Open-Domain Sign Language Translation Learned from Online Video

Bowen Shi
TTI-Chicago
bshi@ttic.edu

Diane Brentari
Univeristy of Chicago
dbrentari@uchicago.edu

Greg Shakhnarovich
TTI-Chicago
greg@ttic.edu

Karen Livescu
TTI-Chicago
klivescu@ttic.edu

Abstract

Existing work on sign language translation—that is, translation from sign language videos into sentences in a written language—has focused mainly on (1) data collected in a controlled environment or (2) data in a specific domain, which limits the applicability to real-world settings. In this paper, we introduce OpenASL, a large-scale American Sign Language (ASL) - English dataset collected from online video sites (e.g., YouTube). OpenASL contains 288 hours of ASL videos in multiple domains from over 200 signers and is the largest publicly available ASL translation dataset to date. To tackle the challenges of sign language translation in realistic settings and without glosses, we propose a set of techniques including sign search as a pretext task for pre-training and fusion of mouthing and handshape features. The proposed techniques produce consistent and large improvements in translation quality, over baseline models based on prior work.[1]

1 Introduction

Sign language, a type of visual language that conveys meaning through gestures, is the most widely used form of linguistic communication among deaf and hard of hearing people. According to the World Health Organization, over 5% of the world's population (∼430 million people) suffer from disabling hearing loss.[2] Automatic sign language processing can facilitate the daily activities of deaf people and make artificial intelligence technologies more accessible to deaf users. For example, such techniques would allow deaf users to interact with intelligent virtual assistants using sign language and would support automatic interpretation between sign languages and spoken languages. Interest in sign language research has recently been growing in the computer vision (CV) (Bragg et al., 2019; Adaloglou et al., 2021; Rastgoo et al., 2021) and natural language processing (NLP) communities (Shterionov, 2021; Yin et al., 2021)

In this paper, we study sign language translation (SLT),[3] the task of translating continuous signing video into written language sentences. Unlike other sign language processing tasks such as sign spotting (Buehler et al., 2009) or continuous sign language recognition (sign-to-gloss transcription) (Dreuw et al., 2007), SLT has not been studied until recently (Camgoz et al., 2018) and is still restricted to specific domains (e.g., weather forecasts (Camgoz et al., 2018), emergency situations (Ko et al., 2019)), characterized by small vocabulary size and lack of visual variability. The lack of large-scale translation datasets in the wild is a central challenge in developing SLT technologies serving real-world use cases.

In terms of translation modeling, most existing SLT approaches (Camgoz et al., 2018, 2020a,a; Zhou et al., 2021; Yin et al., 2021; Gan et al., 2021; Chen et al., 2022) rely on glosses, which are a transliteration system for sign language. Annotating sign language videos with glosses is expensive and hard to scale up. Building effective methods for SLT without glosses is an under-studied challenge.

In this work, we introduce OpenASL, a large-scale ASL-English translation dataset. OpenASL has 288 hours of real-world ASL videos from over 200 signers, making it the largest ASL-English translation dataset to date. OpenASL covers multiple domains drawn from a mix of news and VLOGs. To handle challenges in SLT modeling without glosses, we propose a set of techniques including pre-training with spotted signs and fusion of multiple local visual features, which improves over existing SLT baselines by a large margin.

[1] Our data and code are publicly available at https://github.com/chevalierNoir/OpenASL.

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

[3] Note that "SLT" is often used as an abbreviation for "spoken language technology". In this paper we use it exclusively for "sign language translation" following other recent work.

Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pages 6365 - 6379
December 7-11, 2022

Figure 1: Typical image frames in OpenASL.

2 Related Work

2.1 Datasets for SLT

There has been a large body of work collecting sign language corpora in general. Here we mainly focus on video-based datasets that can be used for SLT (see Table 1), which contain paired continuous signing videos and sentences in a written language.

Dataset	Lang	vocab	# hours	# signers	source
Phoenix-2014T (Camgoz et al., 2018)	DGS	3K	11	9	TV
KETI (Ko et al., 2019)	KSL	419	28	14	Lab
CSL Daily (Zhou et al., 2021)	CSL	2K	23	10	Lab
SWISSTXT-Weather (Camgoz et al., 2021)	DSGS	1K	1	-	TV
SWISSTXT-News (Camgoz et al., 2021)	DSGS	10K	10	-	TV
VRT-News (Camgoz et al., 2021)	VGT	7K	9	-	TV
BOBSL (Albanie et al., 2021)	BSL	78K	1467	39	TV
Purdue RVL-SLLL (Wilbur et al., 2006)	ASL	104	-	14	Lab
Boston 104 (Dreuw et al., 2007)	ASL	103	< 1	3	Lab
How2Sign (Duarte et al., 2021)	ASL	16K	80	11	Lab
OpenASL (Ours)	ASL	33K	288	$\sim$220	Web

Table 1: Statistics of existing SLT datasets. Example images from these datasets can be found in the Appendix (Section A.1). The number of signers in OpenASL is approximate, since we cannot determine the identity of the signers for some of the videos.

Most of the early datasets (Wilbur et al., 2006; Dreuw et al., 2007) were collected in a studio-like environment, where native signers are asked to sign some given content. Recording conditions such as lighting, background and camera perspectives are carefully controlled in such datasets. These corpora provide valuable resources, but do not account for real-world conditions, which has been noted as a limiting factor in recent work on sign language (Yin et al., 2021). Moreover, the high cost of data collection also makes studio-based datasets hard to scale up.

With the advancement of computer vision techniques, there is increasing attention on collecting real-life SLT datasets. Many such datasets (Camgoz et al., 2018, 2021; Albanie et al., 2021) are drawn from TV programs accompanied by sign language interpretation. Despite being highly realistic compared to studio datasets, they are generally limited to a specific domain. For example, the popular Phoenix-2014T DGS-German benchmark contains signed German weather forecasts and includes only 11 hours of signing videos from 9 signers. The largest real-world sign language corpus we are aware of is BOBSL (Albanie et al., 2021), which consists of 1,467 hours of BBC broadcasts from 39 signers interpreted into British Sign Language (BSL). However, access to the dataset is heavily restricted.

Unlike prior datasets, OpenASL contains a mix of spontaneous and (presumably) interpreted sign language videos. It is collected from online video sites and thus contains a diverse set of signers and domains. In addition, the annotations we provide are fully accessible to the public.

2.2 Methods for SLT

Direct translation from videos of continuous signing is practically appealing and has received growing interest recently. Ko et al. (2019) study translation of common Korean sign language sentences (in video) that may be used in an emergency scenario. In this specific domain with restricted vocabulary size (419 words), the model can achieve BLEU-4 score higher than 60. In a larger-vocabulary setting, Camgoz et al. (2018) study translation of German sign language weather forecast videos under various labeling setups. In particular, one of their main findings is the drastic improvement achieved when using gloss labels in training an SLT model. It is hypothesized in (Camgoz et al., 2020b) that glosses, as an intermediate representation of sign language, can provide more direct guidance in

learning sign language video representation. Therefore, most followup work (Camgoz et al., 2020b; Chen et al., 2022; Zhou et al., 2021; Yin and Read, 2020) largely relies on gloss sequences in training.

Given the high cost of gloss labeling, conducting gloss-free SLT is practically appealing but introduces modeling challenges. Glosses, which are monotonically aligned to the video, provide stronger supervision than text in written language translation and facilitate learning of a more effective video representation. On the Phoenix-2014T benchmark, a model trained without glosses (Camgoz et al., 2018) falls behind its counterpart with glosses by over 10.0 (absolute) BLEU-4 score (Camgoz et al., 2020b). Improving translation in real-world sign language video without gloss labels is the modeling focus of this paper. There is little prior work addressing SLT without glosses. In a gloss-free setting, Li et al. (2020b) study the use of segmental structure in translation to boost translation performance. Orbay and Akarun (2020) incorporate handshape features into the translation model. In this paper, we consider sign spotting pre-training and fusion of multiple local features for gloss-free translation.

A typical SLT model is composed of a visual encoder and a sequence model. The visual encoder maps input video into intermediate visual features. In (Camgoz et al., 2018), a sign recognizer CNN-LSTM-HMM trained with gloss labels was used to extract image features. The continuous sign recognizer was replaced by a CTC-based model in (Camgoz et al., 2020b). In addition to RGB-based images, pose is also used (Ko et al., 2019; Gan et al., 2021) as a complementary input modality, which is commonly encoded by graph convolutional neural networks. The sequence models in SLT are usually based on standard sequence-to-sequence models in machine translation with either recurrent neural networks (Camgoz et al., 2018) or transformers (Camgoz et al., 2020b; Yin and Read, 2020; Chen et al., 2022) as the backbone.

2.3 Other related work

Two key components of our proposed approach are searching for spotted signs from video-sentence pairs and fusing multiple local visual features. There has been a substantial amount of prior work (Buehler et al., 2009; Albanie et al., 2020; Varol et al., 2021; Momeni et al., 2020; Shi et al., 2022a) devoted to spotting signs in real-world sign language videos. In contrast to this prior work where sign search is the end goal, here we treat sign spotting as a pretext task in the context of SLT.

The use of multi-channel visual features has also been previously explored for multiple tasks, including sign spotting (Albanie et al., 2020) and continuous sign language recognition (Koller et al., 2020). Specifically for SLT, Camgoz et al. (2020a) learn a multi-channel translation model by including mouthing and handshape features. However, these local modules are trained with in-domain data whose labels are inferred from glosses, which makes it inapplicable for gloss-free translation. In contrast, we utilize models pre-trained on out-of-domain data to extract local features and study the effect of feature transfer to translation.

3 The OpenASL Dataset

Our videos are collected from video websites, mainly YouTube. A large portion of our data consists of ASL news, which come primarily from the YouTube channels `TheDailyMoth` and `Sign1News`. We download all videos with English captions in these two channels through June 2021. The rest of the dataset is collected from short YouTube videos uploaded by the National Association of the Deaf (`NAD`). Those videos are mostly in the form of sign VLOGs of various types including announcements, daily tips, and short conversations.

The raw video is divided into roughly sentence-sized clips based on the associated subtitles. Specifically, we split the transcript into sentences with the NLTK[4] sentence segmentation tool and retrieve the corresponding (video clip, sentence) pairs. This procedure produces 98,417 translation pairs in total, with 33,549 unique words. Figure 4 shows the distribution of sentence length in our data. We randomly select 966 and 975 translation pairs from our data as validation and test sets respectively.

The annotation of the validation and test sets is manually verified. Specifically, the English translation and time boundaries of each video clip are proofread and corrected as needed by professional ASL interpreters. Each annotator views the video clip and is given the original English sentence from the subtitle for reference. The annotator marks the corrected beginning and end of the sentence, and provides a corrected English translation if needed as well as the corresponding gloss sequence. During translation of each sentence, the annotator has

[4] `https://www.nltk.org/`

access to the whole video in case the context is needed for accurate translation.

Figure 2 shows the distribution of several properties in our dataset. Note that these are not ground-truth labels, but rather approximate labels as perceived by an annotator. The goal is to give an idea of the degree of diversity in the data, not to provide ground-truth metadata for the dataset. The label "other" covers a variety of categories, including examples where the annotator is unsure of the label and examples that contain multiple signers.

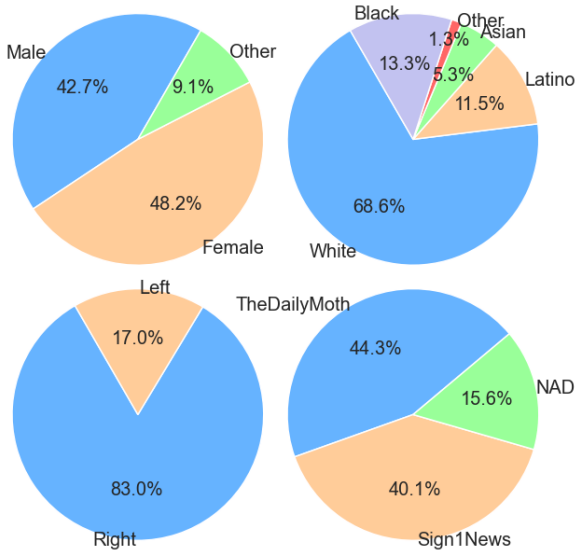

Figure 2: Distribution of several properties in video clips in a subset of OpenASL (top left: gender, top right: race, bottom left: handedness, bottom right: sources).

One feature of our data is the use of subtitles associated with the video as the English translation, thus saving effort on human annotation. Subtitled videos have also been employed in prior work (Camgoz et al., 2021; Albanie et al., 2021) for constructing sign language datasets. As prior work has mostly focused on interpreted signing videos where content originally in the spoken language is interpreted into sign language, the subtitles used there are naturally aligned to the audio instead of the signing stream. As is shown in (Bull et al., 2021), there exists a large time boundary shift between the two. The videos used in OpenASL are "self-generated" rather than interpreted, so the English subtitles are already aligned to the video accurately. As can be seen from Figure 3, the sentence alignment in the subtitles is of overall high quality (usually less than 2 second time shifts, although a small percentage ($< 5\%$) are larger).

We measure the degree of agreement between the original and corrected translations using BLEU

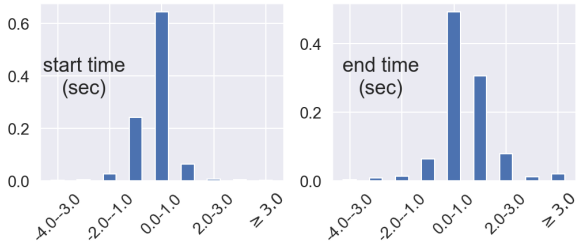

Figure 3: Empirical distribution of alignment errors (in seconds) in a manually checked subset of our data Left: start time, right: end time.

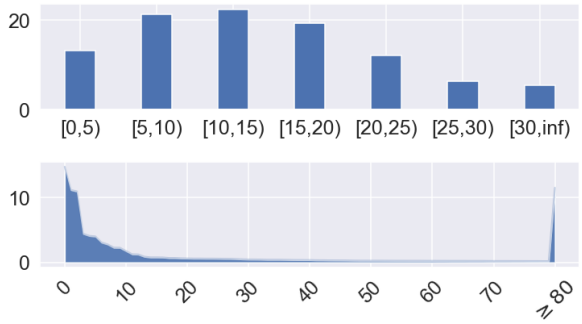

Figure 4: Empirical distribution of sentence length (upper) and percentage of sequences per signer (lower). The sentence length is in # words per sentence. The signer identity is obtained from metadata of the original video. The sequences where the identity is unknown are not counted.

score (Papineni et al., 2002). The original translation achieves 81.0 BLEU-4 score when it is compared against the corrected one. The high agreement in translation, as well as the small alignment error from Figure 3, shows the overall high quality of the subtitles. Thus to save annotation effort, we do not proofread the training data.

4 Model and pre-training for gloss-free translation

A translation model maps a sequence of T image frames $\mathbf{I}_{1:T}$ to a sequence of n words $w_{1:n}$. In the most recent state-of-the-art approaches (Camgoz et al., 2020b; Li et al., 2020b) for SLT, a visual encoder M_g^v first maps $\mathbf{I}_{1:T}$ to a visual feature sequence $\mathbf{f}_{1:T'}$, and a transformer-based sequence-to-sequence model decodes $\mathbf{f}_{1:T_g}$ into $w_{1:n}$. Our approach is based on the same overall architecture (see Figure 6). We further incorporate several techniques for pre-training and local feature modeling, described next.

4.1 Sign spotting pre-training

For M_g^v, we use an inflated 3D convnet (I3D) developed for action recognition (Carreira and Zisserman, 2017). Ideally, the visual encoder should

capture signing-related visual cues (arm movement, handshape, and so on). However, the translated sentence in the target language may not provide sufficiently direct guidance for learning the visual representation, as is observed in prior work (Camgoz et al., 2018).

To alleviate this issue, we pre-train the I3D network on relevant tasks that provide more direct supervision for the convolutional layers than full translation. Specifically, we pre-train I3D for the task of isolated sign recognition on WLASL (Li et al., 2020a), a large-scale isolated ASL sign dataset. Empirically, we observe considerable gains from isolated sign recognition pre-training (see Section A.7).

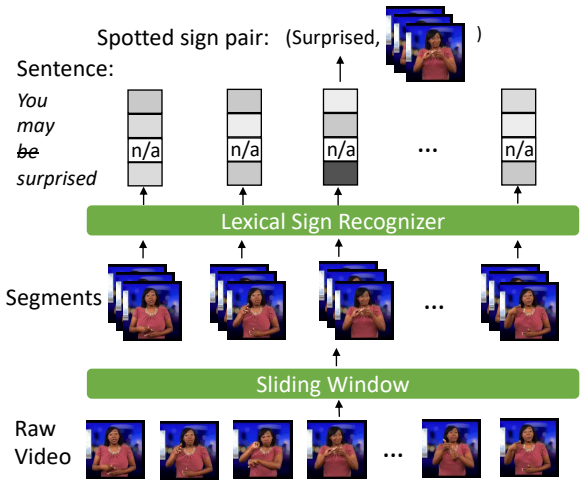

Figure 5: Sign spotting. For illustration purposes, only lexical sign search is shown. Fingerspelling sign search works similarly.

Despite the aforementioned benefits, the isolated sign recognition pre-training causes two potential problems for the translation model. First, there is substantial domain mismatch between isolated signs and the continuous signing data used in translation. The coarticulation in a continuous signing stream is not reflected in isolated sign datasets. In addition, the isolated sign videos are collected from sources such as online lexica, which usually have simpler visual backgrounds and less motion blur than real-world signing video. Finally, existing isolated sign datasets mainly consist of lexical signs and have few instances of fingerspelling. Fingerspelling is used frequently in day-to-day signing and many important content words are commonly fingerspelled. Features related to fingerspelling may not be encoded well due to the lack of fingerspelling-specific pre-training data.

To mitigate the above issues, we propose to search for signs from the signing video (see Figure 5). The searched signs are used to pre-train the visual backbone for translation. The search relies on a lexical sign recognizer M^l and a fingerspelling recognizer M^f, which map a video segment into a word probability vector $\mathbf{p} \in [0,1]^V$ (V: vocabulary size) and a letter sequence $\mathbf{c}_{1:|\mathbf{c}|}$. Given a translation video-sentence pair $(\mathbf{I}_{1:T}, w_{1:n})$, the task is to spot **lexical** and **fingerspelled** signs $\mathcal{P} = \{(\mathbf{I}_{s_i:t_i}, w_i)\}_{1 \le i \le |\mathcal{P}|}$, where the w_i are selected from $w_{1:n}$. The search process is described briefly below (see Section A.3 for details).

We generate a list of candidate time intervals for lexical signs and fingerspelling signs respectively with a sliding window approach and a fingerspelling detector M^d. For each interval, we infer its word probability $\mathbf{p}$ for lexical signs or word hypothesis (i.e., a sequence of characters) $\hat{w}_f$ for fingerspelling. We assign a word from the translated sentence to the target interval if the word probability p_w is high or its edit distance with the fingerspelling hypothesis is low.

Unlike the isolated sign dataset, the spotted signs are sampled from the same data used for translation training. Additionally, the detected fingerspelling signs should also enhance the model's ability to transcribe signs that are fingerspelled.

4.2 Hand and mouth ROI encoding

In sign language, meaning is usually conveyed via a combination of multiple elements including motion of the arms, fingers, mouth, and eyebrows. The corresponding local regions in the image frame play an important role in distinguishing signs. For instance, SENATE and COMMITTEE have the same place of articulation and movement; the difference lies only in the handshape. Furthermore, mouthing (i.e., mouth movement) is commonly used for adjectives or adverbs to add descriptive meaning (Nadolske and Rosenstock, 2008).

Our model's visual backbone does not explicitly employ local visual cues. In principle, learned global features can include sufficient information about the important local cues, but this may require a very large amount of training data. However, it may be helpful to guide the translation model more explicitly by learning local discriminative features using external tasks.

Here we focus on learning features for two local visual modalities: handshape and mouthing. To extract handshape features, we train a fin-

gerspelling recognizer[5] on two large-scale fingerspelling datasets (Shi et al., 2019) and use it to extract features for the hand region of interest (ROI). ASL fingerspelling includes many handshapres that are also used in lexical signs. Recognizing fingerspelling requires distinguishing quick hand motions and nuance in finger positions. The features are extracted for both hands in each frame and are concatenated before feeding into the translation model. We denote the hand feature sequence as $\mathbf{f}_{1:T}^{(h)}$, where T is the video length in frames.

For mouthing, we use an external English lipreading model[6] (Shi et al., 2022b) to extract features $\mathbf{f}_{1:T}^{(m)}$ from the lip regions of the signer. Although mouthing in ASL is not used to directly "say" words, we assume there is sufficient shared lip motion between speaking and ASL mouthing.

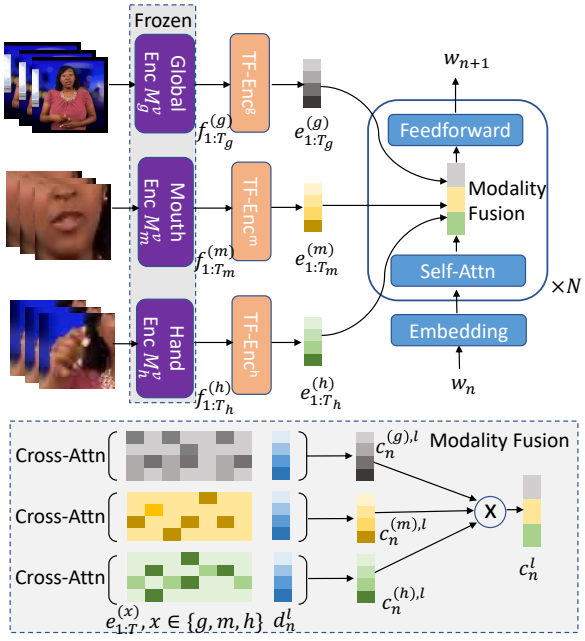

Figure 6: Multi-modal sign language translation.

4.3 Fusion and sequence modeling

Given the global/handshape/mouthing feature sequences $\mathbf{f}_{1:T}^{(g)}$/$\mathbf{f}_{1:T}^{(m)}$/$\mathbf{f}_{1:T}^{(h)}$, the sequence model maps them to text $w_{1:n}$, as illustrated in Figure 6. Since we have multiple feature sequences each with its own sequential properties, we adopt three independent transformer (Vaswani et al., 2017) encoders for the three types of features:

$$\mathbf{e}_{1:T_x}^{(x)} = \text{TF-Enc}^{(x)}(\mathbf{f}_{1:T_x}^{(x)}), x \in \{g, m, h\}$$

where TF-Enc$^{(g)}$, TF-Enc$^{(m)}$, TF-Enc$^{(h)}$ denote the transformer encoders for global, mouthing and hand feature sequences respectively.

For decoding, we use a single transformer decoder that takes all three encoder representations as input. At decoding timestep n, we compute modality-specific context vectors:

$$\mathbf{c}_n^{(x),l} = \text{Cross-Attn}^{(x)}(\mathbf{d}_n^l, \mathbf{e}_{1:T_x}^{(x)}), x \in \{g, m, h\}$$

where Cross-Attn$^{(g)}$, Cross-Attn$^{(m)}$ and Cross-Attn$^{(h)}$ are cross-attention layers (Vaswani et al., 2017) for global/mouthing/hand features. We concatenate the context vectors from the three modalities to form the decoder context vector $\mathbf{c}_n^l = [\mathbf{c}_n^{(g),l}, \mathbf{c}_n^{(m),l}, \mathbf{c}_n^{(h),l}]$, which is passed to a feedforward layer and then the next layer. The final layer output is then passed to a linear projection, followed by a final softmax to produce a probability vector over words in the vocabulary.

5 Experiments

5.1 Setup

For evaluation, we report BLEU-{1,2,3,4} (Papineni et al., 2002) and ROUGE (Lin, 2004) scores, as in prior work on SLT (Camgoz et al., 2018; Ko et al., 2019; Camgoz et al., 2021). As there is only one English sentence as reference for evaluation, we also report BLEURT (Sellam et al., 2020) score, a metric that provides a measure of semantic similarity between the prediction and ground truth. Implementation details can be found in the appendix (Section A.4).

5.2 Main Results

The performance of our proposed approach is shown in Table 2. We compare it to two baseline approaches adapted from prior work. Conv-GRU, which uses ImageNet-pretrained AlexNet as a visual backbone, is an RNN-based sequence-to-sequence model proposed by Camgoz et al. (2018) for sign language translation without glosses. I3D-transformer is a similar model architecture to ours, but it uses only global visual features and the CNN backbone is pre-trained only on the WLASL isolated sign recognition task. See Section A.5 in the appendix for the performance of these two baseline

[5]The implementation is based on (Shi et al., 2019) but extended with VGG-19 encoder. See Section A.4 for implementation details.

[6]We use the publicly available model of (Shi et al., 2022b) without any additional training.

	DEV						TEST					
Models	ROUGE	BLEU-1	BLEU-2	BLEU-3	BLEU-4	BLEURT	ROUGE	BLEU-1	BLEU-2	BLEU-3	BLEU-4	BLEURT
Conv-GRU (Camgoz et al., 2018)†	16.25	16.72	8.95	6.31	4.82	25.36	16.10	16.11	8.85	6.18	4.58	25.65
I3D-transformer	18.88	18.26	10.26	7.17	5.60	29.17	18.64	18.31	10.15	7.19	5.66	28.82
Ours	**20.43**	**20.10**	**11.81**	**8.43**	**6.57**	**31.22**	**21.02**	**20.92**	**12.08**	**8.59**	**6.72**	**31.09**

Table 2: Translation performance of baseline models and our proposed approach. †: based on the public code released by the authors.

methods on the popular DGS-German benchmark Phoenix-2014T.

From the results in Table 2, we observe: (1) Conv-GRU has the worst performance among the three models. One key difference lies in the data used to pre-train the visual encoder: Conv-GRU is pre-trained on ImageNet while the latter two are pre-trained on sign language-specific data. There are, of course, also differences in the model architecture and training pipeline. To isolate the effect of sign language-specific pre-training, we compare I3D-transformer pre-trained with different types of data, and find that isolated sign pre-training leads to consistent gains. See Section A.7 in the Appendix for details. (2) Our proposed approach achieves the best performance. On average, the relative gain over I3D transformer is $\sim 15\%$ in ROUGE and BLEU scores. This demonstrates the effect of including spotted signs in visual backbone pre-training and of incorporating the multiple local visual features. (3) The performance measured by BLEU, ROUGE and BLEURT scores are consistent for different models.

Despite the improvement over baseline approaches, our model's performance is still quite poor. We show some qualitative examples of translated sentence in Section A.9 of the Appendix. The low performance of current translation models has also been observed in prior work on other sign languages (Albanie et al., 2021; Camgoz et al., 2021), highlighting the challenging nature of sign language translation.

In the next sections, we analyze the effects of the main components in our model. For the purpose of these analyses, we report BLEU and ROUGE scores on the validation set.

5.3 Ablation Study

Effect of sign spotting pre-training In Table 3, we compare the performance of models with different pre-training data: WLASL only, WLASL + spotted lexical signs. For both models, the visual backbone is I3D and we do not incorporate local visual features.

Model	ROUGE	BLEU-1	BLEU-2	BLEU-3	BLEU-4
iso only	18.88	18.26	10.26	7.17	5.60
+spotted	**19.65**	**19.72**	**11.18**	**8.56**	**6.51**

Table 3: Effect of sign spotting pre-training (iso: isolated sign, spot: spotted signs) on the development set.

The results show that sign search consistently improves performance. Compared to training with WLASL only, including lexical sign and fingerspelling spotting produces $\sim 10\%$ relative improvements, averaged across metrics. We attribute these gains to the adaptation of I3D to our translation data, which includes coarticulation and visual challenges that the isolated sign data lacks.

An alternative strategy could be to fine-tune the visual backbone on our translation data. However, this strategy downgrades translation performance by a large margin (see Section A.6 for details). Qualitatively, the spotted sign pairs are high-quality in general (see Section A.10).

Effect of local feature incorporation Table 4 compares models without local visual features, and with both mouthing and handshape features. All models are pre-trained with spotted signs. Overall the incorporation of local features produces 5% gains in different metrics. The gain is relatively larger in BLEU scores of lower orders (e.g., BLEU-1). See Section A.10 for qualitative examples of improved translation when using mouthing features.

Model	ROUGE	BLEU-1	BLEU-2	BLEU-3	BLEU-4
global	19.65	19.72	11.08	8.06	6.30
+ local	**20.43**	**20.10**	**11.81**	**8.43**	**6.57**

Table 4: Effect of incorporating local visual features.

5.4 Analysis

For a more detailed analysis of our model, we measure its performance on different evaluation subsets, divided by several criteria.

Duplicate vs. non-duplicate Certain sentences appear frequently in our dataset, which leads to duplicated sentences appearing in both training and evaluation. The duplicate sentences account for 10.9% (105 out of 967) of the dev set and 10.6% (103 out of 976) of the test set. Most of these are sentences that are used frequently in the news, such as "Hello", "Thank you", and "See you tomorrow".

Our model translates videos associated with duplicate sentences with high accuracy (see Figure 7). On the duplicate subset, the BLEU-4 score is 72.91. Duplicates tend to be short clips, which are easy for the model to memorize. In contrast, the BLEU-4 score on the non-duplicate subset is only 4.09.

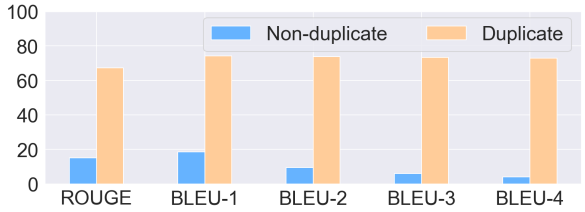

Figure 7: Comparison of translation performance on duplicate and non-duplicate sentences. Duplicate sentences are ones that appear in the training set.

News vs. VLOGs Our data are collected from online sign language resources from two categories: news (`Sign1News` and `TheDailyMoth`) and VLOGs (`NAD`). The two sources differ in multiple aspects, including visual conditions and linguistic content. In the dev set, videos from these two categories account for 63.6%/36.4% of sentences respectively. We break the performance down according to the source (see Figure 8). To avoid the impact of duplicate sentences, we also perform this comparison separately on non-duplicate sentences.

Our model performs better on scripted news videos regardless of whether the duplicates are included or not, which may be attributed to multiple factors. On the one hand, the data from NAD VLOGs contain a larger set of signers than the news videos. The variability in signing among different signers increases the difficulty of translation. NAD VLOG videos also have higher visual variance in terms of image resolution and background diversity. It is also possible that the news videos are more likely to be scripted beforehand while the VLOG videos are more likely to be spontaneous. Spontaneous ASL videos are expected to be more challenging to translate than scripted videos.

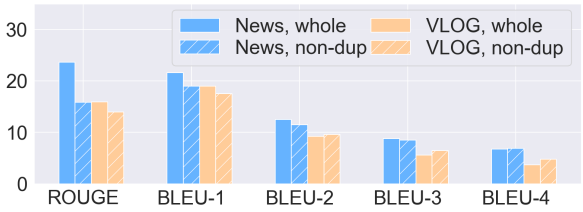

Figure 8: Translation performance for ASL news and VLOGs.

Fingerspelling vs. non-fingerspelling In our dev set, 54.7% of the clips have at least one fingerspelled word. Our model's translation performance on the fingerspelling-free subset is overall higher than on clips with fingerspelling (BLEU-4: 7.74 vs. 6.33). We expect that proper nouns, typically fingerspelled in ASL, are difficult to translate for our model. A more detailed analysis can be found in Section A.8.

6 Conclusion

Our work advances sign language translation "in the wild" (i.e., directly translating real-world sign language videos into written language) both (1) by introducing a new large-scale ASL-English translation dataset, OpenASL, and (2) by developing methods for improved translation in the absence of glosses and in the presence of visually challenging data. OpenASL is the largest publicly available ASL translation dataset to date. By using online captioned ASL videos, we have been able to collect a large amount of high-quality and well-aligned translation pairs (as verified by professional translators) that represent a wide range of signers, domains, and visual conditions. Our translation approach, which combines pre-training via sign spotting and multiple types of local features, outperforms alternative methods from prior work by a large margin. Nevertheless, the overall translation quality for sign language videos, in both our work and prior work, is significantly lower than that of machine translation for written languages. There is therefore much room for future improvement, and we hope that OpenASL will enable additional progress on this task.

Limitations

Despite being the largest ASL translation dataset to date, OpenASL is still of relatively small scale compared to commonly used translation corpora for written languages. Due to resource constraints, we provide only one English translation for each video for the time being. Future work may augment the dataset with multiple English translations per video. In addition, although we strive to collect a diverse dataset, we do not have ground-truth labels for signer gender, race, and handedness, so we cannot be certain about the distribution of these properties in OpenASL. In terms of methodology, the proposed lexical sign search relies on the availability of isolated sign data. Moreover, our approach may have difficulty in handling ASL signs that do not have an equivalent word in English (e.g., PAH!). Finally, the overall quality of English translations produced by our model is still very low, which highlights the challenging nature of open-domain sign language translation.

Ethics Statement

The copyright for all videos in our dataset belongs to their respective owners. The video URL, timestamps and English translations are released under a Creative Commons BY-NC-ND 4.0 license. Our data are collected from online video sites, and the signers may not be representative of the general deaf or ASL signing population. Please be aware of unintended racial or gender bias caused by this fact. Finally, the translation model proposed in this paper still has low performance and hence is unable to serve as an alternative to human interpreters in real-life scenarios.

References

Nikolaos Adaloglou, Theocharis Chatzis, Ilias Papastratis, Andreas Stergioulas, Georgios Papadopoulos, Vassia Zacharopoulou, George Xydopoulos, Klimis Antzakas, Dimitris Papazachariou, and Petros Daras. 2021. A comprehensive study on deep learning-based methods for sign language recognition. *IEEE Transactions on Multimedia*, PP:1–1.

Samuel Albanie, Gül Varol, Liliane Momeni, Triantafyllos Afouras, Joon Son Chung, Neil Fox, and Andrew Zisserman. 2020. BSL-1K: Scaling up co-articulated sign language recognition using mouthing cues. In *ECCV*.

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.

Danielle Bragg, Oscar Koller, Mary Bellard, Larwan Berke, Patrick Boudreault, Annelies Braffort, Naomi K. Caselli, Matt Huenerfauth, Hernisa Kacorri, Tessa Verhoef, Christian Vogler, and Meredith Ringel Morris. 2019. Sign language recognition, generation, and translation: An interdisciplinary perspective. *The 21st International ACM SIGACCESS Conference on Computers and Accessibility*.

Patrick Buehler, Andrew Zisserman, and Mark Everingham. 2009. Learning sign language by watching tv (using weakly aligned subtitles). In *CVPR*, pages 2961–2968.

Hannah Bull, Triantafyllos Afouras, Gül Varol, Samuel Albanie, Liliane Momeni, and Andrew Zisserman. 2021. Aligning subtitles in sign language videos. In *CVPR*.

N. C. Camgoz, B. Saunders, G. Rochette, M. Giovanelli, G. Inches, R. Nachtrab-Ribback, and R. Bowden. 2021. Content4all open research sign language translation datasets. *ArXiv*, abs/2105.02351.

Necati Camgoz, Oscar Koller, Simon Hadfield, and Richard Bowden. 2020a. Multi-channel transformers for multi-articulatory sign language translation. In *ECCV*, pages 301–319.

Necati Cihan Camgoz, Simon Hadfield, Oscar Koller, Hermann Ney, and Richard Bowden. 2018. Neural sign language translation. In *CVPR*, pages 7784–7793.

Necati Cihan Camgoz, Oscar Koller, Simon Hadfield, and Richard Bowden. 2020b. Sign language transformers: Joint end-to-end sign language recognition and translation. In *CVPR*.

João Carreira and Andrew Zisserman. 2017. Quo vadis, action recognition? a new model and the kinetics dataset. *CVPR*, pages 4724–4733.

Yutong Chen, Fangyun Wei, Xiao Sun, Zhirong Wu, and Stephen Lin. 2022. A simple multi-modality transfer learning baseline for sign language translation. In *CVPR*.

Philippe Dreuw, David Rybach, Thomas Deselaers, Morteza Zahedi, and Hermann Ney. 2007. Speech recognition techniques for a sign language recognition system. In *Interspeech*, volume 1, pages 2513–2516.

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 *CVPR*.

Gunnar Farnebäck. 2003. Two-frame motion estimation based on polynomial expansion. In *SCIA*.

Shiwei Gan, Yafeng Yin, Zhiwei Jiang, Linfu Xie, and Sanglu Lu. 2021. Skeleton-aware neural sign language translation. *Proceedings of the 29th ACM International Conference on Multimedia*.

Alex Graves, Santiago Fernández, Faustino J. Gomez, and Jürgen Schmidhuber. 2006. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *ICML*.

Davis E. King. 2009. Dlib-ml: A machine learning toolkit. *J. Mach. Learn. Res.*, 10:1755–1758.

Diederik P. Kingma and Jimmy Ba. 2015. ADAM: A method for stochastic optimization. *CoRR*, abs/1412.6980.

S. Ko, C. Kim, H. Jung, and C. Cho. 2019. Neural sign language translation based on human keypoint estimation. *Applied Sciences*, 9.

Oscar Koller, Necati Cihan Camgoz, Hermann Ney, and Richard Bowden. 2020. Weakly supervised learning with multi-stream cnn-lstm-hmms to discover sequential parallelism in sign language videos. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(9):2306–2320.

Dongxu Li, Cristian Rodriguez-Opazo, Xin Yu, and Hongdong Li. 2020a. Word-level deep sign language recognition from video: A new large-scale dataset and methods comparison. In *WACV*, pages 1448–1458.

Dongxu Li, Chenchen Xu, Xin Yu, Kaihao Zhang, Ben Swift, Hanna Suominen, and Hongdong Li. 2020b. TSPNet: Hierarchical feature learning via temporal semantic pyramid for sign language translation. In *NeurIPS*, volume abs/2010.05468.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *ACL*.

Liliane Momeni, Gül Varol, Samuel Albanie, Triantafyllos Afouras, and Andrew Zisserman. 2020. Watch, read and lookup: learning to spot signs from multiple supervisors. In *ACCV*.

Marie A. Nadolske and Rachel Rosenstock. 2008. *Occurrence of mouthings in American Sign Language: A preliminary study*, pages 35–62. De Gruyter Mouton.

Alptekin Orbay and Lale Akarun. 2020. Neural sign language translation by learning tokenization. In *IEEE International Conference on Automatic Face and Gesture Recognition*, pages 222–228.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *ACL*.

Razieh Rastgoo, Kourosh Kiani, and Sergio Escalera. 2021. Sign language recognition: A deep survey. *Expert Systems with Applications*, 164:113794.

Thibault Sellam, Dipanjan Das, and Ankur P Parikh. 2020. BLEURT: Learning robust metrics for text generation. In *ACL*.

Bowen Shi, Diane Brentari, Greg Shakhnarovich, and Karen Livescu. 2021. Fingerspelling detection in american sign language. In *CVPR*, pages 4164–4173.

Bowen Shi, Diane Brentari, Greg Shakhnarovich, and Karen Livescu. 2022a. Searching for fingerspelled content in american sign language. In *ACL*.

Bowen Shi, Wei-Ning Hsu, Kushal Lakhotia, and Abdelrahman Mohamed. 2022b. Learning audio-visual speech representation by masked multimodal cluster prediction. In *ICLR*.

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. In *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. In *SLT*, pages 145–152.

Dimitar Shterionov. 2021. Proceedings of the 1st international workshop on automatic translation for signed and spoken languages (at4ssl). In *Proceedings of the 1st International Workshop on Automatic Translation for Signed and Spoken Languages (AT4SSL)*.

Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. 2019. Deep high-resolution representation learning for human pose estimation. In *CVPR*, pages 5686–5696.

Gül Varol, Liliane Momeni, Samuel Albanie, Triantafyllos Afouras, and Andrew Zisserman. 2021. Read and attend: Temporal localisation in sign language videos. In *CVPR*, pages 16852–16861.

Ashish Vaswani, Noam M. Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NeurIPS*.

R. B. Wilbur et al. 2006. Purdue RVL-SLLL American Sign Language Database. Technical report, School of Electrical and Computer Engineering, Purdue University.

K. Yin, A. Moryossef, J. Hochgesang, Y. Goldberg, and M. Alikhani. 2021. Including signed languages in natural language processing. In *ACL*.

Kayo Yin and Jesse Read. 2020. Better sign language translation with STMC-Transformer. In *COLING*.

Hao Zhou, Wen gang Zhou, Weizhen Qi, Junfu Pu, and Houqiang Li. 2021. Improving sign language translation with monolingual data by sign back-translation. In *CVPR*, pages 1316–1325.

A Appendix

A.1 Existing datasets

Figure 9 shows typical image frames in existing SLT datasets. Overall, existing SLT data are collected in controlled environments with relatively little visual variability (e.g., background, lighting).

Figure 9: Image frames from existing SLT datasets.

A.2 Instructions for meta annotation

For the meta annotation, the annotators were provided the complete videos, and were required to annotate the following information: (1) Number of signers appearing at any time in the video ("one" or "multi" for multiple signers) (2) Name of signer, if it appears in the video description. If you cannot find it, mark this field "UNK" (3) Handedness (left or right; if multiple signers, mark this field "multi") (4) Perceived gender ("male", "female", or "other/unknown") (5) Perceived race (white/caucasian (including latino/hispanic), black/African American, south Asian, east Asian, other/unknown) (6) Perceived age group (child, young adult, middle aged, older adult) (7) Perceived ASL proficiency (native or near-native, high proficiency, low proficiency).

A.3 Sign Search

The search process for lexical and fingerspelling signs is detailed in Algorithm 1.

Algorithm 1: Sign Search

Data: Translation dataset $\mathcal{D}^t$
Model: isolated sign recognizer M^l, fingerspelling recognizer M^f, fingerspelling detector M^d
Hyperparameters: lexical/fingerspelling threshold δ_l/δ_f
Output: Spotted lexical and fingerspelling sign dataset $\mathcal{D}^s$
Function `SearchSign`($\mathcal{D}^t$, $M^{\{l,d,f\}}$, $\delta_{\{l,f\}}$)**:**

- $\mathcal{D}^s \leftarrow \emptyset$;
- **for** $(\mathbf{I}_{1:T}, w_{1:L}) \in \mathcal{D}^t$ **do**
  - Sliding windows $\Omega_s = \{(s_i, e_i)\}_{1:|\Omega_s|}$;
  - **for** $(s, e) \in \Omega_s$ **do**
    - Let $\mathbf{p} \leftarrow M^l(\mathbf{I}_{s:e})$ be probability vector;
    - Let $\tilde{w}_{1:L'} \leftarrow$ the subset of $w_{1:L}$ within the vocabulary of M^l;
    - Let $\mathbf{q} \leftarrow (p_{\tilde{w}_1}, p_{\tilde{w}_2}, ..., p_{\tilde{w}_{L'}})$;
    - Let $k \leftarrow \text{argmax}\{\mathbf{q}\}$;
    - **if** $q_k > \delta_l$ **then**
      - $\mathcal{D}^s \leftarrow \mathcal{D}^s \cup \{(\mathbf{I}_{s:e}, \tilde{w}_k)\}$
    - **end**
  - **end**
  - Fingerspelling proposals $\Omega_f = \{(s_i, e_i)\}_{1:|\Omega_f|} = M^d(\mathbf{I}_{1:T})$;
  - **for** $(s, e) \in \Omega_f$ **do**
    - Word hypothesis $\hat{w}_f = M^f(\mathbf{I}_{s:e})$;
    - Accuracies $\mathbf{y} = (A(\hat{w}_f, w_1), ..., A(\hat{w}_f, w_L))$, A: letter accuracy function;
    - Let $k \leftarrow \text{argmax}\{\mathbf{y}\}$;
    - **if** $y_k > \delta_l$ **then**
      - $\mathcal{D}^s \leftarrow \mathcal{D}^s \cup \{(\mathbf{I}_{s:e}, w_k)\}$
    - **end**
  - **end**
- **end**
- **return** $\mathcal{D}^s$;

A.4 Implementation details

Preprocessing For training, we use the time boundaries in the associated video caption to segment raw

videos into short clips. We extend the time boundaries of each video clip by 0.5 second at both the beginning and the end to reduce the proportion of potential missing frames caused by misalignment between subtitle and signing video. Each video clip is cropped to include only the signing region of the target signer. Specifically, we employ the DLIB face detector (King, 2009) to detect the face of the target signer and crop an ROI centered on the face which is 4 times the size of the original bounding box. In case there are multiple faces detected, we employ a simple heuristic to determine the target face track (tracks with the highest optical flow (Farnebäck, 2003) magnitude). The selected ROI is resized to 224×224. We use words as output units and keep words appearing at least twice in the training set in the vocabulary (21,475 words).

Visual Backbone For global visual feature extraction, we adopt I3D network (Carreira and Zisserman, 2017) as our backbone. The I3D, pretrained on Kinetics-400 (Carreira and Zisserman, 2017) is further fine-tuned on WLASL (Li et al., 2020a), an isolated ASL sign dataset with 14,289 isolated training videos of 2000 distinct ASL signs. The isolated sign recognizer achieves 42.6% accuracy on the WLASL test set.

For hand feature extraction, we train a fingerspelling recognizer on the ChicagoFSWild (Shi et al., 2018) and ChicagoFSWild+ (Shi et al., 2019) datasets, which include 61,536 ASL fingerspelling sequences. The recognizer is based on a Conv-LSTM architecture (Shi et al., 2021) consisting of the first 11 conv layers of VGG-19 followed by a one-layer Bi-LSTM with 512 hidden units per direction. The model is trained with CTC loss (Graves et al., 2006) and achieves 64.5% letter accuracy on the ChicagoFSWild test set. In order to extract hand features on our data, we use the HRNet whole-body pose estimator (Sun et al., 2019) to detect the hands of the signer and extract features in the hand ROI. Features for left and right hands are concatenated before feeding into the translation model.

To obtain mouthing feature, we employ AV-HuBERT (Shi et al., 2022b), a state-of-the-art lip reading model for English. The mouth ROI, cropped and resized to 96×96 based on the facial landmarks detected with DLIB facial keypoint detector (King, 2009), are fed into the lip reading model for feature extraction.

Sign Search To search lexical signs, we run inference with the aforementioned I3D isolated sign recognizer on 32-frame windows. The window is swept across the whole video clip at a stride of 8 frames. To search fingerspelling, we use the off-the-shelf fingerspelling detector (Shi et al., 2021) trained on raw ASL videos of ChicagoFSWild+, which has achieved 0.448 AP@0.5. The aforementioned fingerspelling recognizer is used for searching fingerspelling signs. We keep proposals with confidence score higher than 0.5. The thresholds δ_l/δ_f are tuned to be 0.6/0.2 respectively. The total number of signs detected from our translation data is 32,602. We combine WLASL and the spotted signs for pre-training I3D (see section 5.3). The model is trained with SGD for 50 epochs at batch size of 8. The learning rate and momentum of SGD are 0.01 and 0.9 respectively. The learning rate is reduced to half at epoch 20 and 40.

Sequence Model The visual backbones are frozen in training translation model. Both transformer encoder and decoder have 2 layers with 512 hidden dimension and 2048 feedforward dimension. The model is trained with Adam (Kingma and Ba, 2015) for 14K iterations at batch size of 64. The learning rate is linearly increased to 0.001 for the first 2K iterations and then decayed to 0 in the later iterations. At test time, we use beam search for decoding. The beam width and length penalty are tuned on the validation set.

Real-time performance Although real-time performance is not a goal of this work, we note that the whole proposed system (including all preprocessing such as mouth/hand ROI estimation) processes ∼25 frames per second on average for a sign language video from scratch on one RTX A4000 GPU.

A.5 Baseline performance on Phoenix-14T

Table 5 shows the performance of the two baseline approaches on Phoenix-14T. In contrast to results on OpenASL, I3D-transformer does not generally outperform Conv-GRU, which is probably due to the linguistic discrepancy between the isolated sign data used to pre-train I3D (WLASL: ASL) and the translation data (Phoenix-14T: DGS).

Model	ROUGE	BLEU-1	BLEU-2	BLEU-3	BLEU-4
Conv-GRU (Camgoz et al., 2018)	**31.80**	**32.24**	**19.03**	12.83	9.58
I3D-transformer	27.92	26.88	18.18	**13.42**	**10.66**

Table 5: Performance of baseline approaches on Phoenix-14T test set.

A.6 Does fine-tuning the visual encoder help?

By default, the visual backbone is frozen in training the translation model. Table 6 compares pre-training and fine-tuning I3D visual encoder for translation. Fine-tuning visual backbone deteriorates the model performance by a large margin. This probably suggests that the proposed benchmark is in a low-resource regime, which does not have enough data for full fine-tuning. We hypothesize that the paired text does not provide strong supervision to learn visual encoder, thus leading to performance degradation. Training a fully end-to-end SLT model potentially requires much larger amount of paired data.

Fine-tuning?	ROUGE	BLEU-1	BLEU-2	BLEU-3	BLEU-4
✗	**18.88**	**18.26**	**10.26**	**7.17**	**5.60**
✓	18.91	16.95	9.12	5.87	4.38

Table 6: Comparison between fine-tuning and freezing visual backbone.

A.7 Which pre-training data to use?

To show the effect of isolated sign pre-training, we compare I3D pre-trained with Kinetics-400 and WLASL in Table 7. Kinetics-400 (Carreira and Zisserman, 2017) is a large-scale action recognition dataset including over 306,245 video clips from 400 action categories, while WLASL contains 14,289 clips from 2,000 ASL signs. Though the size of WLASL is one order of magnitude smaller, using WLASL for pre-training outperforms pre-training with Kinetics only by a large margin. Utilizing isolated sign data, despite its amount being scarce, greatly boosts the visual representation that further benefits translation.

Data	ROUGE	BLEU-1	BLEU-2	BLEU-3	BLEU-4
K	13.63	12.25	5.07	3.14	2.32
K→W	**18.88**	**18.26**	**10.26**	**7.17**	**5.60**

Table 7: Effect of pre-training data (K: Kinetics-400 (Carreira and Zisserman, 2017)),W: WLASL (Li et al., 2020a)).

A.8 Fingerspelling vs. non-fingerspelling

Fingerspelling is an important component in real-world ASL videos. We measure the performance on the subsets with and without fingerspelling respectively. According to Figure 10, the translation quality in non-fingerspelling subsets is consistently higher than the other part. Typical fingerspelled words which our model fails to translate are either proper nouns with low frequency in training (e.g., SCHMIDT, WHALEY), or long words (e.g., MASSACHUSETTS, SALT LAKE CITY). Though the visual backbone of our translation model is pre-trained with fingerspelling sequences, transcribing the fingerspelling segment(s) is still problematic. As our model is based on whole word, it is incapable of translating words unseen during training. Thus proper nouns, typically fingerspelled in ASL, are difficult to translate by our model. In practice, we observed many fingerspelled words are simply replaced with <UNK>. How to improve translation for ASL videos with fingerspelling requires more research.

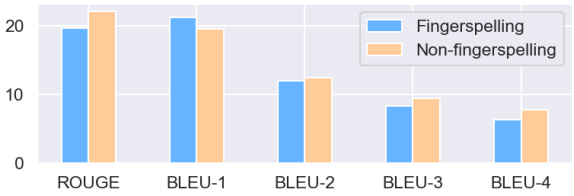

Figure 10: Comparison of translation performance between subsets with and without fingerspelling.

A.9 Translation examples

We randomly select 15 examples from dev set and compare the model prediction against the reference (see Table 8). The exactly correct translations are mostly short and commonly used sentences in daily communication (e.g., thank you). For longer and more complex sentences, the model frequently fails to capture their general meaning though some keywords can be predicted correctly.

A.10 Visualization

The spotted lexical signs and fingerspelling sequences are shown in figure 11. Note that those examples are randomly selected. The spotted signs are mostly accurate. Below are our main observations. First, in lexical sign spotting, the target clip often includes a partial (or whole) segment from adjacent signs. For instance, the third clip of UNIVERSITY has an extra sign of GALLAUDET. This is due to the fixed window size we use for lexical sign search. 2. False positives occur especially when two signs are of similar appearance. The second clip of BEFORE, which has a similar body posture to BEFORE, is a pointing sign indicating that one thing is happening prior to something else. 3. Using a sophisticated fingerspelling detector enables us to spot fingerspelling sequences more precisely compared to lexical signs.

#1	
Ref:	thank you
Hyp:	thank you
#2	
Ref:	come on
Hyp:	come on
#3	
Ref:	now i've come this far and it 's a different team
Hyp:	how do you feel about it
#4	
Ref:	i was there from the beginning to the end and time went by fast
Hyp:	the students were thrilled by this
#5	
Ref:	i'm here at nad's 50th wow
Hyp:	the nad has been <unk> for many years
#6	
Ref:	i entered the yap 2018 competition and won
Hyp:	the competition was started with ideas
#7	
Ref:	you can check out their kickstarter in the link below
Hyp:	you can watch the conversation at lake county
#8	
Ref:	that is one thing i found interesting and wanted to share with you today
Hyp:	i also am the president of the jr. nad conference here
#9	
Ref:	those are the different types of bills
Hyp:	schools have switched to teaching students
#10	
Ref:	dry january has picked up in popularity since it began in 2012
Hyp:	krispy kreme is bringing back its original playstation in 2016
#11	
Ref:	we will be happy to respond give you support and listen to your concerns
Hyp:	please review and submit your time passion and support this important issue
#12	
Ref:	there were videos posted on the internet that showed a person walking on the grass completely engulfed in flames
Hyp:	a video shows the officer walking up to his shoulder and before he was shot
#13	
Ref:	and people would become carpenters laborers mechanics plowers and farmers
Hyp:	the next year 1880 the nad was established in the first operation 30 of the house in 2015
#14	
Ref:	for nad youth programs related information please contact us via facebook at the nad youth programs or email us through
Hyp:	you can contact us through our website where you can check our facebook page online at <unk>
#15	
Ref:	last week suspects gregory mcmichael and his son travis were arrested and charged with felony murder and aggravated assault
Hyp:	last week a black man named <unk> <unk> was arrested and charged with felony murder and aggravated assault

Table 8: Qualitative translation examples. (Ref: reference, Hyp: prediction from our SLT model). Note the examples are randomly chosen without cherrypicking.

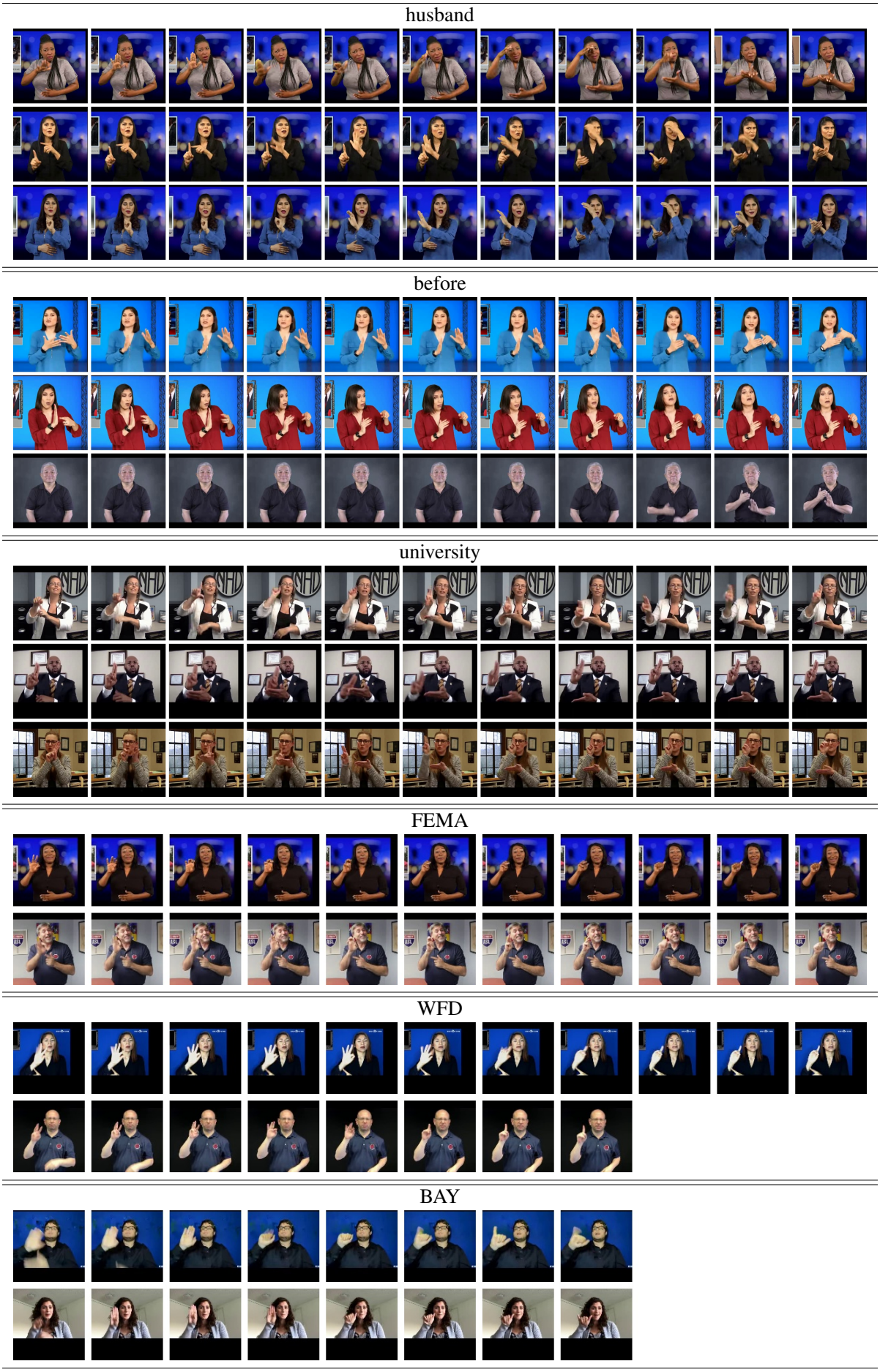

Figure 11: Qualitative examples of signs spotted by our model (FEMA, WFD and BAY are fingerspelled).