-type embeddings are incorporated to preserve temporal order and to distinguish between query and candidate sequences. The transformer encoder then processes the concatenated sequence using multi-head self-attention, where the **attention scores between query and candidate pose tokens** serve as a key mechanism for measuring their semantic and spatial correspondence. These cross-sequence attention patterns help the model identify whether visual and structural similarities exist between the query sign and any segment of the candidate video, thereby assisting the sign spotting task.

2.4.1 Classification Loss

Since the expected outcome is binary(present or absent), our model employs the binary cross-entropy loss(BCE) to penalize incorrect and overconfident predictions. We extract the **[CLS]** token representation via max pooling and project it to an MLP classifier to generate the corresponding logits $\hat{y} \in \mathbb{R}$. The final prediction is obtained by applying the sigmoid(σ) to the logits. The binary cross-entropy loss is computed as:

$$\mathcal{L}_{\text{BCE}} = -\frac{1}{B}\sum_{i=1}^{B}[y_i \log p_i + (1-y_i)\log(1-p_i)] \tag{2}$$

where $p = \sigma(\hat{y}_i))$, σ is the sigmoid function , $y_i \in \{0, 1\}$ is the ground truth label, and B is the batch size.

3 Experiments

3.1 Dataset and Evaluation Metrics

For this experiment, we evaluate our model with *Word Presence Dataset*[2], an ASL sign spotting dataset designed to determine if a query sign appears within a sentence sequence.

The dataset comprises 25,432 sentence-query pairs constructed from 7,857 unique sentence sequences and 1,410 unique query sequences. The dataset is balanced, with equal distribution of positive (query present) and negative (query absent) samples. We employ an 80:20 train-validation split.

[2] https://huggingface.co/datasets/Exploration-Lab/WSLP-AACL-2025

Table 1: Performance comparison on Word Presence Prediction (Test Set).

Method	Acc.	F1	Prec.	Rec.
Ours (1D CNN)	60.95	**59.62**	62.70	61.01
Ours (2D CNN)	**61.66**	58.42	**67.16**	**61.74**

The test set contains 1,266 unique sentence sequences and 555 unique query sequences, ensuring minimal overlap with the training distribution.

During Evaluation, we use standard classification evaluation metrics, i.e., Accuracy, Precision, Recall, and F1-score.

4 Implementation Details

We train the model end-to-end with an initial learning rate of 0.0005 since commonly used values (e.g., 0.001 or 0.01) resulted in suboptimal convergence, using the AdamW optimizer and a temperature of 0.07 for contrastive losses. A dropout of 0.02 was applied to prevent overfitting. Training is carried out for 50 epochs with a patience of 5 if no future improvements. This was done using a single NVIDIA L40S GPU.

To ensure our model focuses on important features, we skipped all early and late frames with no finger movement. During training, we applied different data augmentation techniques such as sequence masking, scaling, jittering, and Gaussian noise to ensure robustness.

4.1 Evaluation on Test Set

Table 1 presents our results on the test set. Given that this is a novel task introduced in the WSLP 2025 shared task, with no prior work to the best of our knowledge, hence no baseline to compare against. Our 2D-CNN approach achieves 61.66% accuracy, outperforming linear(1D-CNN) projection. Notably, 2D-CNN significantly improves precision, indicating fewer false positives, though F1 slightly decreases due to the precision-recall tradeoff.

4.2 Ablation study and analysis

To evaluate the robustness of our model, we conduct some ablations using different training choices with a concentration on Accuracy and F1 Scores.

4.2.1 Effect of different loss function

As shown in Table 2 demonstrate that using only contrastive loss underperforms when compared to using BCE, indicating that contrastive supervision is not satisfactory enough for this task. While combining both losses with contrastive weight $\lambda = 0.5$ achieves 61.39% accuracy, the result is still below BCE-only performance. The contrastive objective may interfere with classification if the weight is not carefully tuned; using either higher or lower weights results in lower performance, and the embedding space learned through mean pooling may be less discriminative than the [CLS] token representation for this verification task.

Table 2: Ablation study on validation set(val set).

Configuration	Acc.	F1	Prec.	Rec.
Loss Functions				
BCE only (ours)	**63.04**	**70.36**	59.19	86.71
Contrast only	57.20	69.27	54.39	**95.38**
BCE + Contrast	61.39	64.13	**60.49**	68.25
Pose Encoding				
1D Conv	53.65	**67.58**	52.29	95.53
2D Conv (ours)	**61.39**	64.13	**60.49**	**68.25**

4.2.2 Effect of Pose Encoding

Table 2 demonstrates that 2D-CNN outperforms other methods in encoding postures. 1D-CNN captures temporal patterns but treats keypoints as a sequence without using their geometric correlations. In contrast, 2D-CNN preserves spatial structure by reshaping each frame as an $n \times 2$ grid, where n is the number of keypoints, allowing the network to learn spatial patterns such as hand configurations and body postures.

5 Conclusion

In the work, we present the first video-to-video word presence verification in sign language, where both the sentence and query are video sequences. Our approach proposes using pose sequence in, combining 2D CNN encoding with a Transformer temporal model, achieving 61.66% accuracy on the *word presence dataset*.

To the best of our knowledge at the time of this research, no prior work has been done in video-to-video sentence-to-query word spotting. Ablation studies and analysis show that 2D spatial encoding of poses and BCE loss are critical design choices. Our work establishes a strong baseline for this task and demonstrates the effectiveness of pose-based representations for SL understanding.

References

Samuel Albanie, Gül Varol, Liliane Momeni, Hannah Bull, Triantafyllos Afouras, Himel Chowdhury, Neil Fox, Bencie Woll, Rob Cooper, Andrew McParland, and Andrew Zisserman. 2021. BOBSL: Bbc-oxford british sign language dataset. https://www.robots.ox.ac.uk/~vgg/data/bobsl.

Abdullah Baihan, Ahmed I. Alutaibi, Mohammed Alshehri, and Sunil Kumar Sharma. 2024. Sign language recognition using modified deep learning network and hybrid optimization: a hybrid optimizer (ho) based optimized cnnsa-lstm approach. *Scientific Reports*, 14:26111.

Subhadeep Das, Subrata Kr. Biswas, and Bidyut Purkayastha. 2024. Occlusion robust sign language recognition system for indian sign language using cnn and pose features. *Multimedia Tools and Applications*, 83(36):84141–84160.

Low Jian He, Ozge Mercanoglu Sincan, and Richard Bowden. 2025. Sign spotting disambiguation using large language models. In *Adjunct Proceedings of the 25th ACM International Conference on Intelligent Virtual Agents*, IVA Adjunct '25, New York, NY, USA. Association for Computing Machinery.

Samuel Ebimobowei Johnny, Blessed Guda, Andrew Blayama Stephen, and Assane Gueye. 2025. Autosign: Direct pose-to-text translation for continuous sign language recognition. *arXiv preprint arXiv:2507.19840*.

Jichao Kan, Kun Hu, Markus Hagenbuchner, Ah Chung Tsoi, Mohammed Bennamoun, and Zhiyong Wang. 2022. Sign language translation with hierarchical spatio-temporal graph neural network. In *2022 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 2131–2140.

Reiner Konrad, Thomas Hanke, Gabriele Langer, Dolly Blanck, Julian Bleicken, Ilona Hofmann, Olga Jeziorski, Lutz König, Susanne König, Rie Nishio, Anja Regen, Uta Salden, Sven Wagner, Satu Worseck, Oliver Böse, Elena Jahn, and Marc Schulder. 2020. Meine dgs – annotiert. öffentliches korpus der deutschen gebärdensprache, 3. release / my dgs – annotated. public corpus of german sign language, 3rd release.

Diksha Kumari and Radhey Shyam Anand. 2024. Isolated video-based sign language recognition using a hybrid cnn-lstm framework based on attention mechanism. *Electronics*, 13(7).

Jian He Low. 2025. Sage: Segment-aware gloss-free encoding for token-efficient sign language translation. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV 2025)*. Institute of Electrical and Electronics Engineers (IEEE).

Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Uboweja, Michael Hays, Fan Zhang, Chuo-Ling Chang, Ming Guang Yong, Juhyun Lee, and 1 others. 2019. Mediapipe: A framework for building perception pipelines. In *arXiv preprint arXiv:1906.08172*.

Rahimeh Rastgoo, Kourosh Kiani, and Sergio Escalera. 2024. Word separation in continuous sign language using isolated signs and post-processing. *Expert Systems with Applications*, 249:123695.

Yazhou Ren, Hongkai Li, Yuhao Li, Jingyu Pu, Xiaorong Pu, Siyuan Jing, Peng Jin, and Lifang He. 2025. Multi-modal isolated sign language recognition based on self-paced learning. *Expert Systems with Applications*, 291:128340.

Siliang Tang, Fan Xue, Jing Wu, Shanghang Wang, and Richang Hong. 2025. Gloss-driven conditional diffusion models for sign language production. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 21(4):105:1–105:17.

Abhishek Tunga, Sravanthi V. Nuthalapati, and Juan Wachs. 2021. Pose-based sign language recognition using gcn and bert. In *Proceedings of the IEEE Winter Conference on Applications of Computer Vision Workshops (WACVW)*, pages 31–40. IEEE.

United Nations. 2023. International day of sign languages. https://www.un.org/en/observances/sign-languages-day. According to the United Nations, there are more than 300 different sign languages used around the world.

Adithya Venugopalan and Rajesh Reghunadhan. 2021. Applying deep neural networks for the automatic recognition of sign language words: A communication aid to deaf agriculturists. *Expert Syst. Appl.*, 185(C).

Harry Walsh, Ozge Mercanoglu Sincan, Ben Saunders, and Richard Bowden. 2023. Gloss alignment using word embeddings. *CoRR*, abs/2308.04248.

Zhen Wang, Dongyuan Li, Renhe Jiang, and Manabu Okumura. 2025. Continuous sign language recognition with multi-scale spatial-temporal feature enhancement. *IEEE Access*, 13:5491–5506.

Kayo Yin and Jesse Read. 2020. Better sign language translation with STMC-transformer. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5975–5989, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Jiangbin Zheng, Yile Wang, Cheng Tan, Siyuan Li, Ge Wang, Jun Xia, Yidong Chen, and Stan Z. Li. 2023. CVT-SLR: Contrastive Visual-Textual Transformation for Sign Language Recognition with Variational Alignment . In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 23141–23150, Los Alamitos, CA, USA. IEEE Computer Society.

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 (ICCV)*, pages 20871–20881.

Finetuning Pre-trained Language Models for Bidirectional Sign Language Gloss to Text Translation

Arshia Kermani, Habib Irani, Vangelis Metsis
Department of Computer Science
Texas State University
San Marcos, TX 78666, USA
{arshia.kermani, habibirani, vmetsis}@txstate.edu

Abstract

Sign Language Translation (SLT) is a crucial technology for fostering communication accessibility for the Deaf and Hard-of-Hearing (DHH) community. A dominant approach in SLT involves a two-stage pipeline: first, transcribing video to sign language glosses, and then translating these glosses into natural text. This second stage, gloss-to-text translation, is a challenging, low-resource machine translation task due to data scarcity and significant syntactic divergence. While prior work has often relied on training translation models from scratch, we show that fine-tuning large, pre-trained language models (PLMs) offers a more effective and data-efficient paradigm. In this work, we conduct a comprehensive bidirectional evaluation of several PLMs (T5, Flan-T5, mBART, and Llama) on this task. We use a collection of popular SLT datasets (RWTH-PHOENIX-14T, SIGNUM, and ASLG-PC12) and evaluate performance using standard machine translation metrics. Our results show that fine-tuned PLMs consistently and significantly outperform Transformer models trained from scratch, establishing new state-of-the-art results. Crucially, our bidirectional analysis reveals a significant performance gap, with Text-to-Gloss translation posing a greater challenge than Gloss-to-Text. We conclude that leveraging the linguistic knowledge of pre-trained models is a superior strategy for gloss translation and provides a more practical foundation for building robust, real-world SLT systems.

1 Introduction

Automatic Sign Language Translation (SLT) is a vital research field focused on bridging communication barriers for the millions of individuals in the Deaf and Hard-of-Hearing (DHH) community (Bragg et al., 2019). The development of robust SLT systems has profound implications for social inclusion, education, and access to essential services, particularly in domains like telehealth where the availability of human interpreters can be limited (Pikoulis et al., 2022).

A dominant paradigm in SLT research decomposes the complex video-to-text translation problem into a more manageable two-stage pipeline (Camgoz et al., 2018). First, a Sign Language Recognition (SLR) module analyzes the input video to generate a sequence of textual labels, known as "glosses." These glosses represent the individual signs in their original signed order. Second, a machine translation module translates this sequence of glosses into a grammatically correct natural language sentence. This paper focuses on this critical second stage: the bidirectional translation between sign language glosses and natural language text (Gloss $\Leftrightarrow$ Text).

The task of translating sign glosses, however, presents unique challenges for Neural Machine Translation (NMT). Glosses are an intermediate representation that simplifies the visual signal into a text-like sequence, but they omit many linguistic features and non-manual markers (e.g., facial expressions). While the lexicon of glosses often overlaps significantly with the target natural language, their syntax follows the grammatical rules of the source sign language, which can be vastly different. For example, American Sign Language (ASL) has a distinct word order and grammatical structure from English (Sandler and Lillo-Martin, 2006). This results in a translation task characterized by high lexical overlap but significant syntactic divergence. Compounding this challenge, the parallel gloss-text corpora available for training are typically small, making this an extremely low-resource NMT problem (Yin and Read, 2020).

Previous neural approaches have demonstrated the viability of the Transformer architecture for this task, but have primarily relied on training models from scratch on these limited datasets (Yin and Read, 2020). We hypothesize that this approach is data-inefficient and that a more effective strategy is

Proceedings of the Workshop on Sign Language Processing (WSLP), pages 77–85

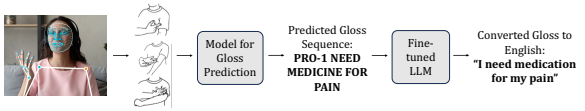

Figure 1: The two-stage Sign Language Translation (SLT) pipeline. This work focuses on the second stage: translating sign language gloss sequences into natural language text and vice-versa. The example shows ASL glosses being translated into an English sentence.

to leverage the vast linguistic knowledge encoded in large, pre-trained language models (PLMs).

Recently, the focus has begun to shift towards fine-tuning LLMs, with work such as (Fayyazsanavi et al., 2024) achieving strong results by developing specialized techniques like novel loss functions and data augmentation for the unidirectional Gloss-to-Text task. Our work complements these efforts by asking a different, foundational question: how do various modern PLMs and architectures perform across the full, bidirectional translation pipeline? By fine-tuning these models, which have already learned the rich grammatical and semantic nuances of the target language from massive text corpora, we can adapt them to the specific task of gloss translation more effectively.

The main contributions of this work are as follows:

- We conduct the first large-scale, systematic comparison of fine-tuning various modern PLMs, including T5, Flan-T5, mBART, and Llama, for the bidirectional gloss-to-text and text-to-gloss translation tasks.
- We empirically demonstrate that our fine-tuning approach significantly outperforms the strong baseline of a Transformer trained from scratch, establishing new state-of-the-art results on the RWTH-PHOENIX-14T, SIGNUM, and ASLG-PC12 benchmarks.
- We provide a comparative analysis of different model architectures (encoder-decoder vs. decoder-only) and pre-training paradigms (e.g., instruction-tuning) to identify the most suitable approaches for this unique translation task.
- We will release our fine-tuned models and experimental code to the research community to foster reproducibility and accelerate future progress in SLT.

2 Related Work

Language Models are increasingly applied across diverse domains, including label quality improvement (Mahjourian and Nguyen, 2025), Sentiment Analysis (Mohammadagha et al., 2025), secure software development practices (Torkamani et al., 2025), and mental health text analysis (Kermani et al., 2025). They have also shown growing potential in advancing translation tasks such as SLT.

2.1 Sign Language Gloss-to-Text Translation

The translation of sign language glosses to natural language text has been an active area of research within SLT. Early approaches often relied on rule-based systems or statistical machine translation (SMT) methods. For instance, the widely-used ASLG-PC12 dataset was itself generated using a rule-based, part-of-speech-based grammar to convert English text into ASL glosses (Othman and Jemni, 2012). However, these methods often struggle to capture the fluency and complexity of natural language.

With the advent of deep learning, the focus shifted to neural machine translation (NMT) models. An initial line of work applied Recurrent Neural Network (RNN) based architectures with attention to the task (Camgoz et al., 2018). A significant step forward was made by (Yin and Read, 2020), who demonstrated the effectiveness of the Transformer architecture (Vaswani et al., 2017) for this task. Their work, which serves as a primary baseline for our study, involved training Transformer models *from scratch* on gloss-text corpora like RWTH-PHOENIX-14T and ASLG-PC12. They showed that this approach could achieve state-of-the-art results, establishing a strong benchmark for neural-based gloss-to-text translation.

The inherent low-resource nature of the problem has also inspired other lines of research, such as data augmentation. For example, (Moryossef et al., 2021) proposed rule-based heuristics to generate pseudo-parallel gloss-text pairs from monolingual text to augment the limited training data. While effective, our work explores a complementary direction: instead of augmenting the data, we propose using more powerful models that are better equipped to learn from sparse data.

Concurrent to our work, (Fayyazsanavi et al., 2024) also explore fine-tuning LLMs for Gloss-to-Text translation. Their primary contributions are the development of tailored data augmenta-

tion techniques (paraphrasing and back-translation) and a novel Semantically Aware Label Smoothing (SALS) loss function to handle gloss ambiguities. Their work demonstrates significant improvements on the PHOENIX-2014T dataset. Our research differs in three key aspects: (1) Scope: We conduct a bidirectional analysis, evaluating both Gloss-to-Text (G2T) and Text-to-Gloss (T2G) tasks, whereas their work focuses solely on G2T. (2) Contribution Type: Our work provides a broad, systematic comparison of multiple PLM families and architectures to establish foundational benchmarks, while their work focuses on developing novel, task-specific techniques for a single model. (3) Evaluation Breadth: We validate our findings across three distinct datasets (RWTH-PHOENIX-14T, SIGNUM, and ASLG-PC12) to ensure generalizability, whereas their experiments are conducted on the PHOENIX-2014T dataset.

2.2 Pre-trained Language Models for NMT

The dominant paradigm in modern Natural Language Processing (NLP) has shifted from training task-specific models from scratch to a pre-train and fine-tune approach (Devlin et al., 2019). Large-scale language models like T5 (Raffel et al., 2020), BART (Lewis et al., 2020), and the Llama family (Aaron Grattafiori, 2024) are first pre-trained on vast, web-scale corpora of unlabeled text. During this phase, they learn rich, general-purpose representations of syntax, semantics, and world knowledge.

This pre-trained knowledge can then be transferred to downstream tasks via a second, much shorter fine-tuning phase on a smaller, labeled dataset. This paradigm has proven exceptionally effective for low-resource NMT (Zoph et al., 2016). Instead of learning the target language's grammar and semantics from a small parallel corpus, the model only needs to learn the *mapping* between the source and target representations. Our work is the first to systematically apply and evaluate this powerful paradigm across a diverse set of modern PLMs for the unique challenges of bidirectional sign language gloss translation.

3 Experimental Setup

We designed a comprehensive experimental setup to rigorously evaluate the performance of fine-tuned pre-trained language models (PLMs) against a from-scratch baseline on bidirectional gloss-text translation. Our setup is standardized across all models to ensure fair and reproducible comparisons. The complete code implementation and benchmarks are made publicly available for reproducibility: `anonymized`.

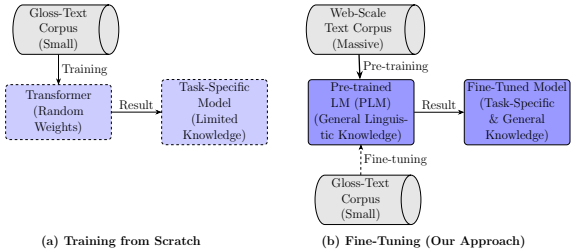

Figure 2: Conceptual comparison of the two training paradigms. (a) The baseline approach trains a Transformer from scratch using only the limited gloss-text corpus. (b) Our approach leverages a large language model pre-trained on vast text corpora and then fine-tunes it on the gloss-text corpus.

3.1 Task Definition

We address two primary translation tasks in this work, treating both as sequence-to-sequence problems:

1. **Gloss-to-Text (G2T):** The model takes a sequence of sign language glosses as input (e.g., `PRO-1 NEED MEDICINE PAIN`) and must generate a grammatically correct sentence in the target natural language (e.g., "I need medicine for the pain.").

2. **Text-to-Gloss (T2G):** The model takes a natural language sentence as input and must generate the corresponding sequence of glosses, reflecting the word order and lexical choices of the target sign language.

3.2 Datasets

We conduct experiments on three publicly available corpora, each with unique characteristics that test different aspects of our models. A summary of the datasets after standard train/dev/test splitting is provided in Table 1.

Dataset	Language Pair	Domain	Train/Dev/Test
PHOENIX	DGS / German	Weather	7,096 / 518 / 642
SIGNUM	DGS / German	Varied	603 / 177 / —
ASLG-PC12	ASL / English	Synthetic	500k / 5k / 5k

Table 1: Overview of datasets. DGS stands for German Sign Language; ASL for American Sign Language. The SIGNUM test set is used for validation.

- **RWTH-PHOENIX-Weather 2014T (Phoenix14T)** (Camgoz et al., 2018) is a widely-used benchmark for continuous sign language research, consisting of German weather forecasts and their corresponding German Sign Language (DGS) gloss transcriptions.
- **SIGNUM** (von Agris and Kraiss, 2010) is a smaller DGS corpus with a more controlled vocabulary, providing a different data condition. We use the original train-test split in our evaluation.
- **ASLG-PC12** (Othman and Jemni, 2012) is a large-scale, synthetically generated corpus of English sentences from Project Gutenberg automatically converted into ASL glosses. While synthetic, its size allows for testing model scalability. We use a 500k-pair subset for training.

3.3 Models and Implementation

We evaluate a from-scratch baseline against four different PLMs.

- **Transformer Baseline (65M params):** For comparison against pre-trained language models (PLMs), we implemented a custom Transformer architecture trained from scratch on the sign language gloss translation tasks.

  The model uses a 4-layer encoder and 4-layer decoder, each with $d_{\text{model}} = 256$ hidden units, 8 attention heads, and a feed-forward dimension of 1024. Positional encodings are added to the token embeddings, and residual connections with dropout (0.2) are applied throughout. To improve parameter efficiency, the output projection layer shares weights with the target embeddings.
- **T5-base (220M params):** A versatile encoder-decoder PLM pre-trained on a text-to-text objective (Raffel et al., 2020).
- **Flan-T5-base (220M params):** An instruction-tuned version of T5, which has been shown to improve zero-shot and few-shot performance on unseen tasks.
- **mBART 50 (610M params):** A multilingual sequence-to-sequence model pre-trained with a denoising objective, which may be particularly suited to handling the ungrammatical nature of glosses (Lewis et al., 2020).
- **Llama 3 8B:** A powerful, modern, decoder-only LLM used to assess the performance of this architectural class (Aaron Grattafiori, 2024).

All models were trained using the HuggingFace Transformers library . For fine-tuning the PLMs, we used the AdamW optimizer with a learning rate of 3×10^{-4} and a batch size of 32. We employed a linear learning rate scheduler with 100 warmup steps and trained for a maximum of 10 epochs with early stopping based on validation loss. For encoder-decoder models, input sequences were prefixed with a task description, e.g., "translate Gloss to English: [GLOSS SEQUENCE]".

3.4 Evaluation Metrics

To provide a comprehensive assessment of translation quality, we use a suite of standard automatic metrics:

- **BLEU** (Papineni et al., 2002): Measures n-gram precision, a standard metric for machine translation quality.
- **ROUGE-L** (Lin, 2004): Measures the longest common subsequence, capturing recall-oriented aspects of the translation.
- **METEOR** (Banerjee and Lavie, 2005): An alignment-based metric that considers synonymy and stemming for a more semantically-aware evaluation.
- **Word Error Rate (WER):** Measures the number of substitutions, deletions, and insertions required to transform the hypothesis into the reference. It is particularly useful for the T2G task where output structure is more rigid.

All scores are computed using the SacreBLEU library (Post, 2018) to ensure consistent and reproducible results. Each experiment was run 10 times with different random initializations.

4 Results and Analysis

We present the results of our experiments on both the Gloss-to-Text (G2T) and Text-to-Gloss (T2G) translation tasks. Our analysis focuses on comparing the performance of fine-tuned pre-trained models against the from-scratch Transformer baseline.

Dataset	Model	BLEU-1 ↑	BLEU-2 ↑	BLEU-3 ↑	BLEU-4 ↑	ROUGE-L ↑	METEOR ↑	WER ↓
RWTH-PHOENIX-14T	Transformer (Baseline)	34.01	23.71	17.24	13.06	34.75	29.51	81.44
	T5-base	48.71	37.16	30.13	22.73	35.04	31.32	34.37
	Flan-T5-base	45.94	32.68	25.29	19.03	33.33	30.06	36.68
	mBART	58.16	45.86	36.52	25.58	46.30	42.26	26.56
	Llama 8B	**63.56**	**53.45**	**43.78**	**29.92**	**53.33**	**49.14**	**21.32**
SIGNUM	Transformer (Baseline)	59.60	47.26	39.76	34.24	61.22	53.09	46.45
	T5-base	71.21	66.09	60.70	52.87	**86.34**	71.64	22.09
	Flan-T5-base	68.12	64.84	59.45	50.72	85.95	73.83	18.45
	mBART	**82.81**	**77.07**	**72.38**	**67.60**	84.80	**79.68**	**17.61**
	Llama 8B	80.56	75.89	70.35	65.78	82.24	78.92	18.23
ASLG-PC12	Transformer (Baseline)	79.28	73.13	67.75	62.81	89.40	80.60	23.41
	T5-base	91.02	81.90	74.82	68.69	89.17	85.63	20.92
	Flan-T5-base	86.38	74.11	64.91	65.40	84.76	82.64	26.81
	mBART	94.55	90.27	86.08	79.58	92.99	88.03	19.31
	Llama 8B	**96.06**	**91.55**	**87.06**	**83.10**	**94.12**	**90.24**	**17.83**

Table 2: Gloss-to-Text (G2T) translation results on RWTH-PHOENIX-14T, SIGNUM, and ASLG-PC12.

Dataset	Model	BLEU-1 ↑	BLEU-2 ↑	BLEU-3 ↑	BLEU-4 ↑	ROUGE-L ↑	METEOR ↑	WER ↓
RWTH-PHOENIX-14T	Transformer (Baseline)	33.47	19.06	11.26	6.98	36.72	25.21	82.46
	T5	**60.30**	27.19	15.17	8.49	49.18	40.85	66.90
	Flan-T5	47.50	25.60	15.00	10.00	45.20	38.50	69.30
	mBART	50.10	30.45	19.20	12.10	44.32	36.51	64.10
	Llama	58.43	**38.32**	**25.54**	**16.81**	**51.25**	**44.63**	**61.45**
SIGNUM	Transformer (Baseline)	61.73	49.65	43.26	**37.51**	64.50	55.31	43.93
	T5	72.15	**56.82**	**43.64**	34.66	68.50	58.90	38.84
	Flan-T5	69.23	54.32	41.50	32.44	65.83	56.22	41.21
	mBART	**75.42**	49.54	37.07	25.43	**70.20**	**61.30**	**37.83**
	Llama	62.53	47.61	35.43	29.74	68.63	56.25	45.72
ASLG-PC12	Transformer (Baseline)	82.21	75.47	68.12	64.12	89.77	81.93	23.10
	T5-base	64.20	44.76	31.34	21.73	76.21	60.66	42.13
	Flan-T5-base	43.41	30.87	23.47	18.51	60.58	52.55	55.68
	mBART	73.76	53.41	38.49	27.68	80.65	67.24	35.80
	Llama	**85.64**	**76.78**	**70.62**	**66.33**	**89.91**	**83.85**	**22.37**

Table 3: Text-to-Gloss (T2G) translation results on RWTH-PHOENIX-14T, SIGNUM, and ASLG-PC12.

4.1 Gloss-to-Text Performance

Table 2 summarizes the performance of all models on the G2T task across the datasets. All scores are averaged over 10 runs. Best scores per metric are in **bold**. The results provide strong evidence for our primary hypothesis.

The results of the gloss-to-text experiments across RWTH-PHOENIX-14T, SIGNUM, and ASLG-PC12 shown in Table 2 demonstrate consistent improvements of pre-trained models over the baseline Transformer trained from scratch. On PHOENIX-14T, all PLMs achieve substantial gains in BLEU-4, with the model mBART reaching 25.58 compared to 13.06 for the baseline. The larger Llama 8B extends this advantage further with 29.92 BLEU-4, underscoring the benefit of large-scale pre-training even in low-resource conditions. On SIGNUM, mBART attains 67.60 BLEU-4, while Llama 8B maintains competitive results. For ASLG-PC12, where the dataset is larger and synthetic, Llama 8B achieves the highest score with 83.10 BLEU-4, indicating that decoder-only models are able to fully exploit large-scale parallel data. Overall, the results confirm that fine-tuning PLMs yields not only higher accuracy but also more fluent and grammatically complete translations across diverse data conditions.

4.2 Text-to-Gloss Performance

For the reverse task of Text-to-Gloss (T2G), we evaluate the models' ability to generate syntactically correct gloss sequences. The results are presented in Table 3.

Compared to gloss-to-text, BLEU scores are generally lower and WER is higher, reflecting the structural difficulty of generating gloss sequences that require word deletion, reordering, and strict adherence to gloss grammar. On PHOENIX-14T, the best-performing model achieves only 16.81 BLEU-4, showing the sharp contrast with gloss-to-text performance. On SIGNUM, pre-trained models again outperform the baseline, with T5 and mBART reaching mid-30 BLEU-4 scores, but still below their gloss-to-text counterparts. On ASLG-PC12, Llama achieves the strongest performance with 66.33 BLEU-4, benefiting from the scale of training data, though this remains substantially

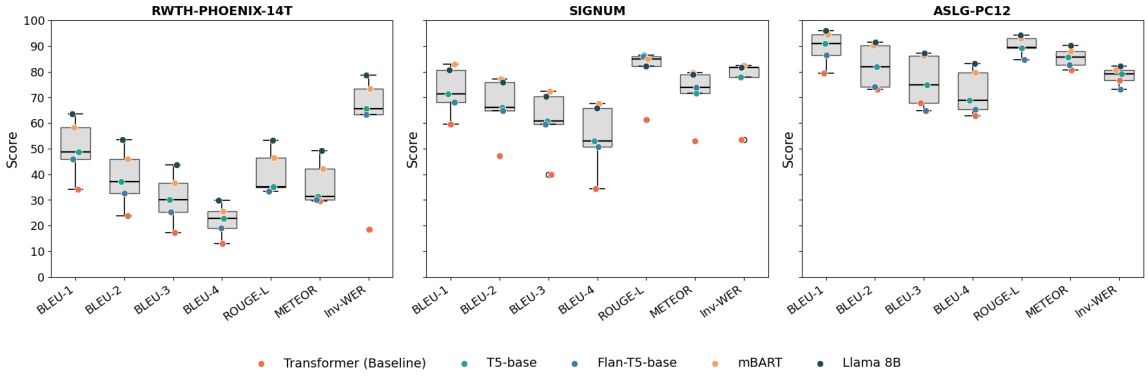

Figure 3: G2T multi-metric comparison across datasets (higher is better; WER inverted).

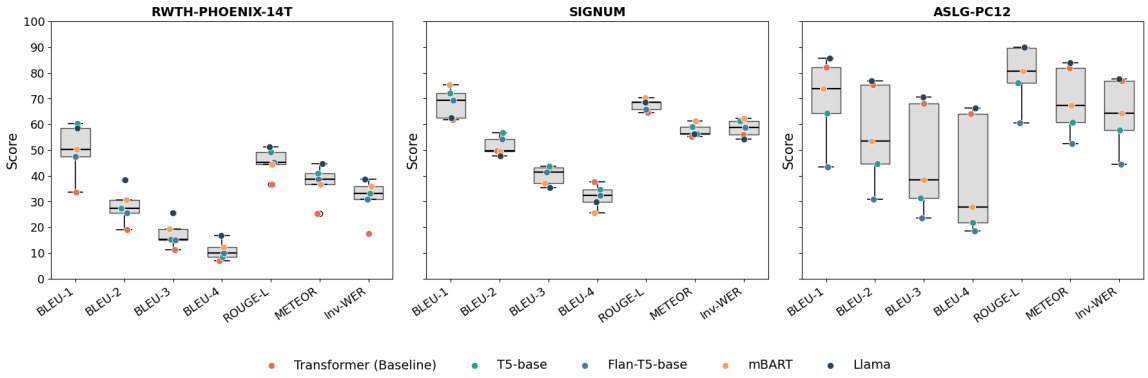

Figure 4: T2G multi-metric comparison across datasets (higher is better; WER inverted).

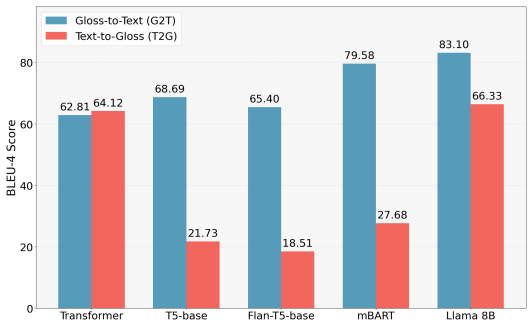

Figure 5: Bidirectional Performance Asymmetry (BLEU-4) on the ASLG-PC12 Dataset.

lower than its gloss-to-text result. These findings confirm the expected asymmetry between the two directions and indicate that text-to-gloss translation is likely to remain a bottleneck in practical bidirectional systems unless further task-specific modeling innovations are introduced. As shown in Figure 5, we observe a clear bidirectional performance asymmetry on the ASLG-PC12 dataset. While G2T achieves higher BLEU, ROUGE-L, and METEOR scores, the corresponding T2G direction results in substantially lower scores across all metrics, highlighting the inherent difficulty of the reverse translation task. A notable exception arises in T2G. As shown in Table3, the from-scratch Transformer achieves the strongest result on SIGNUM and is second only to Llama 8B on ASLG-PC12, outperforming smaller PLMs. This pattern suggests that pre-training for fluent text can conflict with generating non-fluent, rule-like gloss targets.

4.3 Qualitative Analysis

To provide a more intuitive understanding of the performance gap, Table 4 shows example translations from our Flan-T5-base versus the baseline Transformer.

The examples clearly illustrate the advantage of pre-training. The baseline model often produces grammatically incomplete or "telegraphic" text, closely mirroring the structure of the input glosses. In contrast, the fine-tuned Flan-T5 model successfully infers the correct grammatical structure, inserting necessary function words (e.g., "will be", "there is"), handling verb tenses, and producing overall more natural and fluent sentences. This qualitative difference highlights that pre-trained models do not just learn a word-for-word mapping but leverage their internal linguistic models to perform true translation.

Source Text: we are talking about children , the most precious resource that we should protect.
Reference Gloss: X-WE BE TALK ABOUT CHILD , MOST DESC-PRECIOUS RESOURCE THAT X-WE SHOULD PROTECT. **Predicted Gloss:** X-WE BE TALK ABOUT CHILD , MOST FINISH RESOURCE THAT X-WE SHOULD PROTECT.
Source Gloss: IX-1P NOT-YET SEE MOVIE BUT FRIEND RECOMMEND
Reference: I haven't seen the movie yet, but my friend recommended it. **Baseline Output:** I not see movie but friend say good. **Our Output (Flan-T5):** I have not seen the movie yet, but my friend recommended it.

Table 4: Qualitative comparison of example translations from the G2T and T2G tasks. The fine-tuned model generates more fluent and grammatically complete sentences.

5 Discussion

Our experimental results provide compelling evidence that fine-tuning pre-trained language models (PLMs) is a superior strategy to training from scratch for bidirectional gloss translation. Across datasets, PLMs strongly outperform the baseline on G2T, and often on T2G as well—notably Llama 8B on ASLG-PC12—though some PLMs underperform the baseline on ASLG-PC12 T2G. This is demonstrated by concordant gains across BLEU, ROUGE-L, and METEOR metrics, alongside corresponding reductions in WER, as detailed in Tables 2 and 3.

5.1 The Decisive Advantage of Pre-trained Knowledge

A primary finding is the sheer magnitude of the improvement attributable to pre-training. On the challenging PHOENIX-14T dataset (G2T task), even the `T5-base` model achieves a BLEU-4 score of 22.73, a relative gain of roughly 74% over the 13.06 baseline. Larger or more sophisticated PLMs amplify this advantage, with `Llama 8B` reaching an impressive 29.92 BLEU-4.

This performance leap stems from the effective transfer of linguistic knowledge. As the qualitative examples in Table 4 illustrate, PLMs move beyond simple surface-level pattern matching. Compared to the telegraphic and grammatically incomplete outputs of the baseline, fine-tuned models successfully infer correct grammatical structure, inserting necessary function words, handling verb tenses, and producing far more natural and fluent sentences. This demonstrates that the models leverage their vast pre-trained knowledge of language, needing only to learn the mapping from glosses during fine-tuning.

5.2 The Bidirectional Bottleneck: Asymmetry in Translation

A critical insight from our bidirectional analysis is the significant asymmetry between the two translation directions. As shown in Figure 5 and in the detailed results in Table 2 and Table 3, we observe a stark performance asymmetry between the G2T and T2G directions across all models. Text-to-Gloss (T2G) translation is substantially more challenging than Gloss-to-Text (G2T). Across most models and datasets, we observe substantial BLEU reductions (often $\sim$30–60%) when reversing direction.

This difficulty arises because T2G requires the model to generate a syntactically rigid and often non-fluent sequence, which involves precise word deletion and reordering to match sign language grammar. This is an unnatural task for PLMs, whose pre-training objective is biased towards generating fluent, natural language. For instance, while `mBART` achieves a strong 25.58 BLEU-4 on the G2T task for PHOENIX-14T, its performance drops to just 12.10 for T2G. These findings confirm that in any practical bidirectional system, the T2G component is likely to be the primary performance bottleneck if not explicitly optimized with task-specific architectures or objectives. On T2G, a consistent counterexample appears: the from-scratch Transformer surpasses smaller pre-trained models on ASLG-PC12 dataset. This pattern supports a negative-transfer explanation, in which fluency-oriented pre-training conflicts with generating non-fluent, rule-like gloss sequences, while the baseline's neutral inductive bias learns the rigid mapping directly. Only very large models appear to mitigate this interference through additional capacity.

5.3 Architectural and Data Scale Considerations

Our results offer insights into the interplay between model architecture, pre-training objectives, and data conditions. Encoder-decoder models (T5, Flan-T5, mBART) prove highly competitive, especially on the smaller, real-world datasets like PHOENIX-14T and SIGNUM. Notably, the multilingual denoising pre-training of `mBART` appears to provide an advantageous inductive bias for the gloss-to-text mapping.

However, the decoder-only `Llama 8B` model excels where the data scale is largest, achieving the highest scores on the synthetic ASLG-PC12 dataset (83.10 BLEU-4 for G2T). This pattern suggests that while encoder-decoder architectures may be more data-efficient for learning the structured mapping from gloss to text, powerful decoder-only models can surpass them when sufficient parallel data is available to specialize to the task. Furthermore, the mixed results of instruction-tuning (Flan-T5 vs. T5) indicate that generic instruction-following priors do not always translate into downstream advantages for this highly structured translation task.

Finally, dataset characteristics clearly shape outcomes. The high scores on SIGNUM (up to 67.60 BLEU-4 G2T) highlight the effectiveness of PLMs on domain-specific data with controlled vocabularies. In contrast, PHOENIX-14T remains the most realistic and challenging benchmark, where our improvements represent substantial progress towards deployable, real-world systems.

5.4 Limitations and Future Work

Our evaluation relies primarily on automatic metrics and gloss-based representations, which do not capture non-manual markers and may not fully reflect end-user utility. Human evaluation, including DHH raters, should complement automatic metrics. From a modeling standpoint, `Llama 8B` raises compute and memory considerations; future work will investigate parameter-efficient tuning and knowledge distillation. Finally, closing the bidirectional gap likely requires objectives and architectures tailored for T2G (e.g., stronger constraints or structured decoding) and, longer term, integration with end-to-end video models that capture non-manual features.

6 Conclusion

We presented a comprehensive, controlled evaluation of pre-trained language models for bidirectional gloss translation across three distinct datasets. Our findings conclusively show that fine-tuning PLMs consistently and substantially outperforms training Transformers from scratch, with relative BLEU-4 gains on the G2T task ranging from roughly 74% (e.g., 13.06 $\rightarrow$ 22.73 on PHOENIX-14T with `T5-base`) to about 130% (13.06 $\rightarrow$ 29.92 with `Llama 8B`).

Our G2T results establish new state-of-the-art levels on all three benchmarks within our experimental setting, PHOENIX-14T (29.92 BLEU-4), SIGNUM (67.60), and ASLG-PC12 (83.10), demonstrating that transfer learning is a decisive enabler for this low-resource translation problem. The architectural analysis indicates that while encoder-decoder PLMs are highly competitive on smaller datasets, decoder-only LLMs can excel as data scale increases.

At the same time, our bidirectional study underscores a persistent asymmetry: Text-to-Gloss translation remains notably harder than Gloss-to-Text, with $\sim$30–60% BLEU reductions and elevated WER across datasets. Addressing this gap is a key avenue for future research, potentially requiring specialized objectives or constrained decoding.

Practically, these findings lower the barrier to building effective gloss translation systems. Strong models can be obtained via fine-tuning rather than costly training from scratch, making it feasible to extend SLT technology to additional sign languages and domains. We will release our code and fine-tuned checkpoints to support reproducibility and accelerate progress toward inclusive, deployable communication tools for the DHH community.

References

Abhimanyu Dubey Aaron Grattafiori. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Satanjeev Banerjee and Alon Lavie. 2005. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan.

Danielle Bragg, Oscar Koller, Mary Bellard, Larwan Berke, Patrick Boudreault, Annelies Braffort, Naomi Caselli, Matt Huenerfauth, Hernisa Kacorri, Tessa

Verhoef, and 1 others. 2019. Sign language recognition, generation, and translation: An interdisciplinary perspective. In *The 21st International ACM SIGACCESS Conference on Computers and Accessibility*, pages 16–31.

Necati Cihan Camgoz, Simon Hadfield, Oscar Koller, Hermann Ney, and Richard Bowden. 2018. Neural sign language translation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7784–7793.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota.

Pooya Fayyazsanavi, Antonios Anastasopoulos, and Jana Kosecka. 2024. Gloss2Text: Sign language gloss translation using LLMs and semantically aware label smoothing. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 16162–16171, Miami, Florida, USA. Association for Computational Linguistics.

Arshia Kermani, Veronica Perez-Rosas, and Vangelis Metsis. 2025. A systematic evaluation of llm strategies for mental health text analysis: Fine-tuning vs. prompt engineering vs. rag. *Preprint*, arXiv:2503.24307.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2020. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain.

Nazanin Mahjourian and Vinh Nguyen. 2025. Sanitizing manufacturing dataset labels using vision-language models. *arXiv preprint arXiv:2506.23465*.

Mohsen Mohammadagha, Israel Tshitenge, and Ifetilayo Adebambo. 2025. State-of-the-art machine learning techniques in sentiment analysis for social media: L'état de l'art des techniques d'apprentissage automatique en analyse de sentiment pour les médias sociaux.

Amit Moryossef, Kayo Yin, Graham Neubig, and Yoav Goldberg. 2021. Data augmentation for sign language gloss translation. In *Proceedings of the 1st International Workshop on Automatic Translation for Signed and Spoken Languages (AT4SSL)*, pages 1–11, Virtual.

Achraf Othman and Mohamed Jemni. 2012. English-ASL gloss parallel corpus 2012: ASLG-PC12. In *Proceedings of the LREC2012 5th Workshop on the Representation and Processing of Sign Languages: Interactions between Corpus and Lexicon*, pages 151–154, Istanbul, Turkey. European Language Resources Association (ELRA).

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA.

E.V. Pikoulis, A. Bifis, M. Trigka, C. Constantinopoulos, and D. Kosmopoulos. 2022. Context-aware automatic sign language video transcription in psychiatric interviews. *Sensors*, 22(7).

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

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

Mohammad Jalili Torkamani, Negin Mahmoudi, and Kiana Kiashemshaki. 2025. Llm-driven adaptive 6g-ready wireless body area networks: Survey and framework. *arXiv preprint arXiv:2508.08535*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008.

Ulrich von Agris and Karl-Friedrich Kraiss. 2010. SIGNUM database: Video corpus for signer-independent continuous sign language recognition. In *Proceedings of the LREC2010 4th Workshop on the Representation and Processing of Sign Languages: Corpora and Sign Language Technologies*, pages 243–246, Valletta, Malta. European Language Resources Association (ELRA).

Kayo Yin and Jesse Read. 2020. Attention is all you sign: Sign language translation with transformers. In *Sign Language Recognition, Translation and Production (SLRTP) Workshop - Extended Abstracts*.

Barret Zoph, Deniz Yuret, Jonathan May, and Kevin Knight. 2016. Transfer learning for low-resource neural machine translation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1568–1575, Austin, Texas.

Author Index