
Sign Language Understanding using Multimodal Learning



Liliane Momeni
Keble College
University of Oxford

A thesis submitted for the degree of
Doctor of Philosophy

Hilary 2024

To Mum and Dad, my heroes forever.

*Scientists investigate that which
already is.*

*Engineers create that which has
never been.*

Albert Einstein

Abstract

Sign languages are visual-spatial languages, representing the natural means of communication for deaf communities. Despite recent advancements in vision and language tasks, automatic sign language understanding remains largely unsolved. A key obstacle to making progress is the scarcity of appropriate training data. In this thesis, we aim to address this challenge.

First, we focus on visual keyword spotting (KWS) – the task of determining whether and when a keyword is spoken in a video – and leverage the fact that signers sometimes simultaneously mouth the word they sign. We initially propose a convolutional KWS architecture inspired by object detection methods, trained on data of talkings faces. We then improve the cross-modal interaction between the video and keyword representations by leveraging Transformers. Subsequently, we use the KWS model out-of-domain on signer mouthings as a means to localize signs: we automatically annotate hundreds of thousands of signs in readily available sign language interpreted TV data, by leveraging weakly-aligned subtitles to provide query words.

Second, to move beyond mouthings which are sparse, we propose different sign spotting approaches to automatically annotate signs in the continuous interpreted signing: (i) using visual sign language dictionaries in a multiple instance learning framework, (ii) exploiting the attention mechanism of a Transformer trained on a video-to-text sequence prediction task, (iii) pseudo-labelling from a strong sign recognition model, (iv) leveraging in-domain exemplars from previous approaches and sign representation similarities. All four approaches leverage the weakly-aligned subtitles and increase the vocabulary and density of automatic sign annotations. As a result, we obtain a large-scale, diverse, supervised dataset, and facilitate the learning of strong sign representations.

Third, we explore sign language tasks that entail predicting sequences of signs: fingerspelling and continuous sign language recognition (CSLR). For fingerspelling, we propose a weakly-supervised approach to detect and recognise sequences of letters, with a multiple-hypothesis loss function to learn from noisy supervision. For CSLR, we design a multi-task model capable of also performing sign language retrieval, and demonstrate promising results in large-vocabulary settings.

Finally, we explore obtaining stronger supervision from weak signals for a more general task, beyond the domain of sign language. Specifically, our focus shifts to verb understanding in video-language models – an important ability for modeling interactions among people, objects and the environment through space and time. For this task, we introduce a verb-focused contrastive framework consisting of two components: (i) leveraging pretrained large language models to create hard negatives for cross-modal contrastive learning; and (ii) enforcing a fine-grained alignment loss.

Keywords – video understanding, deep learning, vision & language, multimodal, sign language

This thesis is submitted to the Department of Engineering Science, The University of Oxford, in fulfilment of the requirements for the degree of Doctor of Philosophy. This thesis is entirely my own work, and except where otherwise stated, describes my own research.

Liliane Momeni, April 2024.

Acknowledgement

First and foremost, I offer my deepest gratitude to my supervisor Andrew Zisserman, to whom I will forever be indebted. Thank you for your trust, patience and guidance since the very beginning. Your wisdom and infectious enthusiasm made this thesis possible. I am extremely fortunate to have had you by my side throughout this journey.

To my sign language family, working with you has been an absolute privilege and joy. I could not have hoped for a better team. To Triantafyllos Afouras, whose endless support from my very first paper and mentorship instilled in me the confidence to approach any research problem. To Samuel Albanie and Gül Varol, for teaching me how to conduct thorough research, and meet any deadline, no matter its imminence. And to Hannah Bull, KR Prajwal and Charles Raude, for showing me that research is best done with others.

To all the remarkable members of VGG, being part of this group has been a true honor. I am profoundly grateful to each of you for cultivating such a warm, supportive and collaborative research environment. In particular, to Tengda Han, Chuhan Zhang, Max Bain, Yuki Asano, Mandela Patrick, Vicky Kalogeiton, Tom Jakab, Prannay Kaul, Rhydian Windsor, Andrew Brown and Guanqi Zhan for the special times we spent together in Oxford. To Abhishek Dutta and Ashish Thandavan, who are the pillars of our research endeavors.

To my internship collaborators and mentors, thank you for expanding my horizons. At Google, to Mathilde Caron, Arsha Nagrani and Cordelia Schmid for their generous time and energy. To Ahmet Iscen, Anurag Arnab, Paul Hongsuck Seo and Antoine Yang for making my time in Grenoble so memorable. At Meta, to Aishwarya Kamath, Nicolas Carion, and Ross Girshick, for always sharing invaluable insights.

To my dear friends, who have made this journey all the more meaningful. To Walter Goodwin, David Winter, Sophie Aldred, Eli Bernstein, Lewis Bixer, for standing by me in Oxford through thick and thin. To Aimée Buchler, Samuel Bates, Olga Iturri and Iona Tait, for their unwavering friendship and loyalty.

To Sagar Vaze, for always believing in me. Your kindness and support knows no bounds, thank you.

Finally and most importantly, to my family, for their unconditional love and being my constant source of inspiration. In particular, to my parents, Pardis and Firouz, for being

my heroes, role models and best friends. Thank you for the countless sacrifices you have made, for protecting and guiding me at every turn in life. To my brother, Alex, and sister, Roxane, for lighting up my life and for encouraging me to be ambitious. To my late aunts, Parnian and Parand, and grandparents, Guiti and Babajun, who have shaped the person I am today. Thank you for being proud of me always, while keeping me grounded. The memories we shared together have and will forever give me strength.

Contents

1	Introduction	11
1.1	Motivation	13
1.2	Key Ideas	16
1.2.1	Multimodal Learning	16
1.2.2	Learning with Weak Supervision	17
1.3	Thesis Outline and Contributions	18
1.3.1	Keyword Spotting in Sign Language	19
1.3.2	Approaches for Sign Spotting	19
1.3.3	Sequence recognition in Sign Language	21
1.3.4	Enhancing verb representations	22
1.4	Publications	23
I	Keyword Spotting in Sign Language	25
2	Seeing wake words: Audio-Visual Keyword Spotting	26
2.1	Introduction	27
2.2	Related Work	29
2.3	KWS-Net	31
2.4	Experiments	34
2.5	Results	37
2.5.1	Visual-only KWS-Net	37
2.5.2	Audio-visual KWS-Net	40
2.5.3	Extension to other languages: French and German	41
2.6	Conclusion	42
3	Visual Keyword Spotting with Attention	43
3.1	Introduction	44
3.2	Related work	46

3.3	Visual KWS with Attention	48
3.3.1	The Transpotter Architecture	48
3.3.2	Training	50
3.3.3	Discussion	51
3.3.4	Implementation details	51
3.4	Experiments	52
3.4.1	Datasets and Evaluation Protocol	52
3.4.2	Comparison to baselines	53
3.4.3	Architecture ablations	54
3.4.4	Transpotter performance analysis	55
3.5	Mouthing Spotting in Sign Language videos	57
3.6	Conclusion	59
4	BSL-1K: Scaling up co-articulated sign language recognition using mouthing cues	60
4.1	Introduction	62
4.2	Related Work	64
4.3	Learning Sign Recognition with Automatic Labels	67
4.3.1	Finding probable signing windows in public broadcast footage	67
4.3.2	Precise sign localisation through visual keyword spotting . .	68
4.3.3	BSL-1K dataset construction and validation	70
4.4	Models and Implementation Details	72
4.4.1	Visual keyword spotting model	72
4.4.2	Sign recognition model	72
4.4.3	Video pose distillation	73
4.5	Experiments	74
4.5.1	Ablations for the sign recognition model	74
4.5.2	Benchmarking sign recognition and sign spotting	76
4.5.3	Comparison with the state of the art on ASL benchmarks .	78
4.6	Conclusion	79

II	Approaches for Sign Spotting	80
5	Watch, read and lookup: learning to spot signs from multiple supervisors	81
5.1	Introduction	82
5.2	Related Work	85
5.3	Learning Sign Spotting Embeddings from Multiple Supervisors . . .	88
5.3.1	Integrating Cues through Multiple Instance Learning	89
5.3.2	Implementation details	91
5.4	Experiments	93
5.4.1	Datasets	93
5.4.2	Evaluation Protocols	95
5.4.3	Ablation Study	96
5.4.4	Applications	98
5.5	Conclusions	100
6	Read and Attend: Temporal Localisation in Sign Language Videos	101
6.1	Introduction	102
6.2	Related Work	105
6.3	Sign Localisation with Attention	108
6.4	Experiments	111
6.4.1	Data and evaluation protocols	111
6.4.2	Comparison of video features	113
6.4.3	Mining training examples through attention	114
6.4.4	Comparison with other automatic annotations	116
6.4.5	Qualitative analysis	119
6.4.6	Discussion	119
6.5	Conclusions	120
7	Automatic dense annotation of large-vocabulary sign language videos	121
7.1	Introduction	123
7.2	Related Work	125
7.3	Densification	128
7.3.1	Mining more Spottings through In-domain Exemplars (E) . . .	129

7.3.2	Discovering Novel Sign Classes (N)	130
7.3.3	Pseudo-labelling as a Form of Sign Spotting (P)	130
7.3.4	Improving the Old (M^* , D^*)	131
7.3.5	Evaluation Framework	133
7.4	Experiments	135
7.4.1	Data and Evaluation Protocol	135
7.4.2	Results	137
7.5	Conclusion	141

III Sequence recognition in Sign Language 142

8 Weakly-supervised Fingerspelling Recognition in British Sign

	Language Videos	143
8.1	Introduction	145
8.2	Related work	147
8.3	Fingerspelling detection and recognition	149
8.3.1	The Transpeller architecture	150
8.3.2	Multiple Hypotheses (MH) CTC loss	152
8.4	Automatic annotations	153
8.4.1	Exemplar and mouthing annotations	153
8.4.2	Transpeller annotations	154
8.4.3	Multi-stage training	154
8.5	Experiments	155
8.5.1	BOBSL Fingerspelling benchmark	155
8.5.2	Results	156
8.5.3	Architecture ablations	158
8.6	Conclusion	160

9 A Tale of Two Languages: Large-Vocabulary Continuous Sign

	Language Recognition from Spoken Language Supervision	161
9.1	Introduction	163
9.2	Related Work	165
9.3	Joint Space for Signed and Spoken Languages	168
9.3.1	Model overview and inference	169

9.3.2	Training with sentence- and sign-level losses	170
9.3.3	Sources of supervision	171
9.3.4	Implementation details	172
9.4	A New CSLR Evaluation Benchmark	174
9.5	Experiments	176
9.5.1	Data and evaluation protocol	176
9.5.2	Baselines	177
9.5.3	Ablation study	178
9.5.4	Comparison to the state of the art	179
9.5.5	Qualitative analysis	181
9.6	Conclusion	182

IV Enhancing verb representations 183

10 Verbs in Action: Improving verb understanding in video-language

models	184
10.1	Introduction 186
10.2	Related works 188
10.3	Method 191
10.3.1	Preliminaries 191
10.3.2	Verb-Focused Contrastive Pretraining (VFC) 192
10.3.3	Implementation details 195
10.4	Experiments 196
10.4.1	Verb-Focused Benchmarks 197
10.4.2	Ablation Study 198
10.4.3	Comparisons to the State of the Art 202
10.5	Conclusion 205

11 Discussion 206

11.1	Achievements and Impact 206
11.2	Ethical considerations 208
11.2.1	Applications 208
11.2.2	Limitations 209
11.3	Future Work 210

11.3.1 Sign Language Translation	210
11.3.2 Model-assisted annotation for video	211
11.3.3 Temporal modeling	211
11.4 Conclusion	212
References	214
A Statement of Authorship	248

Chapter 1

Introduction

In today’s digital landscape, video content is experiencing an unprecedented surge, swiftly becoming the predominant mode of communication. Hundreds of hours of footage are uploaded to online platforms such as Youtube and Instagram every minute, and the trend of video content creation and consumption shows no signs of slowing down. A recent study¹ reveals that viewers retain 95% of a message when watching it on a video versus only 10% through text, highlighting the intrinsic richness, engagement, and dynamism of video content. For social media and video-sharing platforms, key video applications include search and retrieval, recommendation systems as well as content moderation. Beyond entertainment, video understanding extends to other applications such as autonomous driving [A. Hu et al. 2023], animal behavior analysis [Bain et al. 2021a], sports evaluations [Zhe Wang et al. 2023] and health diagnosis [Ouyang et al. 2020].

In the realm of deep learning, developing models for video understanding tasks demands an abundance of annotated training examples. Over the past decade, significant efforts have been directed towards the manual curation of video datasets. These datasets typically fall into two categories: video classification datasets [Soomro et al. 2012; Kuehne et al. 2011; Joao Carreira and Zisserman 2017], where short video clips are paired with action labels, and video caption datasets [Das et al. 2013; J. Xu et al. 2016], where video clips are associated to captions describing the captured objects and events. While these datasets have facilitated substantial advancements in video modeling, manual annotation remains an arduous, time-

¹<https://www.synthesia.io/post/video-statistics>

consuming and costly process. It does not scale efficiently to the vast amount of unlabelled video data, particularly as the complexity of the task increases (for example, from action classification to object tracking), and as we shift to untrimmed videos lasting minutes or even hours.

Obtaining labeled training data is therefore a fundamental challenge of the field. To alleviate the need for manual supervision, researchers typically pursue two main strategies. The first strategy entails video-only self-supervised learning methods, serving as a promising alternative to traditional supervised approaches [Krizhevsky et al. 2012]. These methods learn from unlabelled, uni-modal data by solving a pretext task. The pretext task typically does not match the final video understanding task of interest. Instead, it is designed to encourage the model to learn strong, general and semantically meaningful representations that can subsequently be transferred to downstream applications. Recent advancements in video modeling have introduced various pretext tasks, including spatio-temporal Jigsaw puzzles [Noroozi and Favaro 2016; Ahsan et al. 2019; Huo et al. 2021], frame or clip ordering [Fernando et al. 2017; Misra et al. 2016; D. Xu et al. 2019], and prediction of masked video tubelets [Tong et al. 2022; Bardes et al. 2024] or future frames [T. Han et al. 2019]. While the concept of self-supervision theoretically enables easy scalability without the need for annotation efforts, the development of effective self-supervised algorithms remains a challenging endeavor. This is particularly true for video and image-based approaches, which often require extensive tuning compared to their language-based counterparts.

An alternative strategy for scaling up datasets without the cost of labeling involves collecting readily available video data from the Web, along with various sources of associated textual metadata such as titles, tags, speech transcriptions [Miech et al. 2019], descriptions [Bain et al. 2021b], or comments [Hanu et al. 2022]. While this form of language supervision is easily accessible, and initially generated by humans (via text or speech), it is considered weak. In fact, the supervision may be incomplete, lacking spatial or temporal correspondences, and may fail to describe all objects and events in the video. Additionally, the supervision may be inaccurate and noisy, with the annotation semantics not aligning with the video content or the language being ambiguous. Despite these challenges, scaling up datasets in this manner offers significant advantages, notably automatic scalability, and the

ability to capture a diverse array of concepts and scenes. Importantly, recent works have demonstrated that such large-scale datasets can facilitate the training of joint vision-text embedding spaces, yielding state-of-the-art results on various tasks [Alayrac et al. 2022; Radford et al. 2021]. In contrast to self-supervised methods (which may sometimes seem unnatural), weakly supervised, multi-modal approaches enable video representation learning by exploiting the inherent shared information between video and language. In addition, by leveraging language, models can be trained to naturally comprehend videos, connecting visual content to human concepts.

While weakly supervised, multi-modal approaches hold considerable promise, the efficacy of the learned representations relies heavily on two factors: (i) the volume of available training data, and (ii) the level of noise in the supervision. This thesis endeavors to address both of these challenges within the domain of automatic sign language understanding, an area that remains largely unsolved, despite advancements in related vision and language tasks. To combat data scarcity, we leverage sign language interpreted TV broadcasts along with corresponding weakly-aligned subtitles, tapping into readily available resources. Furthermore, to enhance the signal to noise ratio, we explore diverse methodologies in subsequent chapters for obtaining stronger supervision from weak supervision for sign language data, thereby facilitating learning strong video representations.

We begin the thesis by introducing motivations in Section 1.1. We then present key concepts behind the work in Section 1.2. Finally, in Section 1.3, we highlight the four primary themes of the thesis along with their respective contributions, followed by a full list of included research papers in Section 1.4.

1.1 Motivation

Sign languages serve as the natural means of communication for deaf communities [Rachel Sutton-Spence and Woll 1999]. They are visual-spatial languages and lack standardized written forms. They exist independently of spoken languages, possessing their own lexicons and grammatical structures. Indeed, the ordering of words between spoken and sign languages is typically not preserved. Sign languages are expressed through both manual and non-manual components,

potentially simultaneously. In addition to hand shape, location and motion, other articulators such as the eyebrows, mouth, head, shoulders and eye gaze all contribute to semantics [Wilbur 2000]. Today, there exist over 200 distinct sign languages (interestingly, American and British sign languages are different) and 70 million deaf individuals worldwide using them.²

In this thesis, we establish the groundwork for developing a robust and scalable solution for sign language translation: the task of predicting a natural text sentence from a video sequence of signs. Despite great progress in related fields such as lipreading [K R Prajwal et al. 2022a] and translation of spoken and written languages [A. Fan et al. 2021], the current performance of automatic sign language translation models remains limited [Koller 2020], with sign language technologies significantly lagging behind [Wojtanowski et al. 2020]. Automatic sign language translation has the potential for large societal impact when deployed in the real world, by fostering inclusivity and bringing communities closer together. Potential applications span various domains, including educational tools for sign language learners with features like auto-correct prompts (‘did you mean this sign?’); translation of signed queries into text for search engines; integration of virtual assistants to respond to signed wake words (e.g. ‘OK Google’, ‘Hey Siri’); automatic transcription of signed content to facilitate efficient search and indexing of sign language videos; and real-time automatic interpreting in video calls, or critical scenarios such as hospitals, police stations, and airports, where human interpreters may not be readily available.

A key obstacle to making progress towards automatic sign language translation is the scarcity of large-scale annotated training data. Sign languages, being low-resource languages, have limited availability of sign language videos online. Moreover, the manual annotation of signing is very challenging, given the absence of standard written forms in sign languages and the use of multiple input streams (for example, the hands and mouth can convey an object and its description simultaneously). This necessitates proficient annotators, skilled in sign language grammar and equipped with advanced annotation tools. To tackle this obstacle, drawing inspiration from [Buehler et al. 2009], we propose to leverage a readily available and large-scale source of data: sign language translated TV broadcasts that con-

²<https://wfdeaf.org/our-work/>

sist of an overlaid interpreter performing signs, and subtitles corresponding to the speech content. In this thesis, we focus mainly on BBC shows and therefore on British Sign Language (BSL), the sign language of the British deaf community.

Although interpreted TV programs are abundant and easily accessible, leveraging written subtitles as the primary source of signing supervision presents additional challenges. The supervision provided by the subtitles is weak because the subtitles are temporally aligned with the speech content, but not necessarily with the signing – a sign may appear several seconds before or after its corresponding translated word appears in the subtitles. Moreover, the supervision is noisy because the presence of a word in the subtitles does not necessarily imply that the word is signed, and vice versa. In fact, sign interpreters translate speech rather than transcribe it, resulting in a many-to-many mapping between signs and subtitle words. Additionally, both signed and spoken language lexicons are extensive, and represent long-tailed distributions. All these factors introduce significant noise and difficulty in training translation models directly from interpreted signing video and subtitle pairs, and in practice, this approach fails to achieve meaningful results [N. Camgoz et al. 2021]. Instead, in this thesis, we propose to focus on automatically and densely annotating the sign sequences in videos, by leveraging the weakly aligned speech from which they are interpreted. We explore various approaches to achieve this goal, primarily involving querying words in the subtitle text and searching for corresponding signs in the sign language video using visual cues. The approaches we explore in subsequent chapters enable us to bootstrap weak supervision to construct larger, more diverse datasets with stronger supervision and improved alignment, which are crucial for training translation models capable of generalizing in real-world scenarios and at scale, where their potential impact is greatest.

We note several limitations of our approach. Firstly, whilst interpreted signing offers the opportunity to scale up training data, it introduces certain biases compared to conversational signing used in deaf communities. Interpreting can lead to a simplification in signing style and vocabulary, and even a reduction in speed for comprehension [Bragg et al. 2019]. Although our ultimate goal is to transition to conversational signing, learning effective representations of signs from interpreted data serves as a foundational step in this direction. However, we highlight the ne-

cessity of future works to bridge this domain gap, potentially by creating datasets with native signing. Secondly, this work focuses solely on densely annotating lexical signs, which are easily associated with spoken language words. However, non/partially-lexical signs, such as pointing signs, depicting signs and fragment buoys (used to make associations between identities) [Belissen et al. 2020a], are integral components of sign language and must be considered for accurate translations. Partially lexical signs exhibit appearances that are highly dependent on context, and are utilized, for instance, to convey position, motion, size and shape of objects [Braffort and Filhol 2014].

1.2 Key Ideas

The research presented in this thesis, centered on sign language understanding, contributes to and builds upon several fundamental themes in machine learning, including multimodal representation learning and learning with weak supervision.

1.2.1 Multimodal Learning

The world inherently presents itself through multiple modalities, and consequently, the data we gather mirrors this diversity of experiences. For example, a video uploaded on Youtube often encompasses additional modalities, such as audio (background sounds, music or speech) and text (titles, comments, and subtitles). Moreover, other modalities, like body keypoints reflecting human pose or optical flow capturing motion, can be extracted. While different modalities may contain overlapping information (for example, speech and subtitle transcriptions corresponding to the audio content), each modality offers a distinct perspective on the captured data (for example, subtitle text may not convey crucial intonations and pauses in the audio). Recent learning algorithms have thus transitioned from uni-modal settings to leverage the natural multimodality of data, accounting for the fact that individual modalities may be noisy and insufficient in conveying the complete meaning of the captured experience. However, in practice, multimodal learning poses challenges due to the considerable variation in representation spaces across modalities. For instance, images are typically viewed as continuous signals, rich in spatial information, while text is discrete, governed by language grammar and

syntax. Developing algorithms therefore demands careful consideration of model design choices and learning objectives to effectively bridge modalities.

In this thesis, focused on sign language understanding, a visual-spatial language expressed through multiple articulators, we naturally explore a range of multi-modal approaches. Here, modalities encompass signing videos and corresponding spoken language text, but also different body parts such as hands and mouth. We explore diverse forms of multimodal learning, including (i) transforming one modality into another. Notably, in Chapters 6 (‘Read and Attend’), 8 (‘Weakly-supervised Fingerspelling’) and 9 (‘Large-vocabulary CSLR’), we introduce models that take continuous signing video sequences as input and generate text as output. Specifically, Chapter 6 aims to predict subtitle text, while Chapters 8 and 9 aim to predict sequences of fingerspelled characters and signed words, respectively. Additionally, we investigate methods for (ii) jointly learning from multiple modalities. For instance, in Chapters 2 (‘Audio-visual KWS’) and 3 (‘Visual KWS’), we propose architectures utilizing audio, visual and textual inputs for keyword spotting, employing late and mid fusion strategies in Chapters 2 and 3, respectively. Finally, we explore (iii) how one modality can enhance learning in another. For example, in Chapter 4 (‘BSL-1K’), we explore how labeled data in one modality (mouth movements in talking faces) can facilitate the transfer of supervision to another modality (hands in sign language). In Chapter 10 (‘Verbs in Action’), we also examine how text can be leveraged to generate valuable negatives in contrastive pretraining to enhance video representations.

1.2.2 Learning with Weak Supervision

Weakly-supervised learning makes use of partial, noisy labels, or even unlabelled data to train models. It offers an effective, scalable learning strategy, shown to achieve remarkable generalization [Radford et al. 2021; Miech et al. 2019]. This approach is particularly well-suited for numerous real-world applications where supervision is often incomplete. However, the lack of detailed supervision can sometimes result in suboptimal model performance, especially when data is limited. Designing such methods is indeed challenging as models need to discern between true patterns and noisy information during training.

In this thesis, we propose to leverage weakly supervised data – specifically, sign

language interpreted TV broadcast along with subtitles – to expand sign language datasets by an order of magnitude, surpassing 1000 hours of video. A key idea in subsequent chapters is to derive stronger supervision from weak signals, thereby improving the learning of sign representations. We investigate various weakly-supervised methods to automatically and densely annotate sign sequences in videos, by using the subtitles which offer insights into potential signs present. In particular, in Chapter 5 (‘Watch Read Lookup’), we present a (i) multiple instance learning approach, where a bag of instances (rather than a single instance) is associated with a single label. In Chapter 8 (‘Weakly-supervised Fingerspelling’), we propose a (ii) multiple hypothesis framework, tasking the model with selecting the most accurate label for the signing instance from a set of hypotheses. We show the advantages of (iii) pseudo-labelling and bootstrapping methods in Chapter 4 (‘BSL-1K’) and Chapter 9 (‘Large-Vocabulary CSLR’), where models trained on labeled data generate labels for unlabeled data iteratively. In Chapter 6 (‘Read and Attend’) and Chapter 7 (‘Automatic dense annotation’), we demonstrate how to leverage (iv) similarities between cross-modal or uni-modal representations, alongside the noisy constraints imposed by the subtitle content. These approaches for acquiring stronger supervision lay the groundwork for training robust and scalable translation models. Beyond sign language, in Chapter 10 (‘Verbs in Action’), we illustrate the benefits of (v) data augmentation through large language models to obtain hard negatives for verb understanding, serving as a form of stronger supervision.

1.3 Thesis Outline and Contributions

In this section, we present an overview of the subsequent chapters in the thesis. The thesis is structured into four main parts: (i) Keyword Spotting in Sign Language, (ii) Approaches for Sign Spotting, (iii) Sequence Recognition in Sign Language, and (iv) Enhancing Verb Representations. For Chapters 2 to 10, we outline the main contributions below. Chapter 11 explores the implications of this research and suggests potential avenues for future exploration.

1.3.1 Keyword Spotting in Sign Language

Signers often simultaneously mouth the word they sign, as an additional signal [Rachel Sutton-Spence and Woll 1999], performing similar lip movements as for the spoken word. Mouthings serve various purposes, including disambiguating manual homonyms – signs that share visual similarity but convey different meanings – or simply to provide redundancy [Woll 2001]. In this theme, we introduce the concept of leveraging mouthing cues from signers to automatically localize sign instances, thereby acquiring stronger supervision. Our approach entails using the weakly-aligned subtitles along with a visual keyword spotting model, whose goal is to automatically determine whether and when a keyword from the subtitle is mouthed within a continuous signing window.

In Chapter 2 (‘Audio-visual KWS’), we leverage lipreading datasets of talking faces [Chung and Zisserman 2016a; Chung et al. 2017] to train a novel convolutional architecture for visual keyword spotting. Inspired by object detection methods, our model uses a similarity map intermediate representation between visual and phonetic modalities to separate the task into two steps: sequence matching and pattern detection. Beyond its applications for sign language, we demonstrate the model’s versatility in leveraging audio and extending to other spoken languages such as French and German. In Chapter 3 (‘Visual KWS’), we improve the visual keyword spotting model by integrating Transformers, which allow for much stronger interaction between the visual and phonetic streams through full cross-modal attention. We also conduct preliminary experiments illustrating the model’s capability to generalize from videos of talking faces to out-of-domain data of signer mouthings. In Chapter 4 (‘BSL-1K’), we use visual keyword spotting of mouthings to automatically annotate hundreds of thousands of sign instances for a vocabulary of 1,000 signs in 1,000 hours of video. We show how the automatically collected data can be used to train strong sign recognition models for co-articulated signs in BSL, with these models also serving as excellent pretraining for other sign languages.

1.3.2 Approaches for Sign Spotting

While leveraging mouthings in Part I provides a strong signal, not all signs can be identified in this manner since signers do not mouth continuously. Under this

theme, we investigate methods to further our sign discovery process by utilizing automatic annotations previously collected to learn robust sign representations, together with noisy constraints imposed by the subtitle content.

There is a rich body of literature on using visual exemplars for spatial localisation of objects or temporal localisation of actions [Deselaers et al. 2010; K. Cao et al. 2020]. In Chapter 5 (‘Watch Read Lookup’), we propose to leverage visual exemplars from sign dictionaries to localize sign instances in continuous signing, without being limited to mouthings. Sign language dictionaries offer the advantage of covering a large vocabulary of signs. However, this task presents challenges: (i) dictionaries typically contain only a few example videos per sign, (ii) a word may be signed in different ways due to semantic or regional variations, and (iii) there exists a significant domain gap between isolated signing in dictionary videos and the co-articulated, continuous signing we wish to annotate. To effectively learn a joint embedding space between these two sources of signing, we leverage prior mouthing-based sign annotations, dictionaries, and subtitles within a framework grounded in Multiple Instance Learning and Noise Contrastive Estimation [Miech et al. 2020]. When combining mouthing and dictionary based automatic annotations, we boost sign recognition performance.

In Chapter 6 (‘Read and Attend’), we leverage the fact that cross-modal attention has been employed in the literature for various localisation problems such as visual grounding in videos [Huijuan Xu et al. 2019] or images [Deng et al. 2018], and audio-visual sound source localisation [Arandjelovic and Zisserman 2017]. Specifically, our approach exploits the attention mechanism of the Transformer, trained on a video-to-text sequence prediction task with weakly aligned subtitles. The core hypothesis motivating this approach is that in order to solve the sequence prediction task, the attention mechanism of the Transformer must be capable of localising sign instances. Through our learned attention, we automatically annotate hundreds of thousands of new sign instances. By adding these automatic annotations to those obtained from mouthings and dictionaries, we train an even stronger sign recognition model.

In Chapter 7 (‘Automatic dense annotation’), we propose a simple, scalable framework to vastly increase the density of automatic annotations. We measure density in two ways: (i) minimizing temporal gaps in the timeline to achieve a densely

spotted signing sequence; and also (ii) increasing the number of words we recall in the corresponding subtitle. We significantly improve previous annotation methods (from mouthings and dictionaries) by making use of synonyms and automatic subtitle-signing alignment. Moreover, we show the value of pseudo-labelling from a sign recognition model as a means of sign spotting. Lastly, we introduce a novel approach for increasing our annotations of known and unknown classes based on in-domain exemplars, effectively propagating previously collected examples across video data by leveraging sign representation similarities.

1.3.3 Sequence recognition in Sign Language

While the focus in Part I and Part II revolves around identifying and localizing individual signs within continuous signing windows, our objective in this theme is to delve into sign language tasks that entail predicting sequences of signs. These tasks represent pivotal foundational steps towards realizing sign language translation capabilities.

In Chapter 8 (‘Weakly-supervised Fingerspelling’), we aim to detect and recognise sequences of letters signed using fingerspelling. Fingerspelling in signed languages is a means to encode words from written language into sign language via a manual alphabet, i.e. one sign per letter. Words from a written language with no known sign may be fingerspelled, such as names of people and places. Within our interpreted TV shows, we estimate roughly 5-10% of signs are fingerspelled. Consequently, it is important to incorporate automatic fingerspelling recognition methods to be able to exhaustively translate. In contrast to other methods, our approach only uses weak annotations from subtitles for training. We propose a Transformer architecture adapted to this task, with a novel multiple-hypothesis CTC loss function to learn from alternative annotation possibilities. We employ a multi-stage training approach to enhance our training data before retraining again to achieve better performance.

In Chapter 9 (‘Large-vocabulary CSLR’), we focus on the task of large-vocabulary continuous sign language recognition (CSLR) – providing time aligned and dense word predictions for each sign within a signing sequence. This is an essential first step towards translation, as English sentence-level annotations have been shown to be difficult to use directly as targets for sign language translation [N. Camgoz

et al. 2021]. To facilitate CSLR evaluation in the large-vocabulary context, we manually curate the most extensive test set of continuous sign-level annotations, spanning over 6 hours. Our proposed approach then involves leveraging weak and noisy pseudo-labels generated from a sign recognition model, and constructing a multi-task model capable of both CSLR and retrieving sign language to subtitle sentences. Through this strategy, we demonstrate promising results in tackling the demanding large-vocabulary setting.

1.3.4 Enhancing verb representations

In this final theme, we explore obtaining stronger supervision from weak supervision for a more general task, beyond the domain of sign language. Specifically, our focus shifts to the task of understanding verbs, which is crucial for modeling interactions among people, objects and the environment through space and time. In fact, recent state-of-the-art video-language pretrained models based on CLIP have limited verb understanding and rely extensively on nouns, as evidenced by evaluations on new benchmarks [Hendricks and Nematzadeh 2021; Park et al. 2022]. This restricts their performance in real-world video applications that require action and temporal understanding.

In Chapter 10 (‘Verbs in Action’), we propose the first method to address the verb understanding challenge in video-language models, while preserving their proficiency in noun-related tasks. We introduce a contrastive framework consisting of two components: (i) leveraging pretrained large language models to create hard negatives for cross-modal contrastive learning, together with a calibration strategy to ensure balanced concept occurrences in positive and negative pairs; and (ii) enforcing a fine-grained alignment loss for extracted verb phrases. Through this pretraining strategy, we develop a unified model that improves zero-shot verb understanding performance across a range of downstream tasks (video-text matching, video question-answering and video classification); while maintaining robust performance in noun-centric scenarios.

1.4 Publications

In this section, we list publication contributions. Excluding Chapter 8 which introduces ongoing work, Chapters 2 to 10 each contains a research paper which has been peer reviewed and accepted for publication in a conference. The published papers are included here without modifications, except for formatting changes. Additional implementation details for each paper can be found in the supplementary materials of their online versions. A statement of authorship is also presented for each paper in the Appendix. The papers included in the thesis are listed below.

Chapter 2: “Seeing wake words: Audio-Visual Keyword Spotting” Liliane Momeni, Triantafyllos Afouras, Themis Stafylakis, Samuel Albanie, Andrew Zisserman. In British Machine Vision Conference, 2020.

Chapter 3: “Visual Keyword Spotting with Attention” K R Prajwal*, Liliane Momeni*, Triantafyllos Afouras, Andrew Zisserman. In British Machine Vision Conference, 2021.

Chapter 4: “BSL-1K: Scaling up co-articulated sign language recognition using mouthing cues” Samuel Albanie*, Gül Varol*, Liliane Momeni, Triantafyllos Afouras, Joon Son Chung, Andrew Zisserman. In European Conference on Computer Vision, 2020.

Chapter 5: “Watch, read and lookup: learning to spot signs from multiple supervisors” Liliane Momeni*, Gül Varol*, Samuel Albanie*, Triantafyllos Afouras, Andrew Zisserman. In Asian Conference on Computer Vision (Best Application Paper), 2020.

Chapter 6: “Read and Attend: Temporal Localisation in Sign Language Videos” Gül Varol*, Liliane Momeni*, Samuel Albanie*, Triantafyllos Afouras*, Andrew Zisserman. In Conference on Computer Vision and Pattern Recognition, 2021.

Chapter 7: “Automatic dense annotation of large-vocabulary sign language videos” Liliane Momeni*, Hannah Bull*, K R Prajwal*, Samuel Albanie, Gül Varol, Andrew Zisserman. In European Conference on Computer Vision, 2022.

Chapter 8: “Weakly-supervised Fingerspelling Recognition in British

Sign Language” K R Prajwal*, Hannah Bull*, **Liliane Momeni***, Samuel Albanie, Gül Varol, Andrew Zisserman. In British Machine Vision Conference, 2022.

Chapter 9: “A Tale of Two Languages: Large-Vocabulary Continuous Sign Language Recognition from Spoken language supervision” Charles Raude*, K R Prajwal*, **Liliane Momeni***, Hannah Bull, Samuel Albanie, Andrew Zisserman, Gül Varol. Work in progress.

Chapter 10: “Verbs in Action: Improving verb understanding in video-language models” **Liliane Momeni**, Mathilde Caron, Arsha Nagrani, Andrew Zisserman, Cordelia Schmid. In International Conference on Computer Vision, 2023.

Papers not included:

“Signer Diarisation in The Wild” Samuel Albanie*, Gül Varol*, **Liliane Momeni***, Triantafyllos Afouras, Andrew Brown, Chuhan Zhang, Ernesto Coto, N. Cihan Camgöz, Ben Saunders, Abhishek Dutta, Neil Fox, Richard Bowden, Bencie Woll, Andrew Zisserman. Technical Report, 2021.

“Aligning Subtitles in Sign Language Videos” Hannah Bull*, Triantafyllos Afouras*, Gül Varol, Samuel Albanie, **Liliane Momeni**, Andrew Zisserman. In International Conference on Computer Vision, 2021.

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

* denotes equal contribution

Part I

Keyword Spotting in Sign Language

Chapter 2

Seeing wake words: Audio-Visual Keyword Spotting

The paper has been accepted for publication at the British Machine Vision Conference (BMVC), 2020.

Seeing wake words: Audio-Visual Keyword Spotting

Liliane Momeni¹ Triantafyllos Afouras¹

Themis Stafylakis² Samuel Albanie¹ Andrew Zisserman¹

¹ Visual Geometry Group, University of Oxford, UK

² Omilia Conversational Intelligence, Athens, Greece

Abstract

The goal of this work is to automatically determine *whether* and *when* a word of interest is spoken by a talking face, with or without the audio. We propose a *zero-shot* method suitable for ‘*in the wild*’ videos. Our key contributions are: (1) a novel convolutional architecture, KWS-Net, that uses a *similarity map* intermediate representation to separate the task into (i) *sequence matching*, and (ii) *pattern detection*, to decide whether the word is there and when; (2) we demonstrate that if audio is available, visual keyword spotting improves the performance both for a clean and noisy audio signal. Finally, (3) we show that our method generalises to other languages, specifically French and German, and achieves a comparable performance to English with less language specific data, by fine-tuning the network pre-trained on English. The method exceeds the performance of the previous state-of-the-art visual keyword spotting architecture when trained and tested on the same benchmark, and also that of a state-of-the-art lip reading method.

2.1 Introduction

Keyword spotting (KWS) is the task of detecting a word of interest within continuous speech. In audio-visual data, the keyword can be detected from the audio stream only, from the visual stream only, or from both streams. The task differs

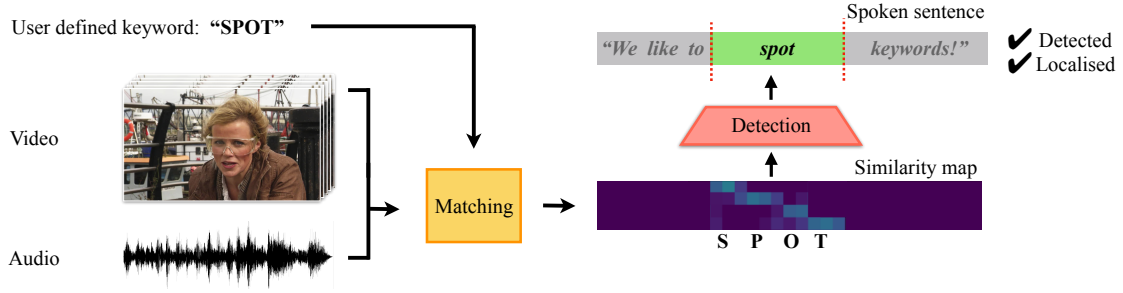


Figure 2.1: **General approach of KWS-Net:** The inputs to the model are a user-specified keyword and either audio, video or both. The objective is to detect *whether* the keyword occurs in the input signal and, if present, then *where* it is.

from automatic speech recognition (ASR) or from automatic visual speech recognition (AVSR, lip reading), where the aim is to recognise the phrases and sentences being spoken from scratch. In KWS, the word that is sought is provided by the user, and consequently the task is easier than recognising with no knowledge as in ASR or AVSR. This suggests that a KWS model can (i) be much simpler than ASR or AVSR, and (ii) have higher performance.

KWS is more practical in many situations. Indeed, ASR is frequently not the aim of real-world speech processing applications and complete speech transcription can therefore be redundant. Keyword search, which consists in retrieving speech utterances including a keyword from a large database, is often a more useful task. KWS also surpasses ASR in cases where context is limited, for example for detecting mouthings in sign language [Albanie et al. 2020].

Visual KWS has clear applications to cases where audio is unavailable such as for browsing archival silent films, and more importantly for cases where audio has been corrupted with noise, including for wake-word recognition (e.g. ‘OK Google’, ‘Hey Siri’ and ‘Alexa’) as well as other human-robot interactions, such as in smart home technologies (for example, turning off the lights) or to assist people with speech impairment or aphonia [Shillingford et al. 2018].

A fundamental constraint for any visual KWS system is detecting words which sound different but involve the same lip movements (they have the same ‘visemes’ – visemes are the visual equivalent of phonemes; phonemes are the smallest unit of sound in speech). For instance, the words ‘may’, ‘pay’ and ‘bay’ cannot be distinguished without audio as the visemes for ‘m’, ‘p’ and ‘b’ look the same. Other difficulties include intra-class differences (such as accents, speed of speech

and mumbling which modify lip movements) and variable imaging conditions (such as lighting, motion, resolution) [Chung and Zisserman 2016a]. Spotting words from continuous speech is also challenging as there may be co-articulation of the lips.

In this paper, we introduce a novel convolutional architecture, KWS-Net, for spotting keywords in *visual* speech. The model introduces a *similarity map* that splits the task into (i) *matching* a token phoneme sequence against a viseme sequence, and (ii) *detecting* an *alignment pattern* to decide whether and when the keyword occurs (see Figure 2.1). Step (ii) is performed in a *detector-by-classification* manner, inspired by sliding window object detection methods. The model is able to spot words that are *unseen* during training, and are specified by a user at test time (zero-shot). We show that KWS-Net exceeds the previous state-of-the-art network of Stafylakis *et al.* [Stafylakis and Tzimiropoulos 2018] for visual KWS on standard benchmarks. Furthermore, we show that audio-visual KWS outperforms the audio-only KWS counterpart marginally for clean audio, but substantially for noisy audio. The visual-only and audio-visual KWS models are described in Section 2.3. Finally, we apply our method to French and German datasets built from TED videos (see Section 2.4) and demonstrate that our model can perform comparably to English in other languages with less language specific training data. The project webpage is at: www.robots.ox.ac.uk/~vgg/research/kws-net/.

2.2 Related Work

Lip reading. Recent deep learning methods involving character-level recognition of visual sequences can be divided into two types: (i) models trained with a Connectionist Temporal Classification (CTC) loss [Graves *et al.* 2006], where frame-wise label predictions are made in search for an optimal alignment with the output sequence, and (ii) models trained with a sequence-to-sequence (seq2seq) loss, that first read the entire input before attending to different parts of it at each step of an autoregressive output sequence prediction process. Examples of CTC models include LipNet [Assael *et al.* 2016] and more recently LSVSR [Shillingford *et al.* 2018], that shows state-of-the-art performance with a word error rate as low as 40.9% when trained on vast amounts of data. Examples of seq2seq models include the LSTM with attention model from Chung *et al.* [Chung *et al.*

2017], which extends the audio model ‘Listen, attend and spell’ [Chan et al. 2016] to visual and audio-visual ASR. Afouras *et al.* [Afouras et al. 2018a] combine the seq2seq loss with self-attention layers and propose a transformer-based model. Hybrid approaches combining CTC and seq2seq losses were also recently proposed [Petridis et al. 2018; Afouras et al. 2019], demonstrating promising results on the LRS2 benchmark [Chung and Zisserman 2016c; Afouras et al. 2019].

Audio KWS. Traditional audio-based KWS methods are based on HMMs [Szoke et al. 2005]. More recent deep learning works investigate fully connected networks [G. Chen et al. 2014; Tucker et al. 2016], time delay neural networks [Myer and Tomar 2018; M. Sun et al. 2017], convolutional neural networks (CNNs) [Sainath and Parada 2015; Yundong Zhang et al. 2017; Yuxuan Wang et al. 2017; Palaz et al. 2016], graph convolutional neural networks [X. Chen et al. 2019], and recurrent neural networks (RNNs) [Fernandez et al. 2007; Hwang et al. 2015; M. Sun et al. 2016]. RNNs are also combined with convolutional layers [Arik et al. 2017; Lengerich and Hannun 2016; Taejun Kim and Nam 2019] to simultaneously model local features and temporal dependencies. Recent works also explore seq2seq models for KWS [Haitong Zhang et al. 2018; Audhkhasi et al. 2017; Zhuang et al. 2016; Rosenberg et al. 2017].

Visual KWS. Yao *et al.* [Yue Yao et al. 2019] use sliding windows to split sentence-level videos into smaller segments on which they perform word-level classification and aggregate across segments using a max pooling layer. Their method is used for a closed-set of 1000 Mandarin keywords, whereas our method is zero-shot. We cannot compare to their work as (i) we do not have access to Mandarin phonetic dictionaries, and (ii) their validation and test sets are unavailable. Jha *et al.* [Jha et al. 2018] propose a query by example visual KWS architecture, where the word query and retrieval are both videos, and a cosine similarity score is used to assign a label query to a target video. Recently, Stafylakis *et al.* [Stafylakis and Tzimiropoulos 2018] devised an end-to-end architecture which uses RNNs to learn correlations between visual features and a keyword representation, extracted from a grapheme-to-phoneme encoder-decoder.

Audio-visual KWS. Ding *et al.* [Ding et al. 2018] build an audio-visual decision fusion KWS system, consisting of 2D CNNs to model the time-frequency features of the log mel-spectrogram and 3D CNNs to model the spatio-temporal features

of the mouth. The softmax outputs of the audio and visual networks are combined through a summation, with fixed weights for each modality, to estimate the posterior probability of each keyword. In [P. Wu et al. 2016], adaptive decision audio-visual fusion based on HMMs is performed using a proposed lip descriptor. Both of these works are evaluated on the private, relatively small PKU-AV dataset of 3000 clips and 30 keywords, involving no more than 20 speakers and excluding any mouth occlusions. These methods are evaluated with keywords seen during training, as opposed to zero-shot.

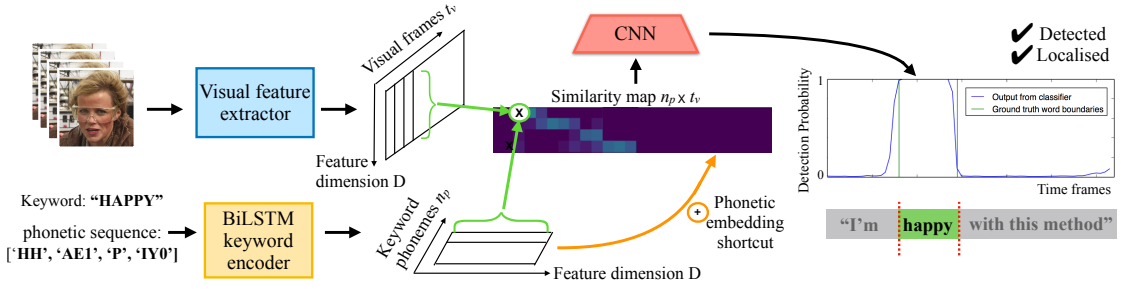


Figure 2.2: **Visual-only KWS-Net pipeline:** The viseme and phonetic sequence embeddings are used to compute a similarity map, which is expected to show a strong diagonal component when the keyword is present. This pattern can be detected by a CNN-based classifier. The output keyword detection probabilities are plotted for the clip. See details in Section 2.3.

2.3 KWS-Net

The visual KWS-Net model, shown in Figure 2.2, contains two input streams: a visual feature extractor and a keyword encoder that produces an embedding for the pronunciation of the queried keyword. The visual and phonetic representations are fused into a single channel similarity bottleneck, which is then passed through a CNN classifier to detect an alignment pattern. Full details of the model are given in the appendix.

Visual feature extractor. The visual feature extractor takes as input a sequence of frames from a clip of a talking face and outputs visual features. The feature extraction is based on an 18-layer spatio-temporal ResNet [K. He et al. 2016; Stafylakis and Tzimiropoulos 2017] which has shown good results on related tasks such as lip reading [Afouras et al. 2019] and audio-visual speech enhancement [Afouras et al. 2018c]. The network applies 3D convolutions on the input image sequence, followed by a 2D ResNet that gradually decreases the spatial dimensions, while

preserving the temporal resolution. The visual encoding obtained is a sequence of dimension $t_v \times 512$, where t_v is the number of input frames. The features are then passed through a BiLSTM [Hochreiter and Schmidhuber 1997; Schuster and Paliwal 1997] to model temporal dynamics.

Keyword encoder. The keyword encoder is a BiLSTM that ingests the phoneme token sequence of the input keyword (e.g. ‘HH,’ ‘AE1,’ ‘P,’ ‘IY0’ for ‘happy’), obtained using the CMU pronouncing dictionary [Speech Group at Carnegie Mellon University 2014], and outputs a phonetic keyword embedding sequence with dimensions $n_p \times 512$, where n_p is the number of phonemes in the keyword.

Similarity map. We compute the dot product between the phonetic sequence embedding P ($n_p \times 512$) and the visual feature sequence V ($t_v \times 512$) which results in a similarity map ($n_p \times t_v$), expected to show high activation when the keyword occurs in the clip (positive pair), i.e. when the two modalities align.

CNN detector and classifier. The similarity map is processed by a shallow CNN, which outputs the probability that the keyword is present at a specific location, by detecting patterns in it (e.g. a strong diagonal component). The CNN gradually subsamples the temporal dimension by a factor of 8 and collapses the phoneme dimension to a singleton, resulting in an output of length $t_v^{out} = t_v/8$. We apply a sigmoid activation on the resulting temporal sequence that outputs for every frame the probability that the keyword occurs around it. The sample is predicted to contain the keyword if the maximum probability over all the frames is above a certain threshold, and the frame position of the maximum is regarded as the predicted location of the keyword.

As shown in Figure 2.2, before feeding the similarity map to the CNN, we concatenate the phonetic sequence embedding (broadcast over time) to it. The intuition for the addition of this shortcut is the following: (i) Some phonemes have a short duration so they may not appear in the map, especially in visual-only experiments where the frame rate is 25Hz. (ii) Some phonemes may appear more than once in the keyword, meaning the diagonal assumption of the pattern might no longer hold since off-diagonal components may appear.

Loss function. For training we create clip-keyword sample pairs which are labeled positive or negative depending on whether or not the keyword occurs in the

clip (which can contain an arbitrarily long utterance). Given a sample pair, the KWS-Net model outputs a probability $p_t(y = 1|V, P)$ representing how likely the keyword is to occur at every temporal location $t \in [1, t_v^{out}]$. We obtain a sequence-level prediction by taking the maximum probability over all time locations. The optimisation objective is then a binary cross-entropy loss between this prediction and the ground truth sequence-level label y^* (1 for positive sample, 0 otherwise):

$$L_{kws}(V, P, y^*) = -y^* \log \max_t p_t(y = 1|V, P) - (1 - y^*) \log(1 - \max_t p_t(y = 1|V, P)) \quad (2.1)$$

If we have access to the exact word time boundaries then the temporal interval is used as extra supervision to help the model learn to correctly localise keywords within the clip: for positive samples, we calculate the maximum only within those time boundaries where the keyword is known to occur, instead of the full length $[1, t_v^{out}]$. If not stated otherwise, this is the method that we use. The boundaries can be obtained by forced alignment and are included with some datasets (e.g. LRS2 [Chung and Zisserman 2016c]).

Differences to prior work. Here, KWS is converted to an object detection problem where the CNN detects patterns from a similarity map that correspond to alignments between viseme and phonetic sequences. Similar alignments can be detected by word-level HMMs, that typically follow a ‘left-to-right, no skips’ structure. Instead of detecting these patterns with probabilistic models, we employ a CNN and train the whole architecture jointly in an end-to-end manner, leveraging the large size of the datasets (see Table 2.1) and following the recent trend in lip reading state-of-the-art methods (see Section 2.2).

In [Stafylakis and Tzimiropoulos 2018], fixed length word embeddings are obtained from a grapheme (character) to phoneme (G2P) encoder-decoder architecture, using an additional decoder loss to encourage word representations that reflect the pronunciation. Instead, we build variable length word embeddings by directly encoding the phonemes using simply a BiLSTM. This approach has several advantages: (i) it strongly reflects the pronunciation and aligns better with the viseme features, (ii) it offers more control of words with multiple pronunciations, compared to G2P, and finally, (iii) phonemes are more language-independent compared to graphemes, enabling the encoder to be shared between languages.

Audio-only KWS-Net. We design an audio-only variation of the model, that operates on audio waveforms instead of video clips. We extract acoustic features by applying a STFT to the audio clip, with a 32ms window and 10ms hop-length, at a 16 kHz sample rate. The resulting spectrograms are projected to mel-scale, yielding 80-dimensional features. Since the video is sampled at 25 fps (40 ms per frame), every video input frame corresponds to 4 acoustic feature frames. The spectrograms are therefore passed through two strided convolutions to get the acoustic features down to video resolution, achieving a common temporal-scale for both modalities. This subsampling step allows us to keep the overall architecture the same for visual-only, audio-only and audio-visual inputs.

Audio-visual KWS-Net. We employ a late decision audio-visual fusion. In this case, the audio-only and visual-only KWS-Net models are trained separately as explained above. The logits from the output of the CNN classifier from each of the audio-only and visual-only models are then averaged before applying the sigmoid activation, with the weights for each modality chosen according to the best performing value on the validation set. We explore the effect of varying modality weights in the appendix.

2.4 Experiments

Datasets. The audio-visual datasets used are summarised in Table 2.1. LRW [Chung and Zisserman 2016a] consists of single-word utterances from BBC television broadcasts. LRS2 [Chung and Zisserman 2016c; Afouras et al. 2019] and LRS3 [Afouras et al. 2018b] consist of thousands of spoken sentences from BBC and TED/TEDx talks respectively. Both datasets contain samples from multiple viewpoints, however LRS3 is more challenging than LRS2: speakers are pictured from a wider range of viewpoints and with microphones/headsets, while addressing the audience results in more frequent head movements. We also use the French and German subsets of LRS3-Lang¹, collected from TED/TEDx videos following the procedure from [Afouras et al. 2018b], and refer to them as LRS3-Fr and LRS3-De respectively. In Section 2.5, we compare the performance of KWS-Net on LRS3-Fr and LRS3-De with that of LRS3, instead of LRS2, as the datasets come from the same

¹Available at www.robots.ox.ac.uk/~vgg/data/lip_reading/lrs3-lang






Dataset	Split	#Utt.	#Words	#Hours	Vocab.	Examples
LRW	Train-val	514k	514k	165	500	
	Test	25k	25k	8	500	
LRS2	Pre-train	96k	2M	195	41k	
	Train-val	47k	336k	29	18k	
	Test	1.2k	6k	0.5	1.7k	
LRS3	Pre-train	132k	3.9M	444	51k	
	Train-val	32k	358k	30	17k	
	Test	1.3k	10k	1	2k	
LRS3-Fr	Train-val	69k	1M	107	28k	
	Test	1.3k	10.7k	1.2	2.1k	
LRS3-De	Train-val	12k	185k	20	11.2k	
	Test	1.7k	10.6k	1.5	1.9k	

Table 2.1: **Statistics on datasets:** Division of development and test data, number of utterances and word instances, duration, vocabulary size and examples for LRW [Chung and Zisserman 2016b], LRS2 [Chung and Zisserman 2016c; Afouras et al. 2019], LRS3 [Afouras et al. 2018b], LRS3-Fr and LRS3-De datasets.

domain.

We set up our experiments following [Stafylakis and Tzimiropoulos 2018]: for both training and evaluation, we use only keywords pronounced with $n_p \geq 6$ phonemes. Moreover, as we want to evaluate on unseen keywords, we ensure that training and testing are performed on disjoint keyword vocabularies. To that end, we use all the words appearing in the test sets with $n_p \geq 6$ phonemes as evaluation keywords and we remove them from the training vocabulary, i.e. those words are not used in training the keyword encoder. We perform the language generalisation experiments on LRS3, LRS3-Fr, and LRS3-De in the seen and unseen keywords setting, therefore we drop the last constraint: test keywords for these datasets may have been seen during training. For exact details about the size of the resulting train and test keyword vocabulary of every dataset, please refer to the appendix.

Baselines. We have four baselines: three are evaluated on LRS2 and the final one on LRW. As a first baseline we use our implementation of the model of Stafylakis *et al.* [Stafylakis and Tzimiropoulos 2018], which we also pre-train on LRW for fair comparison. This architecture is described fully in the appendix. Our second baseline is a variant of Stafylakis *et al.* [Stafylakis and Tzimiropoulos 2018], where the G2P network is switched to phoneme-to-grapheme (P2G) for a more expressive phonetic word representation.

Our third baseline is the lip reading visual-ASR model from Afouras *et al.* [Afouras et al. 2020], a CTC based model learned through cross-modal distillation, which is currently the state of the art on LRS2 for training only on publicly available data. The implementation code and pre-trained models are obtained from the authors. In order to apply the ASR model to KWS, we follow the method in [Y. He et al. 2017]: rather than only using the best decoding prediction, we extract the n highest scoring hypotheses using a beam search and estimate the posterior probability that the keyword occurs in a clip using Equation (7) in [Y. He et al. 2017].

Our final baseline is the work of Jha *et al.* [Jha et al. 2018], although our methods are not directly comparable as they perform query by example (as opposed to query by string). Their retrieval pipeline uses the LRW test set for querying and the LRW validation set for retrieval over 500 words. It should be noted that their model only works for a closed set of words, for which examples are provided, whereas KWS-Net can be used to spot words unseen during training. We directly compare to the results reported in their paper.

Ablations. We consider three ablations for our visual-only KWS-Net architecture: (i) not using the word time boundaries for training, which we refer to as ‘no LOC’ since this training regime does not explicitly encourage the correct localisation of the keyword, (ii) removing the shortcut phonetic embedding, which we refer to as ‘no SH’, and (iii) switching the BiLSTM keyword encoder for a P2G encoder-decoder, which we refer to as ‘+P2G’.

Pre-training and fine-tuning. We initialise the weights of the ResNet-18 visual feature extractor [Stafylakis and Tzimiropoulos 2017] from a model pre-trained on word-level lip reading (code and weights publicly available from [Afouras et al. 2018a]). This part of the network is kept frozen during training: following the practice of [Afouras et al. 2018a], we pre-compute the features on the entire datasets, then train the rest of the model directly on them to accelerate training. We employ a curriculum training procedure for the rest of the network that consists of two stages: (i) it is initially trained on the training set of LRW. As LRW contains clips of single words, here the model is trained without word time boundaries, (ii) the model is then fine-tuned on the sequence-level datasets.

Test setup. The performance of the models is evaluated on the test set of every dataset, using as queries all the held out test words (see datasets). We look for each query keyword in all the clips of the test set. Note that there is no balancing of positive and negative clips during evaluation: there are one or a few positive clips for a given keyword and the rest are negatives. During testing, in order to obtain fine-grained localisation, we apply the CNN classifier with a stride of one.

Evaluation metrics. The performance is evaluated based on ranking metrics. For every keyword in the test vocabulary, we record the percentage of the total clips containing it that appear in the first N retrieved results, with $N=[1,5,10]$, this is the ‘Recall at N ’ ($R@N$). Note that, since several clips may contain a query word, the maximum $R@1$ is not 100%. The mean average precision (mAP) and equal error rate (EER) are also reported. For each keyword-clip pair, the match is considered correct if the keyword occurs in the clip and the maximum detection probability occurs between the ground truth keyword boundaries. For each experiment, the average and standard deviation of each metric is computed over the last 5 checkpoints once the model has converged (validation loss has not improved for 5 epochs).

Audio noise addition. To investigate the robustness of the audio-only and audio-visual models against loud environments, we train by adding babble noise to the audio 50% of the time with signal-to-noise-ratio (SNR) of 0 dB. Babble noise (interference from people talking simultaneously) is commonly used for audio degradation in audio-visual speech recognition [Afouras et al. 2019; P. Wu et al. 2016] as it is more challenging than other types of environmental noise [Krishnamurthy and Hansen 2009].

2.5 Results

2.5.1 Visual-only KWS-Net

Baselines. As can be seen in Table 2.2, Stafylakis *et al.* G2P [Stafylakis and Tzimiropoulos 2018]* performs worse than the P2G baseline we propose. Compared to Stafylakis *et al.* P2G, KWS-Net significantly improves $R@1$ from 30.0% to 37.9% and mAP from 43.5% to 53.9%, with the EER also decreasing from 6.3%

	R@1	R@5	R@10	mAP	EER
Stafylakis <i>et al.</i> (G2P)*	22.8	49.0	59.1	36.0	8.9
Stafylakis <i>et al.</i> (P2G)	30.0	53.7	65.3	43.5	6.3
Visual-ASR	41.9	53.6	54.5	51.3	-
KWS-Net	37.9 ± 0.3	66.8 ± 0.6	75.6 ± 0.5	53.9 ± 0.3	5.7 ± 0.2
no LOC	37.2 ± 0.8	65.1 ± 0.2	73.7 ± 0.3	53.0 ± 0.6	6.9 ± 0.4
no SH	35.0 ± 0.5	62.4 ± 0.4	72.7 ± 0.9	50.4 ± 0.3	7.5 ± 0.4
+P2G	39.1 ± 0.3	66.2 ± 0.6	75.1 ± 0.4	54.3 ± 0.3	5.9 ± 0.3

Table 2.2: **Visual-only results:** Performance of baselines, visual-only KWS-Net, and ablations on the LRS2 test set. *refers to our implementation of [Stafylakis and Tzimiropoulos 2018] and Stafylakis *et al.* P2G refers to switching G2P to P2G. Visual-ASR denotes our lip reading baseline from [Afouras *et al.* 2020]. KWS-Net refers to our architecture from Section 2.3. no LOC represents not using the keyword time boundaries for training; no SH denotes not concatenating the phonetic embedding shortcut; +P2G denotes using a P2G encoder-decoder instead of a BiLSTM keyword encoder.

to 5.7%.

KWS-Net has a higher R@5 compared to the lip reading visual-ASR baseline (66.8% vs. 53.6%) and a higher mAP (53.9% vs. 51.3%). In fact, over a third of the keywords do not appear at all in the n -best list. KWS-Net has the advantage of retrieving more clips containing a keyword by using a higher R@N. Visual-ASR has a slightly higher R@1 (41.9% vs. 37.9%), but the method benefits from context of surrounding words.

Next, we replicate the test setting from [Jha *et al.* 2018] and calculate their metrics on LRW: we achieve (not shown on the table) a higher P@10 of 77.1% compared to 65.2% and a higher R@10 of 15.4% compared to 13.0% as well as a slightly higher mAP of 57.8% compared to 57.0%. See [Jha *et al.* 2018] for P@10 and R@10 metric definitions; note that R@N is defined differently in their experiments compared to in our work.

Ablations. In Table 2.2, we assess the value of each component of the architecture. For example when using the keyword time boundaries during training (see loss description in Section 2.3), the EER is reduced from 6.9% to 5.7%; however even if our method is trained without this extra annotation, KWS-Net no LOC still outperforms the Stafylakis *et al.* P2G baseline (37.2% vs. 30.0% R@1). Similarly, the value of the phonetic shortcut embedding is shown in the decrease from 7.5% to 5.7% EER. Finally, we carry out an ablation by replacing the BiLSTM (KWS-

Net) with P2G (KWS-Net+P2G), and conclude that the ablation performs overall worse than the original BiLSTM.

Visualisations. In practice, we observe quasi-diagonal patterns in the similarity map visualisations in Figure 2.3, which matches our intuition that viseme and phonetic feature sequences align when the keyword occurs in the clip. As explained in Section 2.3, there might be off-diagonal components due to repeated phonemes. Please refer to the appendix and project webpage for more qualitative examples.

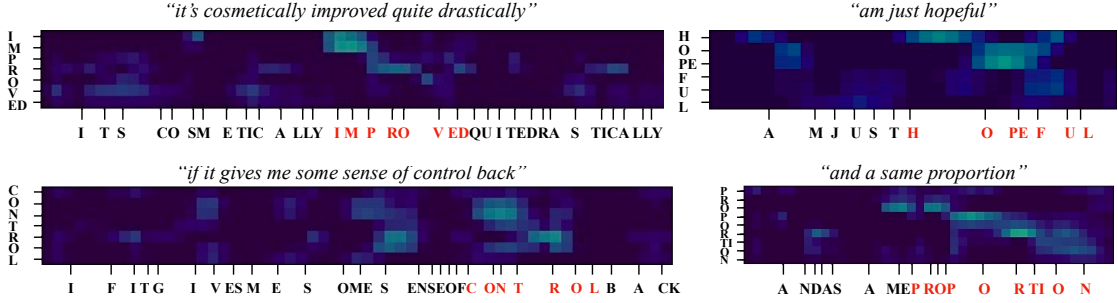


Figure 2.3: **Qualitative results:** Example similarity maps with visual-only KWS-Net for keywords ‘improved’, ‘hopeful’, ‘control’ and ‘proportion’ for clips in the LRS2 test set, with the application of a sigmoid for better visualisation. The vertical axis represents the phonemes in the keyword (graphemes are shown here for simplicity). The horizontal axis corresponds to the visual sequence; for visualisation we add phoneme ground truth start times for the entire clip utterance, with those corresponding to the keyword in red.

Keyword length. We explore how varying the minimum phoneme length of keywords n_p effects the performance of visual-only KWS-Net on the LRS2 test set (see Table 2.3). As n_p increases, the EER decreases and the mAP and R@1 increase as longer keywords are easier to visually spot. For this evaluation, additional shorter words are selected from the original LRS2 test set. Note, the network has not been trained for keywords with $n_p < 6$.

Phrases vs. Keywords. We evaluate visual-only KWS-Net on the LRS2 test set, now using 3 word phrases as queries. For each of the evaluation unseen keywords, we construct a phrase query by concatenating the keyword with its preceding and succeeding words from the clip utterance, resulting in 666 phrases. The R@1 increases from 37.9% to 65.3% (see Table 2.3).

Seen vs. Unseen keywords. We fine-tune our visual-only KWS-Net model, now including the previously unseen keywords from the LRS2 test set that occur in the training set (note there is no overlap between the training and testing videos). As

query type	n_p	vocab.	R@1	R@5	R@10	mAP	EER
unseen words	4	1278	25.8 ± 0.4	50.6 ± 0.5	61.2 ± 0.4	40.4 ± 0.3	11.5 ± 0.2
unseen words	6	644	37.9 ± 0.3	66.8 ± 0.6	75.6 ± 0.5	53.9 ± 0.3	5.7 ± 0.2
unseen words	8	227	53.1 ± 0.9	81.2 ± 0.5	87.1 ± 0.8	68.9 ± 0.3	3.9 ± 0.4
seen words	6	644	39.5 ± 0.6	69.5 ± 0.4	78.9 ± 0.7	56.7 ± 0.6	5.1 ± 0.2
phrases	9	666	65.3 ± 0.9	84.7 ± 0.3	89.1 ± 0.4	74.1 ± 0.6	3.7 ± 0.2

Table 2.3: **Query investigation:** Performance of visual-only KWS-Net on the *extended* LRS2 test set with different query types and minimum phoneme lengths n_p .

seen in Table 2.3, the performance marginally improves for seen words compared to the zero-shot case, showing that our model is robust to words unseen during training (5.7% vs. 5.1% EER).

2.5.2 Audio-visual KWS-Net

We now look at whether we can augment audio with visual information. The results in Table 2.4 indicate that lip movements improve performance even when the audio signal is clean – for example, R@1 increases from 67.7% to 72.2%. When the audio signal is corrupted with noise, the task of audio KWS becomes much harder. This is demonstrated by the decrease in R@1 from 67.7% to 27.6%. However, combining the audio and visual modalities results in a much higher performance, with R@1 increasing from 27.6% to 52.7%. The audio-visual model is more robust, surpassing the performance of both video-only and audio-only KWS-Net with a noisy audio signal, for a range of SNRs (-10 dB to 20 dB), as seen in Figure 2.4.

Mod.	Noise	R@1	R@5	R@10	mAP	EER
V	✗	37.9	66.8	75.6	53.9	5.7
A	✗	67.7	91.1	94.6	83.3	1.9
AV	✗	72.2	94.7	97.0	87.5	1.7
A	✓	27.6	49.8	59.4	39.7	12.8
AV	✓	52.7	81.9	87.0	69.6	4.3

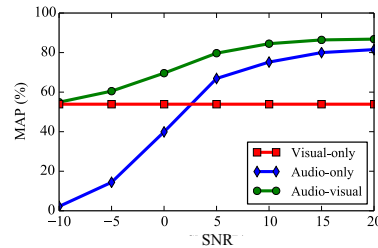


Table 2.4: (Left) **Audio-visual results:** Performance results for visual-only, audio-only and audio-visual KWS-Net on the LRS2 test set with clean audio and in the presence of noise at 0 dB SNR. Standard deviations for this table are given in the appendix. Figure 4: (Right) Mean average precision for visual-only (red), audio-only (blue) and audio-visual (green) KWS-Net with a noisy audio signal, as the SNR is varied between -10 dB and 20 dB.

2.5.3 Extension to other languages: French and German

We now move on to assess the generalisation of our method to other languages. For each of the experiments in Table 2.5, the model is first trained on LRW, then fine-tuned on LRS2 and subsequently LRS3. For LRS3-Fr and LRS3-De, the model is additionally fine-tuned on their corresponding training set. Due to the lack of word timings for LRS3-Fr and LRS3-De, we train the models here without them (see loss description in Section 2.3). During evaluation, we do not consider the location of the maximum keyword detection probability.

The more challenging setting of LRS3 compared to LRS2 (see Section 2.4) is reflected in the visual-only KWS; lip reading is also found to be harder on LRS3 compared to LRS2 [Afouras et al. 2019]. In fact, we split the LRS3 test set into near-frontal and profile views: we find that the model is robust to side views (48.7% mAP) but as expected, the performance is overall better on frontal clips (60.6% mAP).

The performance on LRS3-Fr is close to that on LRS3: the audio-only EER is slightly worse as a lot more English audio from LRW and LRS2 is used for training. The visual-only EER for LRS3-De is higher than LRS3-Fr (13.0% vs. 8.4%). However, LRS3-Fr training set is five times bigger than that of LRS3-De (see Table 2.1). In all cases, the audio-visual model performs better than audio-only and visual-only. The results in Table 2.5 show that KWS-Net can be used for other languages, even if less language specific data is available.

Dataset	Modality	R@1*	R@5*	R@10*	mAP*	EER*
LRS3	V	25.5 \pm 0.4	50.0 \pm 0.5	62.1 \pm 0.3	45.7 \pm 0.3	8.3 \pm 0.3
LRS3	A	52.0 \pm 0.9	88.4 \pm 0.5	94.0 \pm 0.4	85.2 \pm 0.6	2.1 \pm 0.1
LRS3	AV	55.4 \pm 0.9	90.6 \pm 0.2	95.9 \pm 0.2	88.3 \pm 0.4	1.6 \pm 0.1
LRS3-Fr	V	28.8 \pm 0.3	55.3 \pm 0.9	65.8 \pm 0.7	43.9 \pm 0.3	8.4 \pm 0.1
LRS3-Fr	A	52.3 \pm 0.6	86.9 \pm 0.2	92.7 \pm 0.2	72.6 \pm 0.3	3.4 \pm 0.1
LRS3-Fr	AV	53.3 \pm 0.4	88.9 \pm 0.2	93.9 \pm 0.3	74.1 \pm 0.3	3.2 \pm 0.1
LRS3-De	V	13.3 \pm 0.1	33.7 \pm 0.1	43.5 \pm 0.2	24.9 \pm 0.1	13.0 \pm 0.2
LRS3-De	A	48.1 \pm 0.4	79.9 \pm 0.5	88.1 \pm 0.2	67.4 \pm 0.3	3.7 \pm 0.2
LRS3-De	AV	50.5 \pm 0.3	83.3 \pm 0.1	90.2 \pm 0.1	70.3 \pm 0.2	3.4 \pm 0.1

Table 2.5: **Language results:** Performance of visual-only, audio-only and audio-visual KWS-Net on LRS3 (English), LRS3-Fr (French) and LRS3-De (German). *The task here is classifying whether the keyword occurs in the clip, and keywords may be seen during training.

2.6 Conclusion

In this paper, we present a novel CNN-based KWS architecture, KWS-Net, inspired by object detection methods. Our best visual-only model exceeds the performance of the previous state of the art on the LRS2 dataset. We show that combining audio and visual modalities helps KWS for both clean and noisy audio. Finally, we demonstrate that KWS-Net generalises to languages other than English. In future work, we plan to improve KWS-Net by incorporating context of surrounding words.

Acknowledgements. We thank Gül Varol and Olivia Wiles for their helpful comments. Funding for this research is provided by the UK EPSRC CDT in Autonomous Intelligent Machines and Systems, the Oxford-Google DeepMind Graduate Scholarship, and the EPSRC Programme Grant Seebibyte EP/M013774/1.

Chapter 3

Visual Keyword Spotting with Attention

The paper has been accepted for publication at the British Machine Vision Conference (BMVC), 2021.

Visual Keyword Spotting with Attention

K R Prajwal* Liliane Momeni* Triantafyllos Afouras

Andrew Zisserman

Visual Geometry Group, University of Oxford, UK

Abstract

In this paper, we consider the task of spotting spoken keywords in silent video sequences – also known as visual keyword spotting. To this end, we investigate Transformer-based models that ingest two streams, a visual encoding of the video and a phonetic encoding of the keyword, and output the temporal location of the keyword if present. Our contributions are as follows: (1) We propose a novel architecture, the *Transpotter*, that uses full cross-modal attention between the visual and phonetic streams; (2) We show through extensive evaluations that our model outperforms the prior state-of-the-art visual keyword spotting and lip reading methods on the challenging LRW, LRS2, LRS3 datasets by a large margin; (3) We demonstrate the ability of our model to spot words under the extreme conditions of isolated mouthings in sign language videos.

3.1 Introduction

In recent years, there has been significant progress in automatic visual speech recognition (VSR) due to the availability of large-scale annotated datasets and the development of powerful neural network-based learners [Chung et al. 2017; Assael et al. 2016; Afouras et al. 2019]. These methods are continually improving and becoming more sophisticated, by incorporating better visual models, stronger language modelling and training on larger datasets. Indeed the best industrial grade lip reading models today are far superior to humans, and achieve error rates

*Equal contribution.

approaching Automatic Speech Recognition (ASR) performance [Makino et al. 2019; K R et al. 2021].

However, for many applications it is not necessary to transcribe every word that is spoken in a silent video (the task of VSR), rather only specific utterances or keywords need to be recognised. This is for example the case in “wake word” recognition, where only particular keywords need to be spotted over long input sequences. A further drawback of VSR methods is that they are heavily reliant on language modelling; in general, their performance decreases significantly when context is limited (e.g. short utterances) or parts of the input are occluded, e.g. from the speaker’s hands or a microphone. In this work, we focus instead on the task of *Visual Keyword Spotting* (KWS), where the goal is to detect and localise a *given* keyword in (silent) spoken videos.

Automatic visual KWS enables a diverse range of practical applications: indexing archival silent videos by keyword to enable content-based search; helping virtual assistants (e.g. Alexa and Siri) and smart home technologies respond to wake words and phrases; assisting people with speech impairment (e.g. amyotrophic lateral sclerosis patients) or aphonia in communication [Shillingford et al. 2018]; and detecting mouthings in sign language videos [Albanie et al. 2020].

KWS differs in complexity from VSR primarily because in KWS we are armed with the keyword we need to recognise, whereas VSR has the harder task of recognising every word from scratch. The core hypothesis motivating this work is that *this additional knowledge renders visual KWS an easier task* than VSR; and it is therefore expected that KWS should achieve a higher performance than VSR, and generally be more robust to challenging and adversarial situations. Nevertheless, visual KWS remains a *very difficult* task and shares similar challenges to VSR methods: first, some words sound different but involve identical lip movements (‘man’, ‘pan’, ‘ban’), these *homopheme* words cannot be distinguished using only visual information. Second, speech variations such as accents, speed, and mumbling can alter lip movements significantly for the same word. Third, co-articulation of the lips between preceding and subsequent words in continuous speech also affects lip appearance and motion.

In this paper, we make the following three contributions: (i) We propose a novel

Transformer-based architecture, the *Transpotter* (a portmanteau of *Transformer* and *Spotter*), that is tailored to the visual KWS task. The model takes as input two streams, one encoding visual information from a video and the other providing a phonetic encoding of the keyword; the heterogeneous inputs are then fused using full cross-modal attention. (ii) Through extensive evaluations, we show that our Transpotter model outperforms the prior state-of-the-art visual KWS and VSR methods on the challenging LRW, LRS2 and LRS3 lip reading datasets by a large margin. (iii) We test our best model under extreme conditions: finding words in mouthings of people communicating using sign language. Signers sometimes mouth words as they sign as an additional non-manual signal to disambiguate and help understanding [Rachel Sutton-Spence 2007]. This new task is extremely challenging as there is a significant domain shift between full spoken sentences (in our training and test sets) and mouthings, where the context is sporadic and phonemes of the keyword may be missing – as sometimes only parts of words are mouthed [Boyce Braem and RL Sutton-Spence 2001]. Our approach outperforms previous KWS models in this challenging, practical use-case. Video examples are available at the project’s webpage: www.robots.ox.ac.uk/~vgg/research/transpotter.

3.2 Related work

Our work relates to prior work on KWS, lip reading, visual grounding, and applications of Transformers for text and video. We present a brief discussion of these topics below.

KWS. KWS in audio (speech) is a well studied problem with a long history, spanning several decades. Prior to the establishment of deep learning models, KWS methods were based on Hidden Markov Models [Rose and Paul 1990; Wilpon et al. 1989], dynamic time warping [Itakura 1990; Sakoe and Chiba 1978; Yaodong Zhang and Glass 2010] or indexing of ASR lattices [Can and Saraçlar 2011]. A number of works have since used deep architectures suitable for sequence modelling (e.g. RNNs, CNNs, or graph convolutional networks) [Sainath and Parada 2015; Yundong Zhang et al. 2017; Yuxuan Wang et al. 2017; Palaz et al. 2016; X. Chen et al. 2019; Fernandez et al. 2007; Hwang et al. 2015; M. Sun et al. 2016;

Arik et al. 2017; Lengerich and Hannun 2016; Taejun Kim and Nam 2019], including encoder-decoder approaches [Haitong Zhang et al. 2018; Audhkhasi et al. 2017; Zhuang et al. 2016; Rosenberg et al. 2017]. Berg *et al.* [Berg et al. 2021] recently proposed using a Transformer model for the same task. Different from ours, this work uses a single input stream (audio) and only learns to spot a fixed vocabulary of keywords. In contrast, we use Transformers to temporally process, then fuse the multi-modal inputs, building a model that can eventually perform open-set KWS. Visual KWS has also received attention recently. The proposed methods include query-by-example [Jha et al. 2018] approaches, sliding window classification [Yue Yao et al. 2019], or looking up phonetic queries in lip reading feature sequences [Stafylakis and Tzimiropoulos 2018; Momeni et al. 2020a], while audio-visual methods [P. Wu et al. 2016; Ding et al. 2018; Momeni et al. 2020a] that fuse the two modalities to improve robustness to noise have also been proposed. Our method builds upon these approaches: we address various weaknesses and propose superior video-text modelling as well as explicit keyword localization, resulting in significantly improved performance.

Lip reading. Early works in lip reading usually relied on hand-crafted pipelines and features [Potamianos et al. 2003; Gowdy et al. 2004; Papandreou et al. 2009; Ziheng Zhou et al. 2014]. The availability of large scale lip reading datasets [Chung et al. 2017; Afouras et al. 2018b] and the development of deep neural network models resulted in major performance improvements, initially in word-level lip reading [Chung and Zisserman 2016a; Stafylakis and Tzimiropoulos 2017] and constrained sentences [Assael et al. 2016]. Sentence level models were subsequently developed, using sequence-to-sequence architectures based on RNNs [Chung et al. 2017], CTC-based [Shillingford et al. 2018] approaches, or a hybrid of the two [Petridis et al. 2018]. Replacing RNNs with Transformers resulted in better performing architectures [Afouras et al. 2019; X. Zhang et al. 2019; Gulati et al. 2020]. Joint audio-visual training and cross-modal distillation [Afouras et al. 2020; Jianwei Yu et al. 2020; W. Li et al. 2019] have also been investigated. The current state-of-the-art model uses Transformers in the visual front-end and achieves remarkable results with word error rates reaching as low as 30.7% [K R et al. 2021].

Visual grounding. Our work is also related to tasks such as natural language

grounding in videos [Hendricks et al. 2017; Gao et al. 2017; M. Liu et al. 2018; Huijuan Xu et al. 2019; Yuan et al. 2019; Ghosh et al. 2019; Jingyuan Chen et al. 2018; R. Zeng et al. 2020] and subtitle alignment in sign language clips [Bull et al. 2021a].

Transformers. Since their introduction for machine translation, Transformers [Vaswani et al. 2017] have become ubiquitous and are used today in a wide range of applications from natural language processing [Devlin et al. 2019; Radford et al. 2019] and speech recognition [Dong et al. 2018; Karita et al. 2019; Mohamed et al. 2019] to visual representation learning [Dosovitskiy et al. 2021; Bertasius et al. 2021; B. Wu et al. 2020; K R et al. 2021]. In this work, we rely on Transformers as our building blocks for their strong sequence modelling capability and inherent potential for localisation through attention.

3.3 Visual KWS with Attention

In this section, we describe our proposed method shown in Figure 3.1. We outline the architecture of our model (Section 3.3.1), our training procedure (Section 3.3.2) and differences to prior work (Section 3.3.3). We refer the reader to the arXiv version of the paper for further details.

3.3.1 The Transpotter Architecture

Our model ingests two input streams: (i) a textual keyword $q = (q_1, q_2 \dots, q_{n_p})$, and (ii) a silent video clip $v \in \mathbb{R}^{T \times H \times W \times 3}$ in which we need to spot the keyword. For each of the inputs, we have separate encoders that learn initial modality-specific representations. This is followed by a joint multi-modal Transformer that learns cross-modal relationships between the video and text features. The joint transformer predicts two outputs: (i) a sequence-level probability of the keyword occurring in the video and (ii) frame-level probabilities indicating the location of the keyword in the video if present. We describe each of the modules next.

Text Representations. Our textual input is a phonetic representation of the keyword, obtained using a pronunciation dictionary. The input phoneme sequence of length n_p is mapped to a sequence of learnable embedding vectors $Q \in \mathbb{R}^{n_p \times d}$. Sinusoidal positional encodings are added to the input phoneme feature vectors,

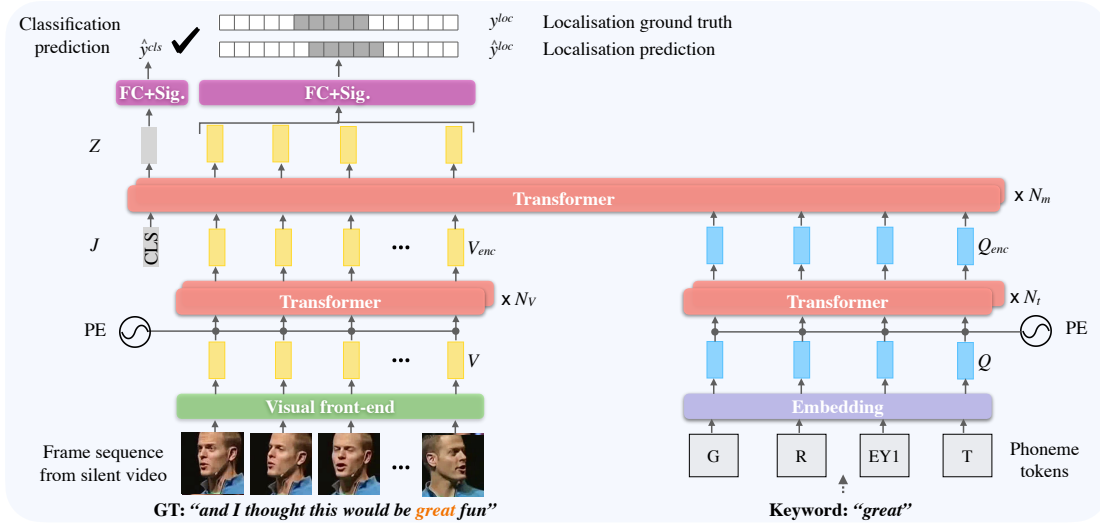


Figure 3.1: **The Transpotter architecture:** Video frames are inputted to a visual front-end (CNN [Afouras et al. 2019] or VTP [K R et al. 2021]) to extract low-level visual features, which are then passed to N_v Transformer layers to encode temporal information. The keyword in the form of a phoneme sequence is encoded using N_t Transformer layers. The text and visual features are finally concatenated in time and processed using a joint multi-modal Transformer which predicts: (i) the probability the keyword occurs in the video, (ii) frame-level probabilities indicating the location of the word. PE corresponds to positional encoding.

and the result is passed through a Transformer Encoder [Vaswani et al. 2017] with N_t layers to capture temporal information across the phoneme sequence:

$$Q_{enc} = \text{encoder}_q(Q + PE_{1:n_p}) \in \mathbb{R}^{n_p \times d}.$$

Video Representations. We use a pre-trained visual front-end (either a CNN [Afouras et al. 2019] or VTP [K R et al. 2021]) to extract a feature vector for each input video frame, $V \in \mathbb{R}^{T \times d}$. Similar to the text encoder Q_{enc} , we pass the visual features through a Transformer Encoder with N_v layers to capture temporal information, after adding positional encodings:

$$V_{enc} = \text{encoder}_v(V + PE_{1:T}) \in \mathbb{R}^{T \times d}.$$

Joint Video-Text Representations. The uni-modal representations V and Q are concatenated along the time dimension to produce a single sequence of feature vectors. A learnable $[CLS]$ token embedding (such as in BERT [Devlin et al. 2019])

and ViT [Dosovitskiy et al. 2021]) is then prepended to the result:

$$J = ([CLS]; V_{enc}; Q_{enc}) \in \mathbb{R}^{(1+T+n_p) \times d}.$$

We use a Transformer encoder with N_m layers to jointly learn the relationships across video and phoneme vectors:

$$Z = \text{encoder}_{vq}(J + PE_{1:(1+T+n_p)}) \in \mathbb{R}^{(1+T) \times d}.$$
¹

Prediction heads. The $[CLS]$ output feature vector Z_1 serves as a joint aggregate representation for the video-text pair. An MLP head for binary classification, f_c is attached to Z_1 to predict the probability of the keyword being present in the video:

$$\hat{y}^{cls} = \sigma(f_c(Z_1)) \in \mathbb{R}^1,$$

where σ denotes a sigmoid activation. To localise the keyword, we attach a second MLP head f_l that is shared across all the video output states from the multi-modal joint Transformer:

$$\hat{y}^{loc} = \sigma(f_l(Z_{2:(T+1)})) \in \mathbb{R}^T.$$

The output y_t^{loc} at each video frame time-step $t \in T$ indicates the probability of the frame t being a part of the keyword utterance.

3.3.2 Training

Optimisation objectives. Given a training dataset \mathcal{D} consisting of tuples (v, q, y^{cls}, y^{loc}) of silent video clips, text queries, class labels and location labels (indicating the position of the keyword within the clip), we define the following objectives:

$$\mathcal{L}^{cls} = -\mathbb{E}_{(v,q,y^{cls}) \in \mathcal{D}} BCE(y^{cls}, \hat{y}^{cls}) \quad (3.1)$$

$$\mathcal{L}^{loc} = -\mathbb{E}_{(v,q,y^{cls},y^{loc}) \in \mathcal{D}} y^{cls} \left[\frac{1}{T} \sum_{t=1}^T BCE(y_t^{loc}, \hat{y}_t^{loc}) \right] \quad (3.2)$$

$$BCE(y, \hat{y}) = y \log \hat{y} + (1 - y) \log(1 - \hat{y}), \quad (3.3)$$

¹the n_p outputs corresponding to the phonetic embeddings are dropped.

where BCE stands for the binary cross-entropy loss. The labels y^{cls} are set to 1 when the given keyword occurs in the video and 0 otherwise; the frame labels y^{loc} are set to 1 for the frames where the keyword is uttered and 0 otherwise. To train the model we optimise the total loss $\mathcal{L} = \lambda\mathcal{L}^{cls} + (1 - \lambda)\mathcal{L}^{loc}$, where λ is a balancing hyper-parameter.

3.3.3 Discussion

Compared to prior approaches, the design of our model offers several important advantages.

Stronger Visual Representations. Previous works [Stafylakis and Tzimiropoulos 2018; Momeni et al. 2020a] model temporal relationships between video frames using RNNs. In contrast, we employ Transformers [Vaswani et al. 2017], which are far more effective in modeling temporal relationships [Al-Rfou et al. 2019; Hochreiter et al. 2001].

Joint Video-text Modeling. Prior works such as KWS-Net [Momeni et al. 2020a] follow a late-fusion strategy. In our model each frame-wise video feature can attend to any keyword token (phoneme) and vice-versa. The information exchange across the modalities occurs at every layer, without restrictions on the receptive field for either modality.

Stronger keyword localisation. Fine-grained localisation of the keyword in the video can be important for applications such as sign spotting [Albanie et al. 2020]. Existing methods [Momeni et al. 2020a; Stafylakis and Tzimiropoulos 2018] “weakly” localise the keyword by taking the sequence-level prediction to be the maximum probability over all the video time-steps. We instead provide stronger frame-level supervision, by enforcing the model to predict the exact temporal extent of the keyword in the video.

3.3.4 Implementation details

Pre-training the visual front-end. We explore two different visual front-end architectures for the Transpotter: (1) a CNN, highly similar in architecture to TM-seq2seq [Afouras et al. 2019] and (2) VTP [K R et al. 2021], the current state-of-the-art for lip reading (trained only on public data). Both models are trained

end-to-end on two-word video clips of LRS2 [Chung et al. 2017] and LRS3 [Afouras et al. 2018b] for lip reading. We refer the reader to the arXiv version of the paper for the exact CNN architecture and training hyper-parameters. We refer the reader to [K R et al. 2021] for architectural hyper-parameters and training protocols for VTP. We pre-compute the visual features for each backbone for both datasets and then train directly on them for faster training. All our models and ablations use the pre-trained CNN features, unless otherwise stated.

Sampling. We form the training dataset \mathcal{D} by randomly sampling with 50% probability a positive or negative video clip v for each query q . Each video v contains word boundary annotations, which allows (i) performing data augmentation by randomly cropping video segments during training, and (ii) creating frame labels y^{loc} , as described in 3.3.2.

Misc. The keyword q is mapped to a phoneme sequence using the CMU dictionary [Speech Group at Carnegie Mellon University 2014]; words not present in the dictionary are discarded from training \mathcal{D} . We set $\lambda = 0.5$.

3.4 Experiments

This section is structured as follows: We first present the datasets used as well the evaluation protocols that we follow in our experiments (Section 3.4.1). Next, we compare the performance of our proposed Transpotter model against strong baselines (Section 3.4.2) and then present a comprehensive study ablating our design choices (Section 3.4.3). Finally, we perform further performance analysis and provide qualitative results (Section 3.4.4).

3.4.1 Datasets and Evaluation Protocol

Datasets. All models and baselines are trained and evaluated on LRS2 [Chung et al. 2017] and LRS3 [Afouras et al. 2018b] lip reading datasets. LRS2 contains BBC broadcast footage from British television and LRS3 is based on TED/TEDx videos downloaded from YouTube (refer to the arXiv version of the paper for detailed statistics). The video clips for both datasets are tightly cropped face-tracks of active speakers only. For each clip, a full transcription of the utterance as well as word boundary alignments are provided. The number of videos, number

of keyword instances and keyword vocabulary for each of the test sets is shown in Table 3.1.

Evaluation Protocol. Evaluation is performed for every test dataset as follows: First, the vocabulary of test keywords is determined, by considering all the words occurring in the test set transcriptions with above a certain phoneme length n_p . If not specified, we use $n_p \geq 3$. Every word in the query vocabulary is then searched for in all the test set videos.

Metrics. Given ground truth video-keyword samples, we assess the performance of our model in two ways. First, we assess classification performance, *i.e.* whether the model can accurately predict whether the keyword occurs in the video or not. We compute accuracy ($\text{Acc}_{@k}^{Cls}$) and mean average precision (mAP^{Cls}) metrics, where $\text{Acc}_{@k}^{Cls}$ measures how often a given keyword occurs in any of the top- k retrievals, and mAP^{Cls} is obtained with the above criterion (where every word in the test keyword vocabulary is considered as a separate class).

Second, we assess the model’s localisation capability, *i.e.* whether the model can accurately localise the keyword in the video clip. We follow common practice from the detection literature: we consider a keyword accurately detected when the intersection-over-union (IOU) between the prediction \hat{y}^{loc} and ground truth label y^{loc} is above a certain threshold, and calculate the mean average precision mAP^{Loc} . To calculate the IOU, we binarise the model’s predictions using a threshold $\tau = 0.5$.

3.4.2 Comparison to baselines

We compare our model’s performance against a state-of-the-art VSR model and KWS-Net [Momeni et al. 2020a], the previous state-of-the-art visual KWS model.

VSR baseline. We use an improved version of the TM-seq2seq [Afouras et al. 2019] VSR model, with the same pre-trained CNN backbone (Section 3.3.4) that we use for the KWS models. The model is trained with the curriculum training strategy of [Afouras et al. 2019] (details in the arXiv version of the paper). The VSR model achieves state-of-the-art Word Error Rate (WER) performance of 36.9% and 48.0% on the LRS2, LRS3 test sets respectively. Since the VSR model only produces text transcriptions of a given video, but no localisation prediction, we can only evaluate its classification performance ($\text{Acc}_{@k}^{Cls}, \text{mAP}^{Cls}$). We follow

the method detailed in [Y. He et al. 2017] to estimate the posterior probability that the keyword occurs in a video clip.

KWS-Net. As a KWS baseline we use the state-of-the-art model of Momeni *et al.* [Momeni et al. 2020a]. For fair comparison, here too we use the same CNN backbone that is also used for our model.

	LRS2				LRS3			
	1.2K vids. / 4.3K inst. / 1.6K vocab.				1.3K vids. / 6.1K inst. / 1.9K vocab.			
Model	Acc _{@1} ^{Cls}	Acc _{@5} ^{Cls}	mAP ^{Cls}	mAP ^{Loc}	Acc _{@1} ^{Cls}	Acc _{@5} ^{Cls}	mAP ^{Cls}	mAP ^{Loc}
KWS-Net [Momeni et al. 2020a]	36.1	61.2	41.0	36.2	29.8	54.6	34.3	29.2
VSR	63.7	76.3	64.3	-	52.3	66.0	50.3	-
Transpotter	65.0	87.1	69.2	68.3	52.0	77.1	55.4	53.6
Transpotter (VTP)	68.7	90.7	72.5	71.6	55.7	78.5	58.2	56.1

Table 3.1: **Comparison to baselines:** We outperform the current state-of-the-art KWS and VSR methods by a large margin. Our Transpotter model is particularly effective in localising the keyword in the video. Moreover, by using the recently proposed VTP [K R et al. 2021] architecture as the Transpotter’s visual backbone instead of a CNN, we achieve even better performance.

State-of-the-art KWS. We report our model’s performance and compare it with strong baselines in Table 3.1. It is clear that our model outperforms both baselines. On the last row, we show the boost in performance by replacing the CNN with the recently proposed VTP backbone [K R et al. 2021], resulting in state-of-the-art performance on both the LRS2 and LRS3 datasets.

Evaluation on LRW. We also compare the performance of KWS-Net [Momeni et al. 2020a] with our proposed Transpotter model on the LRW [Chung and Zisserman 2016a] test set following the same evaluation protocol. The test set contains 25K single-word video clips spanning a vocabulary of 500 words (50 instances per word). Note that KWS-Net has been pretrained on the LRW training split, but the Transpotter has only been trained on LRS2 and LRS3. As we can see in Table 3.2, the Transpotter outperforms the previous state-of-the-art baseline KWS-Net by a large margin. We refer the reader to the arXiv version of the paper for a qualitative error analysis in this setting.

3.4.3 Architecture ablations

To assess our design choices for the Transformer skeleton, we perform a number of ablations considering variations of the model architecture. We briefly explain

Model	$\text{Acc}_{@1}^{Cls}$	$\text{Acc}_{@5}^{Cls}$	mAP^{Cls}
KWS-Net [Momeni et al. 2020a]	66.6	89.0	33.0
Transpotter	85.8	99.6	64.1

Table 3.2: **Comparison on LRW [Chung and Zisserman 2016a]:** The Transpotter outperforms the previous state-of-the-art KWS model on the LRW test set, despite not having been trained on LRW data. The localization metric mAP^{Loc} is not reported as the input videos are single-word clips.

the alternative approaches below; more details can be found in the arXiv version of the paper.

In particular we consider two alternative encoder-decoder architectures, with the video input at the encoder side and the text query at the decoder ($\text{Enc}_{vid}\text{-Dec}_{text}$) and vice versa ($\text{Enc}_{text}\text{-Dec}_{vid}$). Since the latter model outputs at the temporal resolution of the video input, it can explicitly localise the keyword (in the same way as the Transpotter), while the former can only perform classification. We also consider a variant of the Transpotter, where the model does not output localisation predictions (hence no \mathcal{L}^{loc} is used for its training). We show the results in Table 3.3. The selected Transpotter architecture outperforms all variants. In particular, by comparing rows 2 and 4, we observe that training with a localisation head and loss \mathcal{L}^{loc} also improves classification (e.g. 64.0 vs 69.2 mAP^{Cls}).

	LRS2				LRS3			
Model	$\text{Acc}_{@1}^{Cls}$	$\text{Acc}_{@5}^{Cls}$	mAP^{Cls}	mAP^{Loc}	$\text{Acc}_{@1}^{Cls}$	$\text{Acc}_{@5}^{Cls}$	mAP^{Cls}	mAP^{Loc}
$\text{Enc}_{vid}\text{-Dec}_{text}$	52.5	80.0	57.9	-	40.3	66.9	43.2	-
Transpotter w/o loc.	59.4	84.1	64.0	-	46.5	72.1	49.8	-
$\text{Enc}_{text}\text{-Dec}_{vid}$	63.8	86.8	68.4	67.8	52.1	76.6	54.9	53.1
Transpotter	65.0	87.1	69.2	68.3	52.0	77.1	55.4	53.6

Table 3.3: **Model ablations:** Our approach of jointly modeling text and video sequences with a localisation head for stronger supervision outperforms other architectural designs.

3.4.4 Transpotter performance analysis

In this section, we analyse the performance of our proposed method when varying the keyword length and the size of the surrounding visual context.

Keyword length. In Figure 3.2a, we plot the model’s performance on the LRS2 test set against the minimum keyword length in phonemes n_p . As expected, longer keywords are easier to spot and therefore result in better retrieval performance. Indeed for long 7-phoneme keywords, mAP^{Loc} reaches as high as 82.5. We note

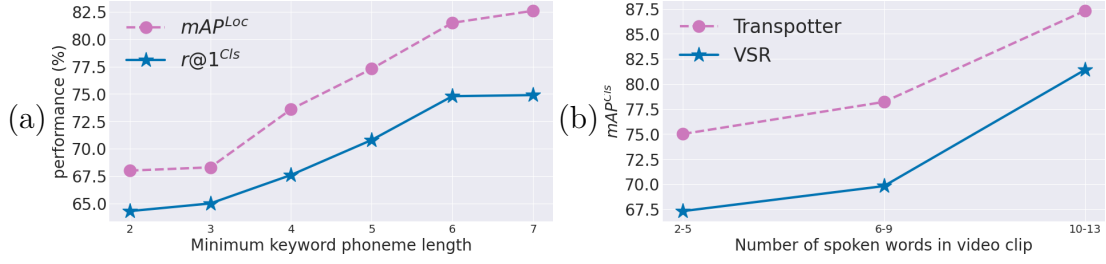


Figure 3.2: (a) Transpotter’s performance increases with the keyword length; (b) Transpotter performs far better than VSR with limited context. Both methods improve with more context.

however that even for very challenging short keywords with only 2 phonemes (such as "my", "to", "at"), mAP^{Loc} stays high at 67.5.

Context. The visual appearance of spoken words can be highly ambiguous [Afouras et al. 2019], therefore recognising isolated words from visual input alone may be very challenging. Current lip reading models utilise the surrounding visual context to resolve this ambiguity. In Figure 3.2b, we illustrate how the performances of our Transpotter KWS model and our VSR baseline vary based on the amount of contextual information available. We plot the mAP^{Cls} against the number of words in the video clip. We observe that both models benefit from larger surrounding context, with the Transpotter outperforming the VSR baseline consistently.

Qualitative analysis. In Figure 3.3, we show qualitative examples from the LRS2 and LRS3 test sets. It is clear that the model produces smooth predictions that precisely indicate the full location of the word. In the bottom right corner we observe a failure case where the model’s confidence is low – the keyword “that’s” in this case is short.

Model response to homophemes. We further probe our Transpotter model for failure cases. In visual-only keyword spotting, a common failure case is due to homophemes, *i.e.* words with identical lip movements. To investigate the response of our model to such cases, we construct a list of keywords from the LRS2 test set sentences that are known to have homophone counterparts (*e.g.* *mark*, which has two matching homophemes, *bark* and *park*) and then for each test set clip that contains one of the keywords, we query that keyword along with its corresponding homophemes and plot the model’s outputs. We illustrate several examples in Figure 3.4. We observe that in such cases, the model spots the keyword as well as its homophemes at the same (ground truth) location.

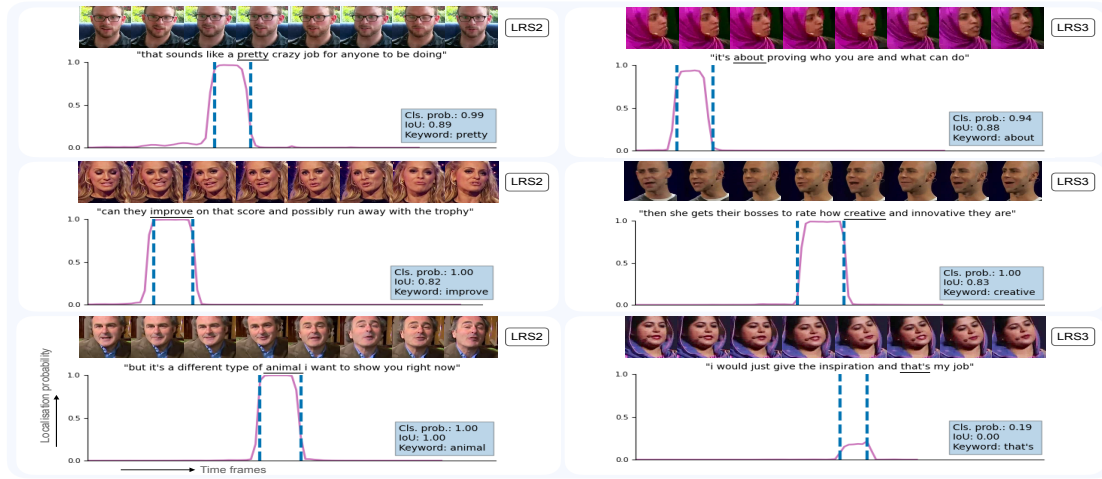


Figure 3.3: **Qualitative results on LRS2 and LRS3:** The Transpotter accurately localises the keyword in most examples. In the bottom right example, the model’s confidence is low, most likely because it is a short word. The IOU is zero since we threshold at $\tau = 0.5$.

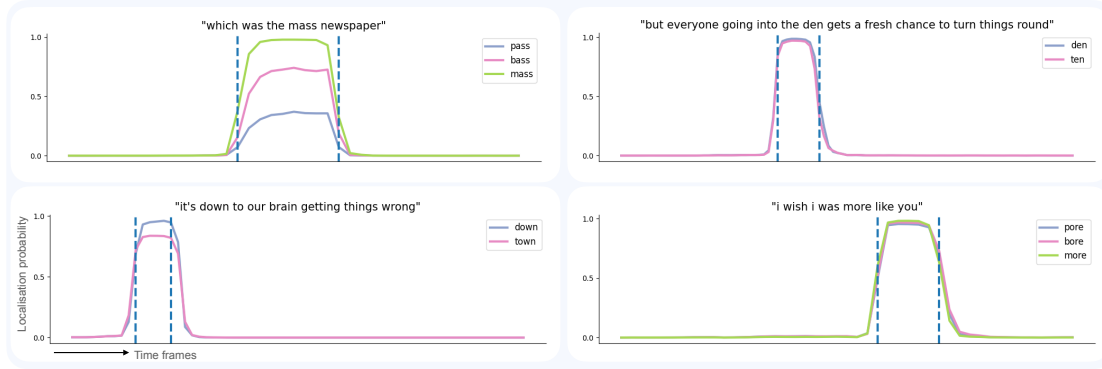


Figure 3.4: **Model’s response to homophemes:** We query words and their corresponding homophemes for LRS2 test set clips. We observe that the model spots the words and their homophemes at the same (ground truth) location.

3.5 Mouthing Spotting in Sign Language videos

In this section, we investigate the application of our method for spotting mouthed words in sign language videos. This is an important application of visual KWS, as it has enabled an entire line of work on sign language recognition [Albanie et al. 2020; Momeni et al. 2020b; Varol et al. 2021].

Data description & evaluation protocol. Here, we use a subset of BSL Corpus [Schembri et al. 2013; Schembri et al. 2017] as a test set. BSL Corpus is a large public dataset containing videos of conversations conducted in sign language by deaf signers, from various regions across the UK. We extend the dataset’s annotations by adding a *Mouthing* tier and asking a deaf annotator to identify and

localise mouthing occurrences that correspond to visible signs. We obtain 383 mouthing instances, from 29 different signers, over a keyword vocabulary size of 187. We use a pre-processing pipeline similar to [Chung et al. 2017] to obtain face-cropped tracks around the faces of the signers. To evaluate KWS performance, we take 8-second video clips centered around the annotated mouthings and follow the same evaluation protocol described in Section 3.4.

Results. We summarise the evaluation results in Table 3.4. The Transpotter model is far superior to the prior state-of-the-art KWS baseline, achieving a great improvement in performance (e.g. 29.6 vs 15.6 mAP^{Cls} score). To complete this analysis, we also show qualitative examples of the spotted mouthings in Figure 3.5.

Discussion. We note that sign language mouthings are often very different from equivalent spoken words. Words may be partially mouthed and can be occluded by the signing hands. There is therefore a significant domain gap between the BSL-Corpus signing videos and our lip reading training videos. However, we note that our proposed model greatly outperforms the KWS-Net baseline – a variant of which has been successfully deployed for detecting mouthings in order to bootstrap learning of sign spotting methods [Albanie et al. 2020; Momeni et al. 2020b; Varol et al. 2021]. This indicates the potential of our proposed method to greatly improve these pipelines.

Model	$\text{Acc}_{@1}^{Cls}$	$\text{Acc}_{@5}^{Cls}$	mAP^{Cls}
KWS-Net [Momeni et al. 2020a]	12.4	29.6	15.6
Transpotter	22.5	47.1	29.6

Table 3.4: **Spotting mouthings in BSL-Corpus:** The Transpotter is far more accurate than the current state-of-the-art in spotting keywords in videos.

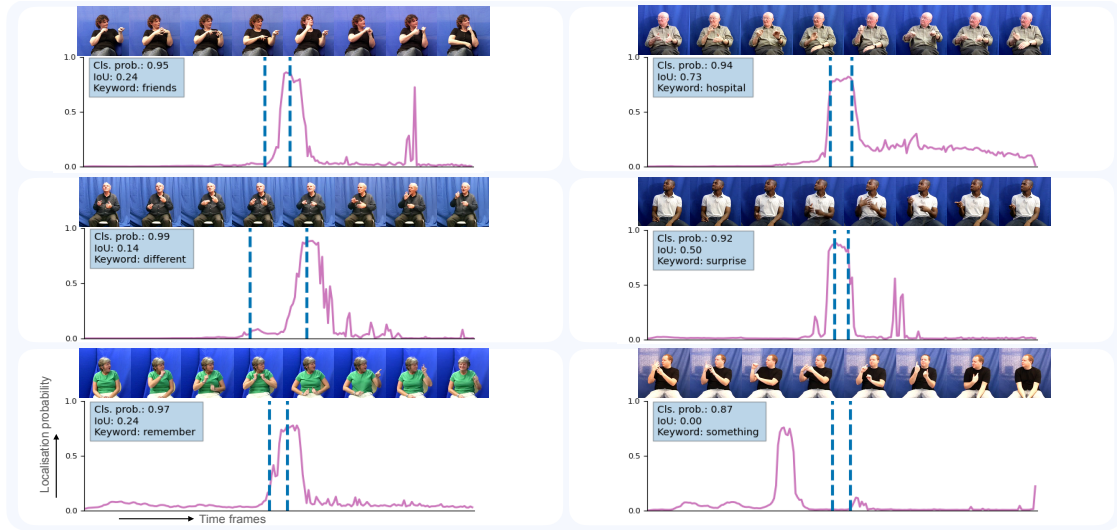


Figure 3.5: **Qualitative results on BSL-Corpus:** Despite the large domain shift from our training examples and additional challenges such as partially mouthed words and hand occlusions, the *Transpotter* succeeds in correctly spotting mouthings in these challenging conditions. We observe a failure case (bottom right) where the localisation is incorrect. We note that contrary to LRS2 and LRS3, where word boundaries are obtained through robust audio-based forced alignment, the annotations for BSL-Corpus are noisier as they are performed manually.

3.6 Conclusion

We have presented the *Transpotter*, a cross-modal attention based architecture for visual keyword spotting. Our method surpasses the performance of the previous best visual keyword spotting approach by a large margin, as well as that of a state-of-the-art lip reading baseline. We demonstrate the ability of our model to generalise to sign language videos where it can be used to spot mouthings, enabling automatic annotation of sign instances. In future work, we plan to further improve our method’s performance by incorporating keyword semantics and context of the surrounding words.

Acknowledgements. Funding for this research is provided by the UK EPSRC CDT in Autonomous Intelligent Machines and Systems, the Oxford-Google DeepMind Graduate Scholarship, the EPSRC Programme Grant VisualAI (EP/T028572/1) and the Royal Society Research Professorships 2019 RP/R1/191132. We thank Samuel Albanie for his invaluable help in applying our method to signer mouthings.

Chapter 4

BSL-1K: Scaling up co-articulated sign language recognition using mouthing cues

The paper has been accepted for publication at the European Conference on Computer Vision (ECCV), 2020.

BSL-1K: Scaling up co-articulated sign language recognition using mouthing cues

Samuel Albanie^{1*} Gül Varol^{1*} Liliane Momeni¹
Triantafyllos Afouras¹ Joon Son Chung² Neil Fox³
Andrew Zisserman¹

¹ Visual Geometry Group, University of Oxford, UK

² Naver Corporation, Seoul, South Korea

³ Deafness, Cognition and Language Research Centre, UCL, UK

Abstract

Recent progress in fine-grained gesture and action classification, and machine translation, point to the possibility of automated sign language recognition becoming a reality. A key stumbling block in making progress towards this goal is a lack of appropriate training data, stemming from the high complexity of sign annotation and a limited supply of qualified annotators. In this work, we introduce a new scalable approach to data collection for sign recognition in continuous videos. We make use of weakly-aligned subtitles for broadcast footage together with a keyword spotting method to automatically localise sign-instances for a vocabulary of 1,000 signs in 1,000 hours of video. We make the following contributions: (1) We show how to use mouthing cues from signers to obtain high-quality annotations from video data—the result is the BSL-1K dataset, a collection of British Sign Language (BSL) signs of unprecedented scale; (2) We show that we can use BSL-1K to train strong sign recognition models for co-articulated signs in BSL and that these models additionally form excellent pretraining for other sign languages and benchmarks—we exceed the state

*Equal contribution.

of the art on both the MSASL and WLASL benchmarks. Finally, (3) we propose new large-scale evaluation sets for the tasks of *sign recognition* and *sign spotting* and provide baselines which we hope will serve to stimulate research in this area.

4.1 Introduction

With the continual increase in the performance of human action recognition there has been a renewed interest in the challenge of recognising sign languages such as American Sign Language (ASL), British Sign Language (BSL), and Chinese Sign Language (CSL). Although in the past isolated sign recognition has seen some progress, recognition of continuous sign language remains extremely challenging [N. C. Camgoz et al. 2018]. Isolated signs, as in dictionary examples, do not suffer from the *naturally* occurring complication of co-articulation (i.e. transition motions) between preceding and subsequent signs, making them visually very different from continuous signing. If we are to recognise ASL and BSL performed *naturally* by signers, then we need to recognise co-articulated signs.

Similar problems were faced by Automatic Speech Recognition (ASR) and the solution, as always, was to learn from very large scale datasets, using a parallel corpus of speech and text. In the vision community, a related path was taken with the modern development of automatic lip reading: first isolated words were recognised [Chung and Zisserman 2016a], and later sentences were recognised [Chung et al. 2017]—in both cases tied to the release of large datasets. The objective of this paper is to design a scalable *method* to generate large-scale datasets of continuous signing, for training and testing sign language recognition, and we demonstrate this for BSL. We start from the perhaps counter-intuitive observation that signers often mouth the word they sign simultaneously, as an additional signal [Bank et al. 2011; Rachel Sutton-Spence and Woll 1999; Rachel Sutton-Spence 2007], performing similar lip movements as for the spoken word. This differs from mouth gestures which are not derived from the spoken language [Crasborn et al. 2008]. The mouthing helps disambiguate between different meanings of the same manual sign [Woll 2001] or in some cases simply provides redundancy. In this way, a sign is not only defined by the hand movements and hand shapes, but also by facial expressions and mouth movements [Cooper et al. 2011].

We harness word mouthings to provide a method of automatically annotating continuous signing. The key idea is to exploit the readily available and abundant supply of sign-language translated TV broadcasts that consist of an overlaid interpreter performing signs and subtitles that correspond to the audio content. The availability of subtitles means that the annotation task is in essence one of alignment between the words in the subtitle and the mouthings of the overlaid signer. Nevertheless, this is a *very* challenging task: a continuous sign may last for only a fraction (e.g. 0.5) of a second, whilst the subtitles may last for several seconds and are not synchronised with the signs produced by the signer; the word order of the English need not be the same as the word order of the sign language; the sign may not be mouthed; and furthermore, words may not be signed or may be signed in different ways depending on the context. For example, the word “fish” has a different visual sign depending on referring to the animal or the food, introducing additional challenges when associating subtitle words to signs.

To detect the mouthings we use *visual keyword spotting*—the task of determining *whether* and *when* a keyword of interest is uttered by a talking face using *only* visual information—to address the alignment problem described above. Two factors motivate its use: (1) direct lip reading of arbitrary isolated mouthings is a fundamentally difficult task, but searching for a particular known word within a short temporal window is considerably less challenging; (2) the recent availability of large scale video datasets with aligned audio transcriptions [Afouras et al. 2019; Chung and Zisserman 2016c] now allows for the training of powerful visual keyword spotting models [Stafylakis and Tzimiropoulos 2018; Yue Yao et al. 2019; Jha et al. 2018] that, as we show in the experiments, work well for this application.

We make the following contributions: (1) we show how to use visual keyword spotting to recognise the mouthing cues from signers to obtain high-quality annotations from video data—the result is the BSL-1K dataset, a large-scale collection of BSL (British Sign Language) signs with a 1K sign vocabulary; (2) We show the value of BSL-1K by using it to train strong sign recognition models for co-articulated signs in BSL and demonstrate that these models additionally form excellent pretraining for other sign languages and benchmarks—we exceed the state of the art on both the MSASL and WLASL benchmarks with this approach; (3) We propose new evaluation datasets for *sign recognition* and *sign spotting* and provide baselines for

each of these tasks to provide a foundation for future research¹.

4.2 Related Work

Sign language datasets. We begin by briefly reviewing public benchmarks for studying automatic sign language recognition. Several benchmarks have been proposed for American [Athitsos et al. 2008; Joze and Koller 2019; D. Li et al. 2019; Wilbur and Kak 2006], German [Koller et al. 2015b; von Agris et al. 2008], Chinese [Chai et al. 2014; J. Huang et al. 2018b], and Finnish [Viitaniemi et al. 2014] sign languages. BSL datasets, on the other hand, are scarce. One exception is the ongoing development of the linguistic corpus [Schembri et al. 2017; Schembri et al. 2013] which provides fine-grained annotations for the atomic elements of sign production. Whilst its high annotation quality provides an excellent resource for sign linguists, the annotations span only a fraction of the source videos so it is less appropriate for training current state-of-the-art data-hungry computer vision pipelines.

Tab. 4.1 presents an overview of publicly available datasets, grouped according to their provision of *isolated* signs or *co-articulated* signs. Earlier datasets have been limited in the size of their video instances, vocabularies, and signers. Within the isolated sign datasets, Purdue RVL-SLLL [Wilbur and Kak 2006] has a limited vocabulary of 104 signs (ASL comprises more than 3K signs in total [Valli and University 2005]). ASLLVD [Athitsos et al. 2008] has only 6 signers. Recently, MSASL [Joze and Koller 2019] and WLASL [D. Li et al. 2019] large-vocabulary isolated sign datasets have been released with 1K and 2K signs, respectively. The videos are collected from lexicon databases and other instructional videos on the web.

Due to the difficulty of annotating co-articulated signs in long videos, continuous datasets have been limited in their vocabulary, and most of them have been recorded in lab settings [J. Huang et al. 2018b; von Agris et al. 2008; Wilbur and Kak 2006]. RWTH-Phoenix [Koller et al. 2015b] is one of the few realistic datasets that supports training complex models based on deep neural networks. A recent extension also allows studying sign language translation [N. C. Camgoz et al. 2018].

¹The project page is at: <https://www.robots.ox.ac.uk/~vgg/research/bsl1k/>

Table 4.1: **Summary of previous public sign language datasets:** The BSL-1K dataset contains, to the best of our knowledge, the largest source of annotated sign data in any dataset. It comprises of co-articulated signs outside a lab setting.

Dataset	lang	co-articulated	#signs	#annos (avg. per sign)	#signers	source
ASLLVD [Athitsos et al. 2008]	ASL	✗	2742	9K (3)	6	lab
Devisign [Chai et al. 2014]	CSL	✗	2000	24K (12)	8	lab
MSASL [Joze and Koller 2019]	ASL	✗	1000	25K (25)	222	lexicons, web
WLASL [D. Li et al. 2019]	ASL	✗	2000	21K (11)	119	lexicons, web
S-pot [Viitaniemi et al. 2014]	FinSL	✓	1211	4K (3)	5	lab
Purdue RVL-SLLL [Wilbur and Kak 2006]	ASL	✓	104	2K (19)	14	lab
Video-based CSL [J. Huang et al. 2018b]	CSL	✓	178	25K (140)	50	lab
SIGNUM [von Agris et al. 2008]	DGS	✓	455	3K (7)	25	lab
RWTH-Phoenix [Koller et al. 2015b; N. C. Camgoz et al. 2018]	DGS	✓	1081	65K (60)	9	TV
BSL Corpus [Schembri et al. 2013]	BSL	✓	5K	50K (10)	249	lab
BSL-1K	BSL	✓	1064	273K (257)	40	TV

However, the videos in [N. C. Camgoz et al. 2018; Koller et al. 2015b] are only from weather broadcasts, restricting the domain of discourse. In summary, the main constraints of the previous datasets are one or more of the following: (i) they are limited in size, (ii) they have a large total vocabulary but only of isolated signs, or (iii) they consist of natural co-articulated signs but cover a limited domain of discourse. The BSL-1K dataset provides a considerably greater number of annotations than all previous public sign language datasets, and it does so in the co-articulated setting for a large domain of discourse.

Sign language recognition. Early work on sign language recognition focused on hand-crafted features computed for hand shape and motion [Ali Farhadi et al. 2007; Fillbrandt et al. 2003; Starner 1995; Tamura and Kawasaki 1988]. Upper body and hand pose have then been widely used as part of the recognition pipelines [Buehler et al. 2009; N. C. Camgoz et al. 2017; Cooper et al. 2011; Ong et al. 2012; Pfister et al. 2014]. Non-manual features such as face [Ali Farhadi et al. 2007; Koller et al. 2015b; T. D. Nguyen and Ranganath 2008], and mouth [Antonakos et al. 2015; Koller et al. 2014a; Koller et al. 2015c] shapes are relatively less considered. For sequence modelling of signs, HMMs [A. Farhadi and D. Forsyth 2006; Agris et al. 2008; Forster et al. 2013; Starner 1995], and more recently LSTMs [N. C. Camgoz et al. 2017; J. Huang et al. 2018b; Ye et al. 2018; H. Zhou et al. 2020b], have been utilised. Koller et al. [Koller et al. 2017] present a hybrid approach based on CNN-RNN-HMM to iteratively re-align sign language videos to the sequence of sign annotations. More recently 3D CNNs have been adopted due to their representation capacity for spatio-temporal data [Bilge et al. 2019; N. C. Camgoz et al. 2016; J. Huang et al. 2015; Joze and Koller 2019; D. Li et al. 2019]. Two recent concurrent works [Joze and Koller 2019; D. Li et al. 2019] showed that I3D mod-

els [João Carreira and Zisserman 2017] significantly outperform their pose-based counterparts. In this paper, we confirm the success of I3D models, while also showing improvements using pose distillation as pretraining. There have been efforts to use sequence-to-sequence translation models for sign language translation [N. C. Camgoz et al. 2018], though this has been limited to the weather discourse of RWTH-Phoenix, and the method is limited by the size of the training set. The recent work of [D. Li et al. 2020b] localises signs in continuous news footage to improve an isolated sign classifier.

In this work, we utilise mouthings to localise signs in weakly-supervised videos. Previous work [Buehler et al. 2009; Cooper and Bowden 2009; Pfister et al. 2014; Chung and Zisserman 2016c] has used weakly aligned subtitles as a source of training data, and both one-shot [Pfister et al. 2014] (from a visual dictionary) and zero-shot [Bilge et al. 2019] (from a textual description) have also been used. Though no previous work, to our knowledge, has put these ideas together. The sign spotting problem was formulated in [Eng-Jon Ong et al. 2014; Viitaniemi et al. 2014].

Using the mouth patterns. The mouth has several roles in sign language that can be grouped into spoken components (mouthings) and oral components (mouth gestures) [Woll 2001]. Several works focus on recognising mouth shapes [Antonakos et al. 2015; Koller et al. 2015c] to recover mouth gestures. Few works [Koller et al. 2014a; Koller et al. 2014b] attempt to recognise mouthings in sign language data by focusing on a few categories of visemes, i.e., visual correspondences of phonemes in the lip region [Fisher 1968]. Most closely related to our work, [Pfister et al. 2013] similarly searches subtitles of broadcast footage and uses the mouth as a cue to improve alignment between the subtitles and the signing. Two key differences between our work and theirs are: (1) we achieve precise localisation through keyword spotting, whereas they only use an open/closed mouth classifier to reduce the number of candidates for a given sign; (2) scale—we gather signs over 1,000 hours of signing (in contrast to the 30 hours considered in [Pfister et al. 2013]).

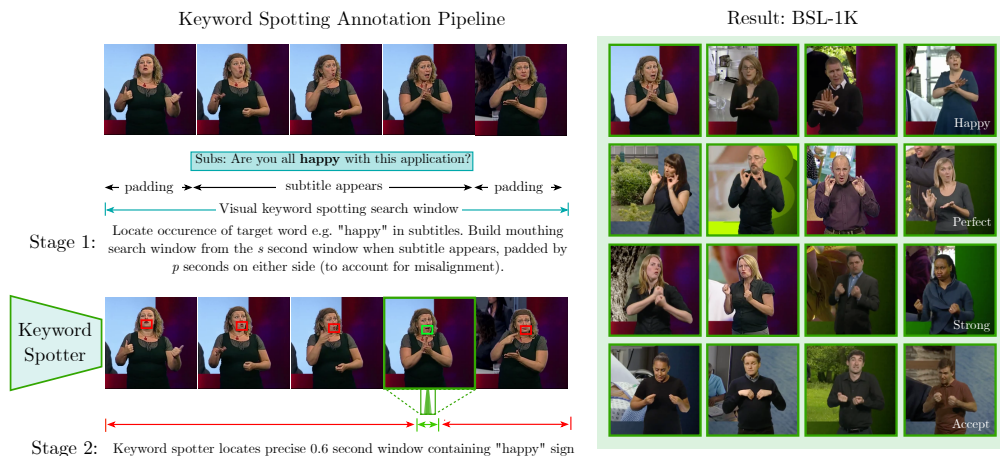


Figure 4.1: **Keyword-driven sign annotation:** (Left, the annotation pipeline): Stage 1: for a given target sign (e.g. “happy”) each occurrence of the word in the subtitles provides a candidate temporal window when the sign may occur (this is further padded by several seconds on either side to account for misalignment of subtitles and signs); Stage 2: a keyword spotter uses the mouthing of the signer to perform precise localisation of the sign within this window. (Right): Examples from the BSL-1K dataset—produced by applying keyword spotting for a vocabulary of 1K words.

4.3 Learning Sign Recognition with Automatic Labels

In this section, we describe the process used to collect BSL-1K, a large-scale dataset of BSL signs. An overview of the approach is provided in Fig. 4.1. In Sec. 4.3.1, we describe how large numbers of video clips that are likely to contain a given sign are sourced from public broadcast footage using subtitles; in Sec. 4.3.2, we show how automatic keyword spotting can be used to precisely localise specific signs to within a fraction of a second; in Sec. 4.3.3, we apply this technique to efficiently annotate a large-scale dataset with a vocabulary of 1K signs.

4.3.1 Finding probable signing windows in public broadcast footage

The source material for the dataset comprises 1,412 episodes of publicly broadcast TV programs produced by the BBC which contains 1,060 hours of continuous BSL signing. The episodes cover a wide range of topics: medical dramas, history and nature documentaries, cooking shows and programs covering gardening, business and travel. The signing represents a translation (rather than a transcription) of

the content and is produced by a total of forty professional BSL interpreters. The signer occupies a fixed region of the screen and is cropped directly from the footage. A full list of the TV shows that form BSL-1K can be found in the appendix. In addition to videos, these episodes are accompanied by subtitles (numbering approximately 9.5 million words in total). To locate temporal windows in which instances of signs are likely to occur within the source footage, we first identify a candidate list of words that: (i) are present in the subtitles; (ii) have entries in both BSL signbank² and sign BSL³, two online dictionaries of isolated signs (to ensure that we query words that have valid mappings to signs). The result is an initial vocabulary of 1,350 words, which are used as queries for the keyword spotting model to perform sign localisation—this process is described next.

4.3.2 Precise sign localisation through visual keyword spotting

By searching the content of the subtitle tracks for instances of words in the initial vocabulary, we obtain a set of candidate temporal windows in which instances of signs may occur. However, two factors render these temporal proposals extremely noisy: (1) the presence of a word in the subtitles does not guarantee its presence in the signing; (2) even for subtitled words that are signed, we find through inspection that their appearance in the subtitles can be misaligned with the sign itself by several seconds.

To address this challenge, we turn to *visual keyword spotting*. Our goal is to detect and precisely localise the presence of a sign by identifying its spoken components [Rachel Sutton-Spence and Woll 1999] within a temporal sequence of mouthing patterns. Two hypotheses underpin this approach: (a) that mouthing provides a strong localisation signal for signs as they are produced; (b) that this mouthing occurs with sufficient frequency to form a useful localisation cue. Our method is motivated by studies in the Sign Linguistics literature which find that spoken components frequently serve to identify signs—this occurs most prominently when the mouth pattern is used to distinguish between manual homonyms⁴

²<https://bslsignbank.ucl.ac.uk/>

³<https://www.signbsl.com/>

⁴These are signs that use identical hand movements (e.g. king and queen) whose meanings are distinguished by mouthings.

(see [Rachel Sutton-Spence and Woll 1999] for a detailed discussion). However, even if these hypotheses hold, the task remains extremely challenging—signers typically do not mouth continuously and the mouthings that are produced may only correspond to a portion of the word [Rachel Sutton-Spence and Woll 1999]. For this reason, existing lip reading approaches cannot be used directly (indeed, an initial exploratory experiment we conducted with the state-of-the-art lip reading model of [Afouras et al. 2019] achieved zero recall on five-hundred randomly sampled sentences of signer mouthings from the BBC source footage).

The key to the effectiveness of visual keyword spotting is that rather than solving the general problem of lip reading, it solves the much easier problem of identifying a single token from a small collection of candidates within a short temporal window. In this work, we use the subtitles to construct such windows. The pipeline for automatic sign annotations therefore consists of two stages (Fig. 4.1, left): (1) For a given target sign e.g. “happy”, determine the times of all occurrences of this sign in the subtitles accompanying the video footage. The subtitle time provides a short window during which the word was spoken, but not necessarily when its corresponding sign is produced in the translation. We therefore extend this candidate window by several seconds to increase the likelihood that the sign is present in the sequence. We include ablations to assess the influence of this padding process in Sec. 4.5 and determine empirically that padding by four seconds on each side of the subtitle represents a good choice. (2) The resulting temporal window is then provided, together with the target word, to a keyword spotting model (described in detail in Sec. 4.4.1) which estimates the probability that the sign was mouthed at each time step (we apply the keyword spotter with a stride of 0.04 seconds—this choice is motivated by the fact that the source footage has a frame rate of 25fps). When the keyword spotter asserts with high confidence that it has located a sign, we take the location of the peak posterior probability as an anchoring point for one endpoint of a 0.6 second window (this value was determined by visual inspection to be sufficient for capturing individual signs). The peak probability is then converted into a decision about whether a sign is present using a threshold parameter. To build the BSL-1K dataset, we select a value of 0.5 for this parameter after conducting experiments (reported in Tab. 4.3) to assess its influence on the downstream task of sign recognition performance.

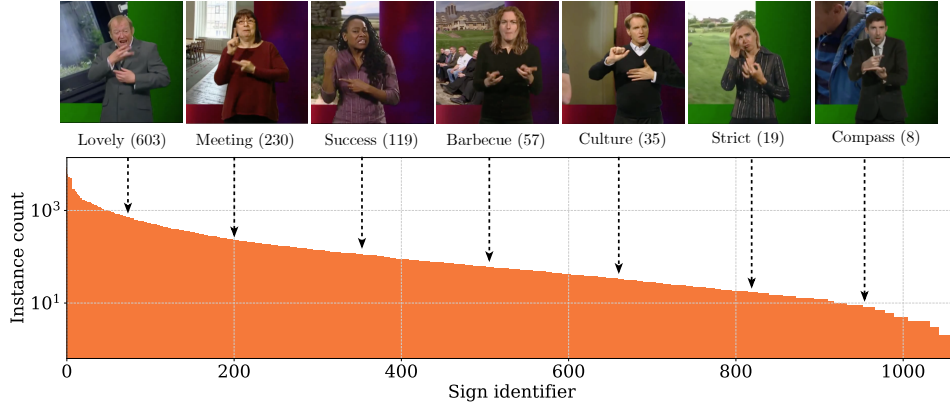


Figure 4.2: **BSL-1K sign frequencies:** Log-histogram of instance counts for the 1,064 words constituting the BSL-1K vocabulary, together with example signs. The long-tail distribution reflects the *real* setting in which some signs are more frequent than others.

4.3.3 BSL-1K dataset construction and validation

Following the sign localisation process described above, we obtain approximately 280k localised signs from a set of 2.4 million candidate subtitles. To ensure that the dataset supports study of signer-independent sign recognition, we then compute face embeddings (using an SENet-50 [J. Hu et al. 2019] architecture trained for verification on the VGGFace2 dataset [Q. Cao et al. 2018]) to group the episodes according to which of the forty signers they were translated by. We partition the data into three splits, assigning thirty-two signers for training, four signers for validation and four signers for testing. We further sought to include an equal number of hearing and non-hearing signers (the validation and test sets both contain an equal number of each, the training set is approximately balanced with 13 hearing, 17 non-hearing and 2 signers whose deafness is unknown). We then perform a further filtering step on the vocabulary to ensure that each word included in the dataset is represented with high confidence (at least one instance with confidence 0.8) in the training partition, which produces a final dataset vocabulary of 1,064 words (see Fig. 4.2 for the distribution and the appendix for the full word list).

Validating the automatic annotation pipeline. One of the key hypotheses underpinning this work is that keyword spotting is capable of correctly locating signs. We first verify this hypothesis by presenting a randomly sampled subset of the test partition to a native BSL signer, who was asked to assess whether the short temporal windows produced by the keyword spotting model with high confidence (each 0.6 seconds in duration) contained correct instances of the target

Table 4.2: **Statistics of the proposed BSL-1K dataset:** The *Test-(manually verified)* split represents a sample from the Test-(automatic) split annotations that have been verified by human annotators (see Sec. 4.3.3 for details).

Set	sign vocabulary	sign annotations	number of signers
Train	1,064	173K	32
Val	1,049	36K	4
Test-(automatic)	1,059	63K	4
Test-(manually verified)	334	2103	4

sign. A screenshot of the annotation tool developed for this task is provided in the appendix. A total of 1k signs were included in this initial assessment, of which 70% were marked as correct, 28% were marked as incorrect and 2% were marked as uncertain, validating the key idea behind the annotation pipeline. Possible reasons for incorrect marks include: BSL mouthing patterns are not always identical to spoken English and mouthings many times do not represent the full word (e.g., “fsh” for “finish”) [Rachel Sutton-Spence and Woll 1999].

Constructing a manually verified test set. To construct a high quality, human verified test set and to maximise yield from the annotators, we started from a collection of sign predictions where the keyword model was highly confident (assigning a peak probability of greater than 0.9) yielding 5,826 sign predictions. Then, in addition to the validated 980 signs (corrections were provided as labels for the signs marked as incorrect and uncertain signs were removed), we further expanded the verified test set with non-native (BSL level 2 or above) signers who annotated a further 2k signs. We found that signers with lower levels of fluency were able to confidently assert that a sign was correct for a portion of the signs (at a rate of around 60%), but also annotated a large number of signs as “unsure”, making it challenging to use these annotations as part of the validation test for the effectiveness of the pipeline. Only signs marked as correct were included into the final verified test set, which ultimately comprised 2,103 annotations covering 334 signs from the 1,064 sign vocabulary. The statistics of each partition of the dataset are provided in Tab. 4.2. All experimental test set results in this paper refer to performance on the verified test set (but we retain the full automatic test set, which we found to be useful for development).

In addition to the keyword spotting approach described above, we explore techniques for further dataset expansion based on other cues in the appendix.

4.4 Models and Implementation Details

In this section, we first describe the visual keyword spotting model used to collect signs from mouthings (Sec. 4.4.1). Next, we provide details of the model architecture for sign recognition and spotting (Sec. 4.4.2). Lastly, we describe a method for obtaining a good initialisation for the sign recognition model (Sec. 4.4.3).

4.4.1 Visual keyword spotting model

We use the improved visual-only keyword spotting model of Stafylakis et al. [Stafylakis and Tzimiropoulos 2018] from [Momeni et al. 2020a] (referred to in their paper as “P2G [Stafylakis and Tzimiropoulos 2018] baseline”), provided by the authors. The model of [Stafylakis and Tzimiropoulos 2018] combines visual features with a fixed-length keyword embedding to determine whether a user-defined keyword is present in an input video clip. The performance of [Stafylakis and Tzimiropoulos 2018] is improved in [Momeni et al. 2020a] by switching the keyword encoder-decoder from grapheme-to-phoneme (G2P) to phoneme-to-grapheme (P2G).

In more detail, the model consists of four stages: (i) visual features are first extracted from the sequence of face-cropped image frames from a clip (this is performed using a 512×512 SSD architecture [W. Liu et al. 2016] trained for face detection on WIDER faces [S. Yang et al. 2016]), (ii) a fixed-length keyword representation is built using a P2G encoder-decoder, (iii) the visual and keyword embeddings are concatenated and passed through BiLSTMs, (iv) finally, a sigmoid activation is applied on the output to approximate the posterior probability that the keyword occurs in the video clip for each input frame. If the maximum posterior over all frames is greater than a threshold, the clip is predicted to contain the keyword. The predicted location of the keyword is the position of the maximum posterior. Finally, non-maximum suppression is run with a temporal window of 0.6 seconds over the untrimmed source videos to remove duplicates.

4.4.2 Sign recognition model

We employ a spatio-temporal convolutional neural network architecture that takes a multiple-frame video as input, and outputs class probabilities over sign categories. Specifically, we follow the I3D architecture [João Carreira and Zisserman

2017] due to its success on action recognition benchmarks, as well as its recently observed success on sign recognition datasets [Joze and Koller 2019; D. Li et al. 2019]. To retain computational efficiency, we only use an RGB stream. The model is trained on 16-frame consecutive frames (i.e., 0.64 sec for 25fps), as [Buehler et al. 2009; Pfister et al. 2013; Viitaniemi et al. 2014] observed that co-articulated signs last roughly for 13 frames. We resize our videos to have a spatial resolution of 224×224 . For training, we randomly subsample a fixed-size, temporally contiguous input from the spatio-temporal volume to have $16 \times 224 \times 224$ resolution in terms of number of frames, width, and height, respectively. We minimise the cross-entropy loss using SGD with momentum (0.9) with mini-batches of size 4, and an initial learning rate of 10^{-2} with a fixed schedule. The learning rate is decreased twice with a factor of 10^{-1} at epochs 20 and 40. We train for 50 epochs. Colour, scale, and horizontal flip augmentations are applied on the input video. When pretraining is used (e.g. on Kinetics-400 [João Carreira and Zisserman 2017] or on other data where specified), we replace the last linear layer with the dimensionality of our classes, and fine-tune all network parameters (we observed that freezing part of the model is suboptimal). Finally, we apply dropout on the classification layer with a probability of 0.5.

At test time, we perform centre-cropping and apply a sliding window with a stride of 8 frames before averaging the classification scores to obtain a video-level prediction.

4.4.3 Video pose distillation

Given the significant focus on pose estimation in the sign language recognition literature, we investigate how explicit pose modelling can be used to improve the I3D model. To this end, we define a *pose distillation* network that takes in a sequence of 16 consecutive frames, but rather than predicting sign categories, the 1024-dimensional (following average pooling) embedding produced by the network is used to regress the poses of individuals appearing in each of the frames of its input. In more detail, we assume a single individual per-frame (as is the case in cropped sign translation footage) and task the network with predicting 130 human pose keypoints (18 body, 21 per hand, and 70 facial) produced by an OpenPose [Z. Cao et al. 2018] model (trained on COCO [T.-Y. Lin et al. 2014])

that is evaluated per-frame. The key idea is that, in order to effectively predict pose across multiple frames from a single video embedding, the model is encouraged to encode information not only about pose, but also descriptions of relevant dynamic gestures. The model is trained on a portion of the BSL-1K training set (due to space constraints, further details of the model architecture and training procedure are provided in the appendix).

4.5 Experiments

We first provide several ablations on our sign recognition model to answer questions such as which cues are important, and how to best use human pose. Then, we present baseline results for sign recognition and sign spotting, with our best model. Finally, we compare to the state of the art on ASL benchmarks to illustrate the benefits of pretraining on our data.

4.5.1 Ablations for the sign recognition model

In this section, we evaluate our sign language recognition approach and investigate (i) the effect of mouthing score threshold, (ii) the comparison to pose-based approaches, (iii) the contribution of multi-modal cues, and (iv) the video pose distillation. Additional ablations about the influence of the search window size for the keyword spotting and the temporal extent of the automatic annotations can be found in the appendix.

Evaluation metrics. Following [Joze and Koller 2019; D. Li et al. 2019], we report both top-1 and top-5 classification accuracy, mainly due to ambiguities in signs which can be resolved in context. Furthermore, we adopt both per-instance and per-class accuracy metrics. Per-instance accuracy is computed over all test instances. Per-class accuracy refers to the average over the sign categories present in the test set. We use this metric due to the unbalanced nature of the datasets.

The effect of the mouthing score threshold. The keyword spotting method, being a binary classification model, provides a confidence score, which we threshold to obtain our automatically annotated video clips. Reducing this threshold yields an increased number of sign instances at the cost of a potentially noisier set of annotations. We denote the training set defined by a mouthing threshold 0.8 as

Table 4.3: **Trade-off between training noise vs. size:** Training (with Kinetics initialisation) on the full training set BSL-1K_{*m.5*} versus the subset BSL-1K_{*m.8*}, which correspond to a mouthing score threshold of 0.5 and 0.8, respectively. Even when noisy, with the 0.5 threshold, mouthings provide automatic annotations that allow supervised training at scale, resulting in 70.61% accuracy on the manually validated test set.

Training data	#videos	per-instance		per-class	
		top-1	top-5	top-1	top-5
BSL-1K _{<i>m.8</i>} (mouthing \geq 0.8)	39K	69.00	83.79	45.86	64.42
BSL-1K _{<i>m.5</i>} (mouthing \geq 0.5)	173K	70.61	85.26	47.47	68.13

BSL-1K_{*m.8*}. In Tab. 4.3, we show the effect of changing this hyper-parameter between a low- and high-confidence model with 0.5 and 0.8 mouthing thresholds, respectively. The larger set of training samples obtained with a threshold of 0.5 provide the best performance. For the remaining ablations, we use the smaller BSL-1K_{*m.8*} training set for faster iterations, and return to the larger BSL-1K_{*m.5*} set for training the final model.

Pose-based model versus I3D. We next verify that I3D is a suitable model for sign language recognition by comparing it to a pose-based approach. We implement Pose→Sign, which follows a 2D ResNet architecture [K. He et al. 2016] that operates on $3 \times 16 \times P$ dimensional dynamic pose images, where P is the number of keypoints. In our experiments, we use OpenPose [Z. Cao et al. 2018] (pretrained on COCO [T.-Y. Lin et al. 2014]) to extract 18 body, 21 per hand, and 70 facial keypoints. We use 16-frame inputs to make it comparable to the I3D counterpart. We concatenate the estimated normalised xy coordinates of a keypoint with its confidence score to obtain the 3 channels. In Tab. 4.4, we see that I3D significantly outperforms the explicit 2D pose-based method (65.57% vs 49.66% per-instance accuracy). This conclusion is in accordance with the recent findings of [Joze and Koller 2019; D. Li et al. 2019].

Contribution of individual cues. We carry out two set of experiments to determine how much our sign recognition model relies on signals from the mouth and face region versus the manual features from the body and hands: (i) using Pose→Sign, which takes as input the 2D keypoint locations over several frames, (ii) using I3D, which takes as input raw video frames. For the pose-based model, we train with only 70 facial keypoints, 60 body&hand keypoints, or with the combination. For I3D, we use the pose estimations to mask the pixels outside

Table 4.4: **Contribution of individual cues:** We compare I3D (pretrained on Kinetics) with a keypoint-based baseline both trained and evaluated on a subset of BSL-1K_{m.8}, where we have the pose estimates. We also quantify the contribution of the body&hands and the face regions. We see that significant information can be attributed to both types of cues, and the combination performs the best.

	body&hands	face	per-instance		per-class	
			top-1	top-5	top-1	top-5
Pose→Sign (70 points)	✗	✓	24.41	47.59	9.74	25.99
Pose→Sign (60 points)	✓	✗	40.47	59.45	20.24	39.27
Pose→Sign (130 points)	✓	✓	49.66	68.02	29.91	49.21
I3D (face-crop)	✗	✓	42.23	69.70	21.66	50.51
I3D (mouth-masked)	✓	✗	46.75	66.34	25.85	48.02
I3D (full-frame)	✓	✓	65.57	81.33	44.90	64.91

Table 4.5: **Effect of pretraining** the I3D model on various tasks before fine-tuning for sign recognition on BSL-1K_{m.8}. Our dynamic pose features learned on 16-frame videos provide body-motion-aware cues and outperform other pretraining strategies.

Task	Pretraining Data	per-instance		per-class	
		top-1	top-5	top-1	top-5
Random init.	-	39.80	61.01	15.76	29.87
Gesture recognition	Jester [Materzynska et al. 2019]	46.93	65.95	19.59	36.44
Sign recognition	WLASL [D. Li et al. 2019]	69.90	83.45	44.97	62.73
Action recognition	Kinetics [João Carreira and Zisserman 2017]	69.00	83.79	45.86	64.42
Video pose distillation	Signers	70.38	84.50	46.24	65.31

of the face bounding box, to mask the mouth region, or use all the pixels from the videos. The results are summarised in Tab. 4.4. We observe that using only the face provides a strong baseline, suggesting that mouthing is a strong cue for recognising signs, e.g., 42.23% for I3D. However, using all the cues, including body and hands (65.57%), significantly outperforms using individual modalities.

Pretraining for sign recognition. Next we investigate different forms of pretraining for the I3D model. In Tab. 4.5, we compare the performance of a model trained with random initialisation (39.80%), fine-tuning from gesture recognition (46.93%), sign recognition (69.90%), and action recognition (69.00%). Video pose distillation provides a small boost over the other pretraining strategies (70.38%), suggesting that it is an effective way to force the I3D model to pay attention to the dynamics of the human keypoints, which is relevant for sign recognition.

4.5.2 Benchmarking sign recognition and sign spotting

Next, we combine the parameter choices suggested by each of our ablations to establish baseline performances on the BSL-1K dataset for two tasks: (i) sign recognition, (ii) sign spotting. Specifically, the model comprises an I3D archi-



Figure 4.3: **Qualitative analysis:** We present results of our sign recognition model on BSL-1K for success (top) and failure (bottom) cases, together with their confidence scores in parentheses. To the right of each example, we show a random training sample for the predicted sign (in small). We observe that failure modes are commonly due to high visual similarity in the gesture (bottom-left) and mouthing (bottom-right).

Table 4.6: **Benchmarking:** We benchmark our best sign recognition model (trained on BSL-1K_{m.5}, initialised with pose distillation, with 4-frame temporal offsets) for sign recognition and sign spotting task to establish strong baselines on BSL-1K.

	per-instance		per-class		mAP (334 sign classes)	
	top-1	top-5	top-1	top-5		
Sign Recognition	75.51	88.83	52.76	72.14	Sign Spotting	0.159

texture trained on BSL-1K_{m.5} with pose-distillation as initialisation and random temporal offsets of up to 4 frames around the sign during training (the ablation studies for this temporal augmentation parameter are included in the appendix). The sign recognition evaluation protocol follows the experiments conducted in the ablations, the sign spotting protocol is described next.

Sign spotting. Differently from sign recognition, in which the objective is to classify a pre-defined temporal segment into a category from a given vocabulary, *sign spotting* aims to locate all instances of a particular sign within long sequences of untrimmed footage, enabling applications such as content-based search and efficient indexing of signing videos for which subtitles are not available. The evaluation protocol for assessing sign spotting on BSL-1K is defined as follows: for each sign category present amongst the human-verified test set annotations (334 in total), windows of 0.6-second centred on each verified instance are marked as positive and all other times within the subset of episodes that contain at least one instance of the sign are marked as negative. To avoid false penalties at signs that

were not discovered by the automatic annotation process, we exclude windows of 8 seconds of footage centred at each location in the original footage at which the target keyword appears in the subtitles, but was not detected by the visual keyword spotting pipeline. In aggregate this corresponds to locating approximately one positive instance of a sign in every 1.5 hours of continuous signing negatives. A sign is considered to have been correctly spotted if its temporal overlap with the model prediction exceeds an IoU (intersection-over-union) of 0.5, and we report mean Average Precision (mAP) over the 334 sign classes as the metric for performance.

We report the performance of our strongest model for both the sign recognition and sign spotting benchmarks in Tab. 4.6. In Fig. 4.3, we provide some qualitative results from our sign recognition method and observe some modes of failure which are driven by strong visual similarity in sign production.

4.5.3 Comparison with the state of the art on ASL benchmarks

BSL-1K, being significantly larger than the recent WLASL [D. Li et al. 2019] and MSASL [Joze and Koller 2019] benchmarks, can be used for pretraining I3D models to provide strong initialisation for other datasets. Here, we transfer the features from BSL to ASL, which are two distinct sign languages.

As models from [Joze and Koller 2019] were not publicly available, we first reproduce the I3D Kinetics pretraining baseline with our implementation to achieve fair comparisons. We use 64-frame inputs as isolated signs in these datasets are significantly slower than co-articulated signs. We then train I3D from BSL-1K pretrained features. Tab. 4.7 compares pretraining on Kinetics versus our BSL-1K data. BSL-1K provides a significant boost in the performance, outperforming the state-of-the-art results (46.82% and 64.71% top-1 accuracy). Find additional details, as well as similar experiments on co-articulated datasets in the appendix.

Table 4.7: **Transfer to ASL:** Performance on American Sign Language (ASL) datasets with and without pretraining on our data. I3D results are reported from the original papers for MSASL [Joze and Koller 2019] and WLASL [D. Li et al. 2019]. I3D† denotes our implementation and training, adopting the hyperparameters from [Joze and Koller 2019]. We show that our features provide good initialisation, even if it is trained on BSL.

	pretraining	WLASL [D. Li et al. 2019]				MSASL [Joze and Koller 2019]			
		per-instance		per-class		per-instance		per-class	
		top-1	top-5	top-1	top-5	top-1	top-5	top-1	top-5
I3D [Joze and Koller 2019]	Kinetics	-	-	-	-	-	-	57.69	81.08
I3D [D. Li et al. 2019]	Kinetics	32.48	57.31	-	-	-	-	-	-
I3D†	Kinetics	40.85	74.10	39.06	73.33	60.45	82.05	57.17	80.02
I3D	BSL-1K	46.82	79.36	44.72	78.47	64.71	85.59	61.55	84.43

4.6 Conclusion

We have demonstrated the advantages of using visual keyword spotting to automatically annotate continuous sign language videos with weakly-aligned subtitles. We have presented BSL-1K, a large-scale dataset of co-articulated signs that, coupled with a 3D CNN training, allows high-performance recognition of signs from a large-vocabulary. Our model has further shown beneficial as initialisation for ASL benchmarks. Finally, we have provided ablations and baselines for sign recognition and sign spotting tasks. A potential future direction is leveraging our automatic annotations and recognition model for sign language translation.

Acknowledgements. This work was supported by EPSRC grant ExTol. We also thank T. Stafylakis, A. Brown, A. Dutta, L. Dunbar, A. Thandavan, C. Camgoz, O. Koller, H. V. Joze, O. Kopuklu for their help.

Part II

Approaches for Sign Spotting

Chapter 5

Watch, read and lookup: learning to spot signs from multiple supervisors

The paper has been accepted for publication at the Asian Conference on Computer Vision (ACCV), 2020. It was awarded Best Application Paper.

Watch, read and lookup: learning to spot signs from multiple supervisors

Liliane Momeni* Gül Varol* Samuel Albanie*

Triantafyllos Afouras Andrew Zisserman

Visual Geometry Group, University of Oxford, UK

Abstract

The focus of this work is *sign spotting*—given a video of an isolated sign, our task is to identify *whether* and *where* it has been signed in a continuous, co-articulated sign language video. To achieve this sign spotting task, we train a model using multiple types of available supervision by: (1) *watching* existing sparsely labelled footage; (2) *reading* associated subtitles (readily available translations of the signed content) which provide additional *weak-supervision*; (3) *looking up* words (for which no co-articulated labelled examples are available) in visual sign language dictionaries to enable novel sign spotting. These three tasks are integrated into a unified learning framework using the principles of Noise Contrastive Estimation and Multiple Instance Learning. We validate the effectiveness of our approach on low-shot sign spotting benchmarks. In addition, we contribute a machine-readable British Sign Language (BSL) dictionary dataset of isolated signs, BSLDICT, to facilitate study of this task. The dataset, models and code are available at our project page¹.

5.1 Introduction

The objective of this work is to develop a *sign spotting* model that can identify and localise instances of signs within sequences of continuous sign language.

*Equal contribution.

¹<https://www.robots.ox.ac.uk/~vgg/research/bsldict/>



Figure 5.1: We consider the task of *sign spotting* in co-articulated, continuous signing. Given a query dictionary video of an isolated sign (e.g. “apple”), we aim to identify *whether* and *where* it appears in videos of continuous signing. The wide domain gap between dictionary examples of *isolated* signs and target sequences of *continuous* signing makes the task extremely challenging.

Sign languages represent the natural means of communication for deaf communities [Rachel Sutton-Spence and Woll 1999] and sign spotting has a broad range of practical applications. Examples include: indexing videos of signing content by keyword to enable content-based search; gathering diverse dictionaries of sign exemplars from unlabelled footage for linguistic study; automatic feedback for language students via an auto-correct tool (e.g. ‘did you mean this sign?’); making voice activated wake word devices accessible to deaf communities; and building sign language datasets by automatically labelling examples of signs.

The recent marriage of large-scale, labelled datasets with deep neural networks has produced considerable progress in audio [Coucke et al. 2019; Véniat et al. 2019] and visual [Momeni et al. 2020a; Stafylakis and Tzimiropoulos 2018] keyword spotting in *spoken languages*. However, a direct replication of these keyword spotting successes in sign language requires a commensurate quantity of labelled data (note that modern audiovisual spoken keyword spotting datasets contain millions of densely labelled examples [Chung et al. 2017; Afouras et al. 2018b]). Large-scale corpora of continuous, co-articulated² signing from TV broadcast data have recently been built [Albanie et al. 2020], but the labels accompanying this data are: (1) sparse, and (2) cover a limited vocabulary.

It might be thought that a sign language dictionary would offer a relatively straightforward solution to the sign spotting task, particularly to the problem of covering only a limited vocabulary in existing large-scale corpora. But, unfortunately, this is not the case due to the severe *domain differences* between dictionaries and continuous signing in the wild. The challenges are that sign language dictionaries typi-

²*Co-articulation* refers to changes in the appearance of the current sign due to neighbouring signs.

cally: (i) consist of *isolated signs* which differ in appearance from the *co-articulated* sequences of continuous signs (for which we ultimately wish to perform spotting); and (ii) differ in speed (are performed more slowly) relative to co-articulated signing. Furthermore, (iii) dictionaries only possess a few examples of each sign (so learning must be *low shot*); and as one more challenge, (iv) there can be multiple signs corresponding to a single keyword, for example due to regional variations of the sign language [Schembri et al. 2017]. We show through experiments in Sec. 5.4, that directly training a sign spotter for continuous signing on dictionary examples, obtained from an internet-sourced sign language dictionary, does indeed perform poorly.

To address these challenges, we propose a unified framework in which sign spotting embeddings are learned from the dictionary (to provide broad coverage of the lexicon) in combination with two additional sources of supervision. In aggregate, these multiple types of supervision include: (1) *watching* sign language and learning from existing sparse annotations; (2) exploiting weak-supervision by *reading* the subtitles that accompany the footage and extracting candidates for signs that we expect to be present; (3) *looking up* words (for which we do not have labelled examples) in a sign language dictionary. The recent development of large-scale, subtitled corpora of continuous signing providing sparse annotations [Albanie et al. 2020] allows us to study this problem setting directly. We formulate our approach as a Multiple Instance Learning problem in which positive samples may arise from any of the three sources and employ Noise Contrastive Estimation [Gutmann and Hyvärinen 2010] to learn a domain-invariant (valid across both isolated and co-articulated signing) representation of signing content.

We make the following six contributions: (1) We provide a machine readable British Sign Language (BSL) dictionary dataset of isolated signs, BSLDICT, to facilitate study of the sign spotting task; (2) We propose a unified Multiple Instance Learning framework for learning sign embeddings suitable for spotting from three supervisory sources; (3) We validate the effectiveness of our approach on a co-articulated sign spotting benchmark for which only a small number (low-shot) of isolated signs are provided as labelled training examples, and (4) achieve state-of-the-art performance on the BSL-1K sign spotting benchmark [Albanie et al. 2020] (closed vocabulary). We show qualitatively that the learned embeddings can be

used to (5) automatically mine new signing examples, and (6) discover faux amis (false friends) between sign languages.

5.2 Related Work

Our work relates to several themes in the literature: *sign language recognition* (and more specifically *sign spotting*), *sign language datasets*, *multiple instance learning* and *low-shot action localization*. We discuss each of these themes next.

Sign language recognition. The study of automatic sign recognition has a rich history in the computer vision community stretching back over 30 years, with early methods developing carefully engineered features to model trajectories and shape [Kadir et al. 2004; Tamura and Kawasaki 1988; Starner 1995; Fillbrandt et al. 2003]. A series of techniques then emerged which made effective use of hand and body pose cues through robust keypoint estimation encodings [Buehler et al. 2009; Cooper et al. 2011; Ong et al. 2012; Pfister et al. 2014]. Sign language recognition also has been considered in the context of sequence prediction, with HMMs [Agris et al. 2008; Forster et al. 2013; Starner 1995; Kadir et al. 2004], LSTMs [N. C. Camgoz et al. 2017; J. Huang et al. 2018b; Ye et al. 2018; H. Zhou et al. 2020b], and Transformers [N. C. Camgoz et al. 2020b] proving to be effective mechanisms for this task. Recently, convolutional neural networks have emerged as the dominant approach for appearance modelling [N. C. Camgoz et al. 2017], and in particular, action recognition models using spatio-temporal convolutions [João Carreira and Zisserman 2017] have proven very well-suited for video-based sign recognition [Joze and Koller 2019; D. Li et al. 2019; Albanie et al. 2020]. We adopt the I3D architecture [João Carreira and Zisserman 2017] as a foundational building block in our studies.

Sign language spotting. The sign language spotting problem—in which the objective is to find performances of a sign (or sign sequence) in a longer sequence of signing—has been studied with Dynamic Time Warping and skin colour histograms [Viitaniemi et al. 2014] and with Hierarchical Sequential Patterns [Eng-Jon Ong et al. 2014]. Different from our work which learns representations from multiple weak supervisory cues, these approaches consider a fully-supervised setting with a single source of supervision and use hand-crafted features to represent

signs [Ali Farhadi et al. 2007]. Our proposed use of a dictionary is also closely tied to *one-shot/few-shot learning*, in which the learner is assumed to have access to only a handful of annotated examples of the target category. One-shot dictionary learning was studied by [Pfister et al. 2014] – different to their approach, we explicitly account for dialect variations in the dictionary (and validate the improvements brought by doing so in Sec. 5.4). Textual descriptions from a dictionary of 250 signs were used to study zero-shot learning by [Bilge et al. 2019] – we instead consider the practical setting in which a handful of video examples are available per-sign (and make this dictionary available). The use of dictionaries to locate signs in subtitled video also shares commonalities with *domain adaptation*, since our method must bridge differences between the dictionary and the target continuous signing distribution. A vast number of techniques have been proposed to tackle distribution shift, including several adversarial feature alignment methods that are specialised for the few-shot setting [Motiian et al. 2017; Junyi Zhang et al. 2019]. In our work, we explore the domain-specific batch normalization (DSBN) method of [Chang et al. 2019], finding ultimately that simple batch normalization parameter re-initialization is most effective when jointly training on two domains after pre-training on the bigger domain. The concurrent work of [D. Li et al. 2020b] also seeks to align representation of isolated and continuous signs. However, our work differs from theirs in several key aspects: (1) rather than assuming access to a large-scale labelled dataset of isolated signs, we consider the setting in which only a handful of dictionary examples may be used to represent a word; (2) we develop a generalised Multiple Instance Learning framework which allows the learning of representations from weakly aligned subtitles whilst exploiting sparse labels and dictionaries (this integrates cues beyond the learning formulation in [D. Li et al. 2020b]); (3) we seek to label and improve performance on co-articulated signing (rather than improving recognition performance on isolated signing). Also related to our work, [Pfister et al. 2014] uses a reservoir of weakly labelled sign footage to improve the performance of a sign classifier learned from a small number of examples. Different to [Pfister et al. 2014], we propose a multi-instance learning formulation that explicitly accounts for signing variations that are present in the dictionary.

Sign language datasets. A number of sign language datasets have been proposed

for studying Finnish [Viitaniemi et al. 2014], German [Koller et al. 2015b; von Agris et al. 2008], American [Athitsos et al. 2008; Joze and Koller 2019; D. Li et al. 2019; Wilbur and Kak 2006] and Chinese [Chai et al. 2014; J. Huang et al. 2018b] sign recognition. For British Sign Language (BSL), [Schembri et al. 2013] gathered a corpus labelled with sparse, but fine-grained linguistic annotations, and more recently [Albanie et al. 2020] collected BSL-1K, a large-scale dataset of BSL signs that were obtained using a mouthing-based keyword spotting model. In this work, we contribute BSLDICT, a dictionary-style dataset that is complementary to the datasets of [Schembri et al. 2013; Albanie et al. 2020] – it contains only a handful of instances of each sign, but achieves a comprehensive coverage of the BSL lexicon with a 9K vocabulary (vs a 1K vocabulary in [Albanie et al. 2020]). As we show in the sequel, this dataset enables a number of sign spotting applications.

Multiple instance learning. Motivated by the readily available sign language footage that is accompanied by subtitles, a number of methods have been proposed for learning the association between signs and words that occur in the subtitle text [Buehler et al. 2009; Cooper and Bowden 2009; Pfister et al. 2014; Chung and Zisserman 2016c]. In this work, we adopt the framework of Multiple Instance Learning (MIL) [Dietterich et al. 1997] to tackle this problem, previously explored by [Buehler et al. 2009; Pfister et al. 2013]. Our work differs from these works through the incorporation of a dictionary, and a principled mechanism for explicitly handling sign variants, to guide the learning process. Furthermore, we generalise the MIL framework so that it can learn to further exploit sparse labels. We also conduct experiments at significantly greater scale to make use of the full potential of MIL, considering more than two orders of magnitude more weakly supervised data than [Buehler et al. 2009; Pfister et al. 2013].

Low-shot action localization. This theme investigates semantic video localization: given one or more query videos the objective is to localize the segment in an untrimmed video that corresponds semantically to the query video [Feng et al. 2018; H. Yang et al. 2018; K. Cao et al. 2020]. Semantic matching is too general for the sign-spotting considered in this paper. However, we build on the temporal ordering ideas explored in this theme.

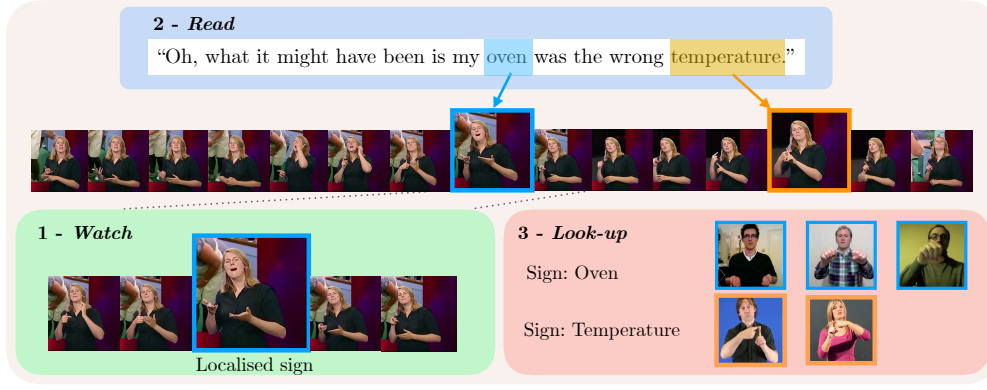


Figure 5.2: The proposed *Watch, Read and Lookup* framework trains sign spotting embeddings with three cues: (1) *watching* videos and learning from sparse annotation in the form of localised signs (lower-left); (2) *reading* subtitles to find candidate signs that may appear in the source footage (top); (3) *looking up* corresponding visual examples in a sign language dictionary and aligning the representation against the embedded source segment (lower-right).

5.3 Learning Sign Spotting Embeddings from Multiple Supervisors

In this section, we describe the task of *sign spotting* and the three forms of supervision we assume access to. Let $\mathcal{X}_{\mathcal{L}}$ denote the space of RGB video segments containing a frontal-facing individual communicating in sign language \mathcal{L} and denote by $\mathcal{X}_{\mathcal{L}}^{\text{single}}$ its restriction to the set of segments containing a single sign. Further, let \mathcal{T} denote the space of subtitle sentences and $\mathcal{V}_{\mathcal{L}} = \{1, \dots, V\}$ denote the *vocabulary*—an index set corresponding to an enumeration of written words that are equivalent to signs that can be performed in \mathcal{L} ³.

Our objective, illustrated in Fig. 5.1, is to discover all occurrences of a given keyword in a collection of continuous signing sequences. To do so, we assume access to: (i) a subtitled collection of videos containing continuous signing, $\mathcal{S} = \{(x_i, s_i) : i \in \{1, \dots, I\}, x_i \in \mathcal{X}_{\mathcal{L}}, s_i \in \mathcal{T}\}$; (ii) a sparse collection of temporal sub-segments of these videos that have been annotated with their corresponding word, $\mathcal{M} = \{(x_k, v_k) : k \in \{1, \dots, K\}, v_k \in \mathcal{V}_{\mathcal{L}}, x_k \in \mathcal{X}_{\mathcal{L}}^{\text{single}}, \exists (x_i, s_i) \in \mathcal{S} \text{ s.t. } x_k \subseteq x_i\}$; (iii) a curated *dictionary* of signing instances $\mathcal{D} = \{(x_j, v_j) : j \in \{1, \dots, J\}, x_j \in \mathcal{X}_{\mathcal{L}}^{\text{single}}, v_j \in \mathcal{V}_{\mathcal{L}}\}$. To address the sign spotting task, we propose to learn a *data representation* $f : \mathcal{X}_{\mathcal{L}} \rightarrow \mathbb{R}^d$ that maps video segments to vectors such that they

³Sign language dictionaries provide a word-level or phrase-level correspondence (between sign language and spoken language) for many signs but no universally accepted *glossing* scheme exists for transcribing languages such as BSL [Rachel Sutton-Spence and Woll 1999].

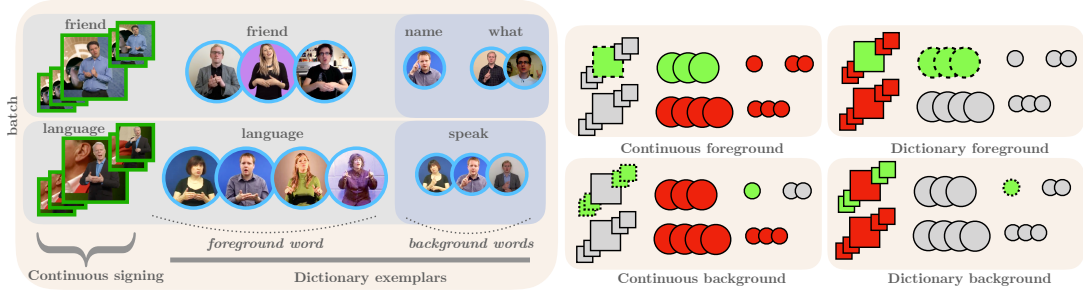


Figure 5.3: **Batch sampling and positive/negative pairs:** We illustrate the formation of a batch when jointly training on continuous signing video (squares) and dictionaries of isolated signing (circles). **Left:** For each continuous video, we sample the dictionaries corresponding to the labelled word (foreground), as well as to the rest of the subtitles (background). **Right:** We construct positive/negative pairs by anchoring at 4 different portions of a batch item: continuous foreground/background and dictionary foreground/background. Positives and negatives (defined across continuous and dictionary domains) are green and red, respectively; anchors have a dashed border (see supplementary).

are *discriminative* for sign spotting and *invariant* to other factors of variation. Formally, for any labelled pair of video segments $(x, v), (x', v')$ with $x, x' \in \mathcal{X}_{\mathcal{S}}$ and $v, v' \in \mathcal{V}_{\mathcal{S}}$, we seek a data representation, f , that satisfies the constraint $\delta_{f(x)f(x')} = \delta_{vv'}$, where δ represents the Kronecker delta.

5.3.1 Integrating Cues through Multiple Instance Learning

To learn f , we must address several challenges. First, as noted in Sec. 5.1, there may be a considerable distribution shift between the dictionary videos of isolated signs in \mathcal{D} and the co-articulated signing videos in \mathcal{S} . Second, sign languages often contain multiple sign variants for a single written word (resulting from regional dialects and synonyms). Third, since the subtitles in \mathcal{S} are only weakly aligned with the sign sequence, we must learn to associate signs and words from a noisy signal that lacks temporal localisation. Fourth, the localised annotations provided by \mathcal{M} are sparse, and therefore we must make good use of the remaining segments of subtitled videos in \mathcal{S} if we are to learn an effective representation.

Given full supervision, we could simply adopt a pairwise metric learning approach to align segments from the videos in \mathcal{S} with dictionary videos from \mathcal{D} by requiring that f maps a pair of isolated and co-articulated signing segments to the same point in the embedding space if they correspond to the same sign (*positive* pairs)

and apart if they do not (*negative* pairs). As noted above, in practice we do not have access to positive pairs because: (1) for any annotated segment $(x_k, v_k) \in \mathcal{M}$, we have a set of potential sign variations represented in the dictionary (annotated with the common label v_k), rather than a single unique sign; (2) since \mathcal{S} provides only weak supervision, even when a word is mentioned in the subtitles we do not know where it appears in the continuous signing sequence (if it appears at all). These ambiguities motivate a Multiple Instance Learning [Dietterich et al. 1997] (MIL) objective. Rather than forming positive and negative pairs, we instead form positive *bags* of pairs, $\mathcal{P}^{\text{bags}}$, in which we expect at least one pairing between a segment from a video in \mathcal{S} and a dictionary video from \mathcal{D} to contain the same sign, and negative bags of pairs, $\mathcal{N}^{\text{bags}}$, in which we expect no (video segment, dictionary video) pair to contain the same sign. To incorporate the available sources of supervision into this formulation, we consider two categories of positive and negative bag formations, described next (due to space constraints, a formal mathematical description of the positive and negative bags described below is deferred to the supp. mat.).

Watch and Lookup: using sparse annotations and dictionaries. Here, we describe a baseline where we assume no subtitles are available. To learn f from \mathcal{M} and \mathcal{D} , we define each positive bag as the set of possible pairs between a *labelled (foreground)* temporal segment of a continuous video from \mathcal{M} and the examples of the corresponding sign in the dictionary. The key assumption here is that each labelled sign segment from \mathcal{M} matches *at least one* sign variation in the dictionary. Negative bags are constructed by (i) anchoring on a continuous foreground segment and selecting dictionary examples corresponding to different words from other batch items; (ii) anchoring on a dictionary foreground set and selecting continuous foreground segments from other batch items. To maximize the number of negatives within one minibatch, we sample a different word per batch item.

Watch, Read and Lookup: using sparse annotations, subtitles and dictionaries. Using just the labelled sign segments from \mathcal{M} to construct bags has a significant limitation: f is not encouraged to represent signs beyond the initial vocabulary represented in \mathcal{M} . We therefore look at the subtitles (which contain words beyond \mathcal{M}) to construct additional bags. We determine more positive bags

between the set of *unlabelled (background)* segments in the continuous footage and the set of dictionaries corresponding to the background words in the subtitle (green regions in Fig. 5.3, right-bottom). Negatives (red regions in Fig. 5.3) are formed as the complements to these sets by (i) pairing continuous background segments with dictionary samples that can be excluded as matches (through subtitles) and (ii) pairing background dictionary entries with the foreground continuous segment. In both cases, we also define negatives from other batch items by selecting pairs where the word(s) have no overlap, e.g., in Fig. 5.3, the dictionary examples for the background word ‘speak’ from the second batch item are negatives for the background continuous segments from the first batch item, corresponding to the unlabelled words ‘name’ and ‘what’ in the subtitle.

To assess the similarity of two embedded video segments, we employ a similarity function $\psi : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}$ whose value increases as its arguments become more similar (in this work, we use cosine similarity). For notational convenience below, we write ψ_{ij} as shorthand for $\psi(f(x_i), f(x_j))$. To learn f , we consider a generalization of the InfoNCE loss [Oord et al. 2018; P. Wu et al. 2016] (a non-parametric softmax loss formulation of Noise Contrastive Estimation [Gutmann and Hyvärinen 2010]) recently proposed by [Miech et al. 2020]:

$$\mathcal{L}_{\text{MIL-NCE}} = -\mathbb{E}_i \left[\log \frac{\sum_{(j,k) \in \mathcal{P}(i)} \exp(\psi_{jk}/\tau)}{\sum_{(j,k) \in \mathcal{P}(i)} \exp(\psi_{jk}/\tau) + \sum_{(l,m) \in \mathcal{N}(i)} \exp(\psi_{lm}/\tau)} \right], \quad (5.1)$$

where $\mathcal{P}(i) \in \mathcal{P}^{\text{bags}}$, $\mathcal{N}(i) \in \mathcal{N}^{\text{bags}}$, τ , often referred to as the *temperature*, is set as a hyperparameter (we explore the effect of its value in Sec. 5.4).

5.3.2 Implementation details

In this section, we provide details for the learning framework covering the embedding architecture, sampling protocol and optimization procedure.

Embedding architecture. The architecture comprises an I3D spatio-temporal trunk network [João Carreira and Zisserman 2017] to which we attach an MLP consisting of three linear layers separated by leaky ReLU activations (with negative slope 0.2) and a skip connection. The trunk network takes as input 16 frames

from a 224×224 resolution video clip and produces 1024-dimensional embeddings which are then projected to 256-dimensional sign spotting embeddings by the MLP. More details about the embedding architecture can be found in the supplementary material.

Joint pretraining. The I3D trunk parameters are initialised by pretraining for sign classification jointly over the sparse annotations \mathcal{M} of a continuous signing dataset (BSL-1K [Albanie et al. 2020]) and examples from a sign dictionary dataset (BSLDICT) which fall within their common vocabulary. Since we find that dictionary videos of isolated signs tend to be performed more slowly, we uniformly sample 16 frames from each dictionary video with a random shift and random frame rate n times, where n is proportional to the length of the video, and pass these clips through the I3D trunk then average the resulting vectors before they are processed by the MLP to produce the final dictionary embeddings. We find that this form of random sampling performs better than sampling 16 consecutive frames from the isolated signing videos (see supplementary material for more details). During pretraining, minibatches of size 4 are used; and colour, scale and horizontal flip augmentations are applied to the input video, following the procedure described in [Albanie et al. 2020]. The trunk parameters are then frozen and the MLP outputs are used as embeddings. Both datasets are described in detail in Sec. 5.4.1.

Minibatch sampling. To train the MLP given the pretrained I3D features, we sample data by first iterating over the set of labelled segments comprising the sparse annotations, \mathcal{M} , that accompany the dataset of continuous, subtitled sampling to form minibatches. For each continuous video, we sample 16 consecutive frames around the annotated timestamp (more precisely a random offset within 20 frames before, 5 frames after, following the timing study in [Albanie et al. 2020]). We randomly sample 10 additional 16-frame clips from this video outside of the labelled window, i.e., continuous background segments. For each subtitled sequence, we sample the dictionary entries for all subtitle words that appear in $\mathcal{V}_{\mathcal{L}}$ (see Fig. 5.3 for a sample batch formation).

Our minibatch comprises 128 sequences of continuous signing and their corresponding dictionary entries (we investigate the impact of batch size in Sec. 5.4.3). The embeddings are then trained by minimising the loss defined in Eqn.(5.3.1) in con-

junction with positive bags, $\mathcal{P}^{\text{bags}}$, and negative bags, $\mathcal{N}^{\text{bags}}$, which are constructed on-the-fly for each minibatch (see Fig. 5.3).

Optimization. We use a SGD optimizer with an initial learning rate of 10^{-2} to train the embedding architecture. The learning rate is decayed twice by a factor of 10 (at epoch 40 and 45). We train all models, including baselines and ablation studies, for 50 epochs at which point we find that learning has always converged.

Test time. To perform spotting, we obtain the embeddings learned with the MLP. For the dictionary, we have a single embedding averaged over the video. Continuous video embeddings are obtained with sliding window (stride 1) on the entire sequence. We calculate the cosine similarity score between the continuous signing sequence embeddings and the embedding for a given dictionary video. We determine the location with the maximum similarity as the location of the queried sign. We maintain embedding sets of all variants of dictionary videos for a given word and choose the best match as the one with the highest similarity.

5.4 Experiments

In this section, we first present the datasets used in this work (including the contributed BSLDICT dataset) in Sec. 5.4.1, followed by the evaluation protocol in Sec. 5.4.2. We illustrate the benefits of the *Watch, Read and Lookup* learning framework for sign spotting against several baselines with a comprehensive ablation study that validates our design choices (Sec. 5.4.3). Finally, we investigate three applications of our method in Sec. 5.4.4, showing that it can be used to (i) not only spot signs, but also identify the specific sign variant that was used, (ii) label sign instances in continuous signing footage given the associated subtitles, and (iii) discover faux amis between different sign languages.

5.4.1 Datasets

Although our method is conceptually applicable to a number of sign languages, in this work we focus primarily on BSL, the sign language of British deaf communities. We use BSL-1K [Albanie et al. 2020], a large-scale, subtitled and sparsely annotated dataset of more than 1000 hours of continuous signing which offers an ideal setting in which to evaluate the effectiveness of the *Watch, Read and Lookup*

Dataset	#Videos	Vocab.	#Signers
BSL-1K[Albanie et al. 2020]	273K	1,064	40
BSLDICT	14,210	9,283	148

Table 5.1: **Datasets:** We provide (i) the number of individual sign videos, (ii) the vocabulary size of the annotated signs, and (iii) the number of signers for BSL-1K and BSLDICT. BSL-1K is large in the number of annotated signs whereas BSLDICT is large in the vocabulary size. Note that we use a different partition of BSL-1K with longer sequences around the annotations as described in Sec. 5.4.1.

sign spotting framework. To provide dictionary data for the *lookup* component of our approach, we also contribute BSLDICT, a diverse visual dictionary of signs. These two datasets are summarised in Table 5.1 and described in more detail below.

BSL-1K [Albanie et al. 2020] comprises a vocabulary of 1,064 signs which are sparsely annotated over 1,000 hours of video of continuous sign language. The videos are accompanied by subtitles. The dataset consists of 273K localised sign annotations, automatically generated from sign-language-interpreted BBC television broadcasts, by leveraging weakly aligned subtitles and applying keyword spotting to signer *mouthings*. Please refer to [Albanie et al. 2020] for more details on the automatic annotation pipeline. In this work, we process this data to extract long videos with subtitles. In particular, we pad ± 2 seconds around the subtitle timestamps and we add the corresponding video to our training set if there is a sparse annotation word falling within this time window, assuming that the signing is reasonably well-aligned with its subtitles in these cases. We further consider only the videos whose subtitle duration is longer than 2 seconds. For testing, we use the automatic test set (corresponding to mouthing locations with confidences above 0.9). Thus we obtain 78K training and 3K test videos, each of which has a subtitle of 8 words on average and 1 sparse mouthing annotation.

BslDict. BSL dictionary videos are collected from a BSL sign aggregation platform [signbsl.com](https://www.signbsl.com/) [<https://www.signbsl.com/> n.d.], giving us a total of 14,210 video clips for a vocabulary of 9,283 signs. Each sign is typically performed several times by different signers, often in different ways. The dictionary videos are downloaded from 28 known website sources and each source has at least 1 signer. We used face embeddings computed with SENet-50 [J. Hu et al. 2019] (trained on

VGGFace2 [Q. Cao et al. 2018]) to cluster signer identities and manually verified that there are a total of 148 different signers. The dictionary videos are of isolated signs (as opposed to co-articulated in BSL-1K): this means (i) the start and end of the video clips usually consist of a still signer pausing, and (ii) the sign is performed at a much slower rate for clarity. We first trim the sign dictionary videos, using body keypoints estimated with OpenPose [Z. Cao et al. 2018] which indicate the start and end of wrist motion, to discard frames where the signer is still. With this process, the average number of frames per video drops from 78 to 56 (still significantly larger than co-articulated signs). To the best of our knowledge, BSLDICT is the first curated, BSL sign dictionary dataset for computer vision research, which will be made available. For the experiments in which BSLDICT is filtered to the 1,064 vocabulary of BSL-1K (see below), we have a total of 2,992 videos. Within this subset, each sign has between 1 and 10 examples (average of 3).

5.4.2 Evaluation Protocols

Protocols. We define two settings: (i) training with the entire 1064 vocabulary of annotations in BSL-1K; and (ii) training on a subset with 800 signs. The latter is needed to assess the performance on novel signs, for which we do not have access to co-articulated labels at training. We thus use the remaining 264 words for testing. This test set is therefore common to both training settings, it is either ‘seen’ or ‘unseen’ at training. However, we do not limit the vocabulary of the dictionary as a practical assumption, for which we show benefits.

Metrics. The performance is evaluated based on ranking metrics. For every sign s_i in the test vocabulary, we first select the BSL-1K test set clips which have a mouthing annotation of s_i and then record the percentage of dictionary clips of s_i that appear in the first 5 retrieved results, this is the ‘Recall at 5’ (R@5). This is motivated by the fact that different English words can correspond to the same sign, and vice versa. We also report mean average precision (mAP). For each video pair, the match is considered correct if (i) the dictionary clip corresponds to s_i and the BSL-1K video clip has a mouthing annotation of s_i , and (ii) if the predicted location of the sign in the BSL-1K video clip, i.e. the time frame where the maximum similarity occurs, lies within certain frames around the ground truth

		Train (1064)		Train (800)	
Embedding arch.	Supervision	Seen (264)		Unseen (264)	
		mAP	R@5	mAP	R@5
$\text{I3D}^{\text{BSLDICT}}$	Classification	2.68	3.57	1.21	1.29
$\text{I3D}^{\text{BSL-1K}}$ [Albanie et al. 2020]	Classification	13.09	17.25	6.74	8.94
$\text{I3D}^{\text{BSL-1K,BSLDICT}}$	Classification	19.81	25.57	4.81	6.89
$\text{I3D}^{\text{BSL-1K,BSLDICT}} + \text{MLP}$	Classification	36.75	40.15	10.28	14.19
$\text{I3D}^{\text{BSL-1K,BSLDICT}} + \text{MLP}$	InfoNCE	42.52	53.54	10.88	14.23
$\text{I3D}^{\text{BSL-1K,BSLDICT}} + \text{MLP}$	Watch-Lookup	43.65	53.03	11.05	14.62
$\text{I3D}^{\text{BSL-1K,BSLDICT}} + \text{MLP}$	Watch-Read-Lookup	48.11	58.71	13.69	17.79

Table 5.2: **The effect of the loss formulation:** Embeddings learned with the classification loss are suboptimal since they are not trained for matching the two domains. Contrastive-based loss formulations (NCE) significantly improve, particularly when we adopt the multiple-instance variant introduced as our Watch-Read-Lookup framework of multiple supervisory signals.

mouth timing. In particular, we determine the correct interval to be defined between 20 frames before and 5 frames after the labelled time (based on the study in [Albanie et al. 2020]). Finally, because BSL-1K test is class-unbalanced, we report performances averaged over the test classes.

5.4.3 Ablation Study

In this section, we evaluate different components of our approach. We first compare our contrastive learning approach with classification baselines. Then, we investigate the effect of our multiple-instance loss formulation. We provide ablations for the hyperparameters, such as the batch size and the temperature, and report performance on a sign spotting benchmark.

I3D baselines. We start by evaluating baseline I3D models trained with classification on the task of spotting, using the embeddings before the classification layer. We have three variants in Tab. 5.2: (i) $\text{I3D}^{\text{BSL-1K}}$ provided by [Albanie et al. 2020] which is trained only on the BSL-1K dataset, and we also train (ii) $\text{I3D}^{\text{BSLDICT}}$ and (iii) $\text{I3D}^{\text{BSL-1K,BSLDICT}}$. Training only on BSLDICT ($\text{I3D}^{\text{BSLDICT}}$) performs significantly worse due to the few examples available per class and the domain gap that must be bridged to spot co-articulated signs, suggesting that dictionary samples alone do not suffice to solve the task. We observe improvements with fine-tuning $\text{I3D}^{\text{BSL-1K}}$ jointly on the two datasets ($\text{I3D}^{\text{BSL-1K,BSLDICT}}$), which becomes our base feature extractor for the remaining experiments to train a shallow MLP.

Loss formulation. We first train the MLP parameters on top of the frozen I3D

Supervision	Dictionary Vocab	mAP	R@5
Watch-Read-Lookup	800 training vocab	13.69	17.79
Watch-Read-Lookup	9k full vocab	15.39	20.87

Table 5.3: **Extending the dictionary vocabulary:** We show the benefits of sampling dictionary videos outside of the sparse annotations, using subtitles. Extending the lookup to the dictionary from the subtitles to the full vocabulary of BSLDICT brings significant improvements for novel signs (the training uses sparse annotations for the 800 words, and the remaining 264 for test).

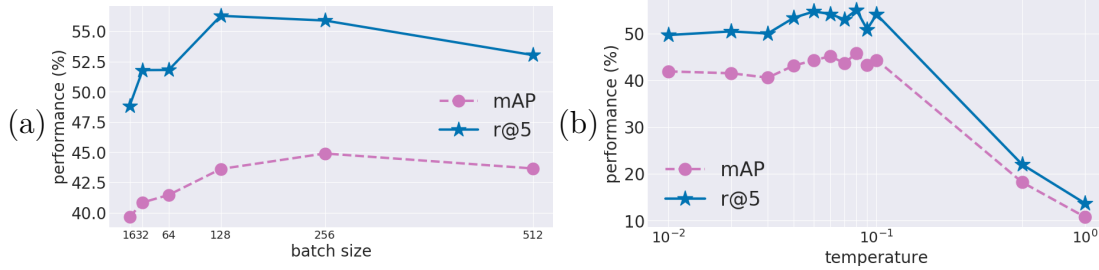


Figure 5.4: The effect of (a) the **batch size** that determines the number of negatives across sign classes and (b) the **temperature** hyper-parameter for the MIL-NCE loss in Watch-Lookup against mAP and R@5 (trained on the full 1064 vocab.)

trunk with classification to establish a baseline in a comparable setup. Note that, this shallow architecture can be trained with larger batches than I3D. Next, we investigate variants of our loss to learn a joint sign embedding between BSL-1K and BSLDICT video domains: (i) standard single-instance InfoNCE [Oord et al. 2018; P. Wu et al. 2016] loss which pairs each BSL-1K video clip with *one* positive BSLDICT clip of the same sign, (ii) Watch-Lookup which considers multiple positive dictionary candidates, but does not consider subtitles (therefore limited to the annotated video clips). Table 5.2 summarizes the results. Our Watch-Read-Lookup formulation which effectively combines multiple sources of supervision in a multiple-instance framework outperforms the other baselines in both *seen* and *unseen* protocols.

Extending the vocabulary. The results presented so far were using the same vocabulary for both continuous and dictionary datasets. In reality, one can assume access to the entire vocabulary in the dictionary, but obtaining annotations for the continuous videos is prohibitive. Table 5.3 investigates removing the vocabulary limit on the dictionary side, but keeping the continuous annotations vocabulary at 800 signs. We show that using the full 9k vocabulary from BSLDICT significantly improves the results on the unseen setting.

Batch size. Next, we investigate the effect of increasing the number of negative pairs by increasing the batch size when training with Watch-Lookup on 1064 categories. We observe in Figure 5.4(a) an improvement in performance with greater numbers of negatives before saturating. Our final Watch-Read-Lookup model has high memory requirements, for which we use 128 batch size. Note that the effective size of the batch with our sampling is larger due to sampling extra video clips corresponding to subtitles.

Temperature. Finally, we analyze the impact of the temperature hyperparameter τ on the performance of Watch-Lookup. We observe a major decrease in performance when τ approaches 1. We choose $\tau = 0.07$ used in [P. Wu et al. 2016; K. He et al. 2020] for all other experiments. Additional ablations are provided in the supplementary material.

BSL-1K Sign spotting benchmark. Although our learning framework primarily targets good performance on unseen continuous signs, it can also be naively applied to the (closed-vocabulary) sign spotting benchmark proposed by [Albanie et al. 2020]. We evaluate the performance of our Watch-Read-Lookup model and achieve a score of 0.170 mAP, outperforming the previous state-of-the-art performance of 0.160 mAP [Albanie et al. 2020].

5.4.4 Applications

In this section, we investigate three applications of our sign spotting method.

Sign variant identification. We show the ability of our model to spot specifically which variant of the sign was used. In Fig. 5.5, we observe high similarity scores when the variant of the sign matches in both BSL-1K and BSLDICT videos. Identifying such sign variations allows a better understanding of regional differences and can potentially help standardisation efforts of BSL.

Dense annotations. We demonstrate the potential of our model to obtain dense annotations on continuous sign language video data. Sign spotting through the use of sign dictionaries is not limited to mouthings as in [Albanie et al. 2020] and therefore is of great importance to scale up datasets for learning more robust sign language models. In Fig. 5.6, we show qualitative examples of localising multiple signs in a given sentence in BSL-1K, where we only query the words that occur

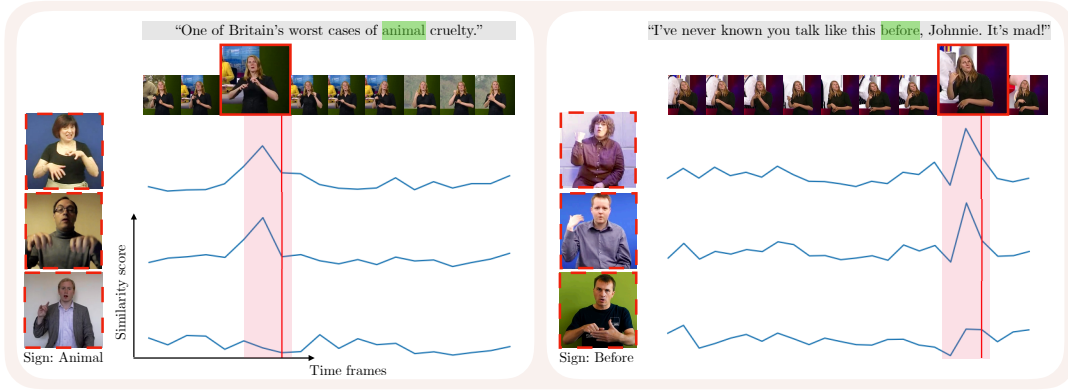


Figure 5.5: **Sign variant identification:** We plot the similarity scores between BSL-1K test clips and BSLDICT variants of the sign “animal” (left) and “before” (right) over time. The labelled mouthing times are shown by red vertical lines and the sign proposal regions are shaded. A high similarity occurs for the first two rows, where the BSLDICT examples match the variant used in BSL-1K.

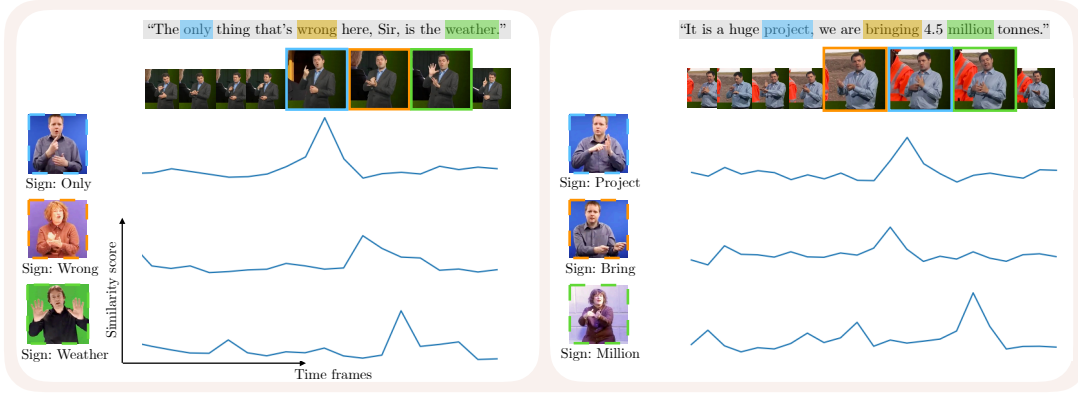


Figure 5.6: **Densification:** We plot the similarity scores between BSL-1K test clips and BSLDICT examples over time, by querying only the words in the subtitle. The predicted locations of the signs correspond to the peak similarity scores.

in the subtitles, reducing the search space. In fact, if we assume the word to be known, we obtain 83.08% sign localisation accuracy on BSL-1K with our best model. This is defined as the number of times the maximum similarity occurs within $-20/+5$ frames of the end label time provided by [Albanie et al. 2020].

“Faux Amis”. There are works investigating lexical similarities between sign languages manually [SignumMcKee and Kennedy 2000; Aldersson and McEntee-Atalianis 2007]. We show qualitatively the potential of our model to discover similarities, as well as “faux-amis” between different sign languages, in particular between British (BSL) and American (ASL) Sign Languages. We retrieve nearest neighbors according to visual embedding similarities between BSLDICT which has a 9K vocabulary and WLASL [D. Li et al. 2019], an ASL isolated sign language dataset, with a 2K vocabulary. We provide some examples in Fig. 5.7.



Figure 5.7: “**Faux amis**” in BSL/ASL: Same/similar manual features for different English words (left), as well as for the same English words (right), are identified between BSLDICT and WLASL isolated sign language datasets.

5.5 Conclusions

We have presented an approach to spot signs in continuous sign language videos using visual sign dictionary videos, and have shown the benefits of leveraging multiple supervisory signals available in a realistic setting: (i) sparse annotations in continuous signing, (ii) accompanied with subtitles, and (iii) a few dictionary samples per word from a large vocabulary. We employ multiple-instance contrastive learning to incorporate these signals into a unified framework. Our analysis suggests the potential of sign spotting in several applications, which we think will help in scaling up the automatic annotation of sign language datasets.

Acknowledgements. This work was supported by EPSRC grant ExTol. The authors would to like thank A. Sophia Koepke, Andrew Brown, Necati Cihan Camgöz, and Bencie Woll for their help. S.A would like to acknowledge the generous support of S. Carlson in enabling his contribution, and his son David, who bravely waited until after the submission deadline to enter this world.

Chapter 6

Read and Attend: Temporal Localisation in Sign Language Videos

The paper has been accepted for publication at the Computer Vision and Pattern Recognition Conference (CVPR), 2021.

Read and Attend:

Temporal Localisation in Sign Language Videos

Gül Varol^{1*} Liliane Momeni^{2*} Samuel Albanie^{2*}
Triantafyllos Afouras^{2*} Andrew Zisserman²

¹ LIGM, École des Ponts, Univ Gustave Eiffel, CNRS, France

² Visual Geometry Group, University of Oxford, UK

Abstract

The objective of this work is to annotate sign instances across a broad vocabulary in continuous sign language. We train a Transformer model to ingest a continuous signing stream and output a sequence of written tokens on a large-scale collection of signing footage with weakly-aligned subtitles. We show that through this training it acquires the ability to attend to a large vocabulary of sign instances in the input sequence, enabling their localisation. Our contributions are as follows: (1) we demonstrate the ability to leverage large quantities of continuous signing videos with weakly-aligned subtitles to localise signs in continuous sign language; (2) we employ the learned attention to *automatically* generate hundreds of thousands of annotations for a large sign vocabulary; (3) we collect a set of 37K *manually verified* sign instances across a vocabulary of 950 sign classes to support our study of sign language recognition; (4) by training on the newly annotated data from our method, we outperform the prior state of the art on the BSL-1K sign language recognition benchmark.

6.1 Introduction

Sign languages are visual languages that, for deaf communities, represent the natural means of communication [Rachel Sutton-Spence and Woll 1999]. Our goal in

*Equal contribution.

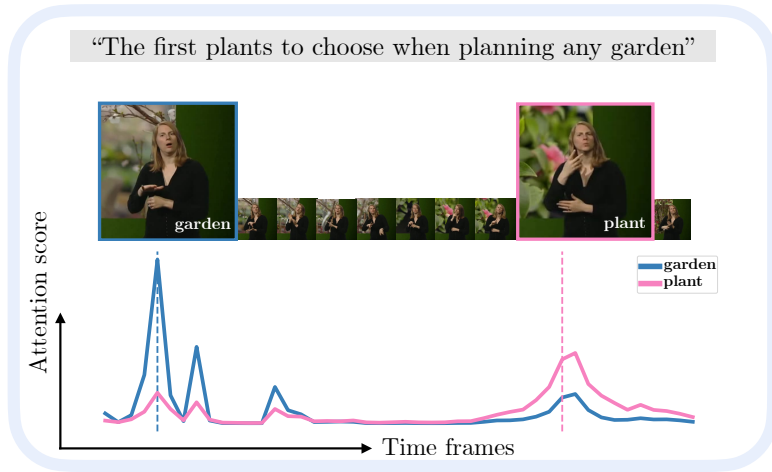


Figure 6.1: **Sign localisation emerges from sequence prediction.** In this work, we show that the ability to localise instances of signs emerges naturally by training a Transformer model [Vaswani et al. 2017] to perform a sequence prediction task on hundreds of hours of continuous signing videos with weakly-aligned subtitles.

this paper is to identify and temporally localise instances of signs among sequences of continuous sign language. Achieving automatic sign localisation enables a diverse range of practical applications: construction of sign language dictionaries to support language learners, indexing of signing content to enable efficient search and intelligent fast-forward to topics of interest, automatic sign language dataset construction, wake-word recognition for signers [Rodolitz et al. 2019] and tools to assist linguistic analysis of large-scale signing corpora.

In recent years, there has been a great deal of progress in temporally localising human actions within video streams [Shou et al. 2016; H. Zhao et al. 2019] and spotting words in spoken languages through aural [Coucke et al. 2019] and visual [Stafylakis and Tzimiropoulos 2017; Momeni et al. 2020a] keyword spotting methods. In both cases, a key driver of progress has been the availability of large-scale annotated datasets, enabling the powerful representation learning abilities of convolutional neural networks to be brought to bear on the task.

By contrast, annotated datasets for sign language are limited in scale and typically orders of magnitude smaller than their spoken counterparts [Bragg et al. 2019]. Widely used datasets such as RWTH-PHOENIX [Koller et al. 2015a; N. C. Camgoz et al. 2018] and the CSL dataset [J. Huang et al. 2018b] provide continuous sign annotations in the form of *glosses*¹ or free-form sentences, but lack precise temporal

¹Glosses are atomic lexical units used to annotate sign languages.

annotations and are limited in content diversity, vocabulary, and scale. Large-scale collections of continuous signing videos exist, but are limited to sparse annotation coverage [Albanie et al. 2020; Schembri et al. 2013].

In the absence of large-scale annotated training data, in this work we turn to a readily available and large-scale source: sign-interpreted TV broadcast footage together with subtitles of the corresponding speech in English. We propose to annotate this data with signs by training a Transformer [Vaswani et al. 2017] to predict, given input streams of continuous signing, the corresponding subtitles, and then using its trained attention mechanism to perform alignment from English words to signs.

This is a very challenging task: first, subtitles are only *weakly aligned* to the signing content—a sign may appear several seconds before or after its corresponding translated word appears in the subtitles, thus subtitles provide a relatively imprecise cue about the temporal location of a sign. Second, sign interpreters produce a *translation* of the speech that appears in subtitles, rather than a *transcription*—words in the subtitle may not correspond directly to individual signs produced by interpreters, and vice versa. Third, grammatical structures between sign languages and spoken languages differ considerably [Rachel Sutton-Spence and Woll 1999], and consequently the *ordering* of words in the subtitle is typically not preserved in the signing.

The core hypothesis motivating this approach is that *in order to solve the sequence prediction task, the attention mechanism of the Transformer must be capable of localising sign instances*. We demonstrate that by employing recent sign spotting techniques [Albanie et al. 2020; Momeni et al. 2020b] to coarsely align subtitles, sequence prediction is rendered tractable. One of the primary findings of this work is that, when performed at large scale (across hundreds of hours of continuous signing content), the ability to localise signs indeed emerges from the attention patterns of the sequence prediction model (Fig. 6.1).

We make the following four contributions: (1) by training on an appropriate sequence prediction task, we show that the attention mechanism of the Transformer learns to attend to specific signs, enabling their *localisation*; (2) we employ the learned attention to *automatically* generate hundreds of thousands of annotations

for a large sign vocabulary; (3) we collect a set of 37K *manually verified* sign instances across a vocabulary of 950 sign classes to support our study of sign language recognition; (4) by training on the newly annotated data from our method, we outperform the prior state of the art on the BSL-1K sign language recognition benchmark.

6.2 Related Work

Our approach relates to prior work on sign language recognition, translation, spotting, and in particular automatic annotation of sign language data. We present a discussion of these, followed by a brief overview of Transformers in natural language processing (NLP) and works in other domains using attention mechanisms for localisation.

Sign language recognition and translation. The computer vision community has a long history of efforts to develop systems for sign language recognition, reaching back to the 1980s [Tamura and Kawasaki 1988]. Initial work focused on hand-crafting features [Tamura and Kawasaki 1988; Fillbrandt et al. 2003] to model discriminative shape and motion cues and explored their usage in combination with Hidden-Markov Models [Starner 1995; Vogler and Metaxas 2001]. These works were followed by approaches that employed pose estimation as a basis for recognition [Ong et al. 2012; Pfister et al. 2014]. The community later transitioned to employing convolutional neural networks (CNNs) for appearance modelling [N. C. Camgoz et al. 2017]. In particular, the I3D architecture, originally developed for action recognition [Joao Carreira and Zisserman 2017], has proven to be effective for sign recognition [D. Li et al. 2019; D. Li et al. 2020b; Joze and Koller 2019; Albanie et al. 2021a; Momeni et al. 2020a]—we similarly employ this model in our work.

Continuous sign language recognition entails important challenges compared to *isolated* sign recognition, including epenthesis effects and co-articulation [Bragg et al. 2019] as well as the non-trivial definition of temporal boundaries between signs [Brentari 2009]. Towards dealing with these problems, [K. L. Cheng et al. 2020] uses the CTC loss [Graves et al. 2006] to infer an alignment between sequence-level annotations and visual input and introduces an auxiliary loss to use

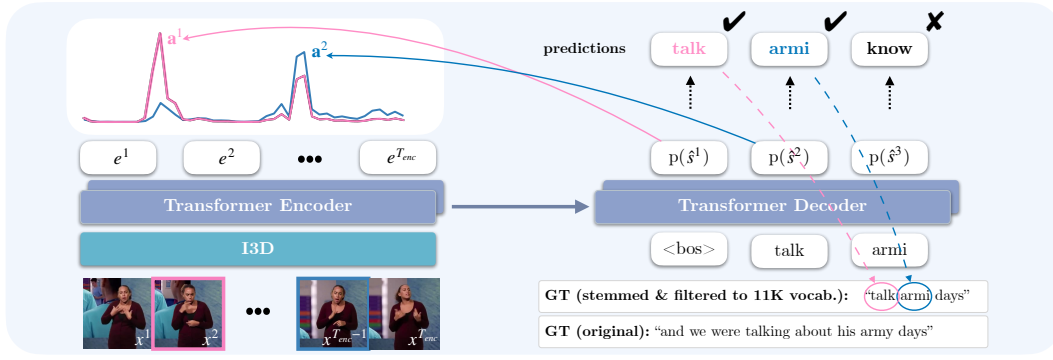


Figure 6.2: **Pipeline:** We use an I3D model pretrained on sign classification to extract spatio-temporal visual features by using a sliding window. We then train a 2-layer Transformer model to predict stemmed subtitles from the input video feature sequence. We use the learned model’s attention vectors to spot new instances of signs by checking which words in the predicted hypothesis overlap with the stemmed subtitle. For example, here the tokens “talk” and “armi”, found in the model’s hypothesis, also appear in the subtitle and are therefore retained, while “know” does not and is hence discarded. The location of a new spotting is determined by the index at which the corresponding encoder-decoder attention peaks. Note: we omit the sample index, subscript i , shared by all variables (described in Sec. 6.3).

the alignments as pseudolabels; while [Bull et al. 2020] proposes a graph convolutional network to automatically segment large sign language video sequences into short sentences, aligned with their subtitle transcription.

Recent works have applied sequence-to-sequence models to sign language translation. Camgöz et al. [N. C. Camgoz et al. 2018] use a two-stage pipeline that translates a video into gloss sequences then those into spoken language. Subsequent work [N. C. Camgoz et al. 2020b] replaces this framework with a Transformer model trained on frame-level features jointly for recognition and translation, while [N. C. Camgoz et al. 2020a] combines multiple articulators including face and upper body pose to train a translation system without gloss annotations. These approaches [N. C. Camgoz et al. 2018; N. C. Camgoz et al. 2020b; N. C. Camgoz et al. 2020a] have shown improvements towards translation in the restricted domain of discourse of the RWTH-PHOENIX-Weather-2014T German Sign Language (DGS) dataset [N. C. Camgoz et al. 2018]. Ko et al. [Ko et al. 2019b] train a sequence-to-sequence model using keypoint features on Korean Sign Language translation. Although these methods show promising results in constrained conditions, open-vocabulary sign language translation in the wild remains largely unsolved.

Automatic annotation of sign language data. Sign language datasets either

offer isolated gloss-level annotations of single signs, e.g., MSASL [Joze and Koller 2019], WLASL [D. Li et al. 2019], or are heavily constrained in visual domain and vocabulary, e.g., RWTH-PHOENIX [Koller et al. 2015a; N. C. Camgoz et al. 2018], KETI [Ko et al. 2019b] (only 105 sentences). Large-scale continuous sign language datasets, on the other hand, are not exhaustively annotated [Albanie et al. 2020; Schembri et al. 2017]. The recent efforts of Albanie et al. [Albanie et al. 2020] scale up the automatic annotation of sign language data, and construct the BSL-1K dataset with the help of a visual keyword spotter [Stafylakis and Tzimiropoulos 2018; Momeni et al. 2020a] trained on lip reading to detect instances of mouthed words as a proxy for spotting signs. *Sign spotting* refers to a specialised form of sign language recognition in which the objective is to find whether and where a given sign has occurred within a sequence of signing. It has emerged as an intermediate step to collect more annotated sign language data. With this goal, Momeni et al. [Momeni et al. 2020b] use dictionary lookups in subtitled videos and improve low-shot sign spotting. Other automatic annotation approaches include an automatic pipeline for active signer detection and sign language diarisation [Albanie et al. 2021a]. While these previous methods are *context-free*, in this work, we introduce a *context-aware* approach that can be used to localise signs automatically. In fact, while we profit from annotations obtained in prior works using mouthing cues [Albanie et al. 2020] and dictionaries [Momeni et al. 2020b], our approach differs considerably from theirs in method—we define the supervision directly on subtitles and formulate the problem as a sequence-to-sequence prediction task. We demonstrate the benefits of our approach empirically in Sec. 6.4.

Transformers in NLP. Incorporating an attention mechanism into encoder-decoder architectures led to a revolution in neural machine translation [Bahdanau et al. 2015] by reducing dependency on strong text alignment. Vaswani et al. [Vaswani et al. 2017] further extended this approach by replacing all recurrent and convolutional components of a sequence-to-sequence model with self-attention. Even though such methods implicitly model source-to-target alignment with attention, their primary focus is on translation performance, rather than word-alignment. [Garg et al. 2019] further studies how to simultaneously optimise for accurate word-alignment without sacrificing translation performance—we investigate a variant of their approach in Sec. 6.4.

Attention mechanisms for localisation. Cross-modal attention has been employed in the literature for various localisation problems such as visual grounding in videos [Huijuan Xu et al. 2019; Yuan et al. 2019; M. Liu et al. 2018; Jingyuan Chen et al. 2018] or images [Deng et al. 2018; L. Yu et al. 2018], keyword spotting in audio [Shan et al. 2018] or visual speech [Stafylakis and Tzimiropoulos 2018; Momeni et al. 2020a] and audio-visual sound source localisation [Arandjelovic and Zisserman 2017; Senocak et al. 2018; Harwath et al. 2018]. However, to the best of our knowledge, our work is the first to apply these ideas at large-scale to sign localisation from weakly-aligned subtitles.

6.3 Sign Localisation with Attention

In this section, we describe how we train a Transformer model on a weakly-supervised sign language sequence-to-sequence task and then use the trained model to perform sign localisation (see Fig. 6.2 for an overview).

Let $\mathcal{X}_{\mathfrak{L}}$ denote the space of sign language video segments \mathfrak{L} , and \mathcal{T} denote the space of subtitle sentences. Further, let $\mathcal{V}_{\mathfrak{L}} = \{1, \dots, V\}$ represent the *vocabulary* (an enumeration of spoken language tokens that correspond to signs that can be performed in \mathfrak{L}) and let \mathcal{S} denote a subtitled collection of I videos containing continuous signing, $\mathcal{S} = \{(x_i, s_i) : i \in \{1, \dots, I\}, x_i \in \mathcal{X}_{\mathfrak{L}}, s_i \in \mathcal{T}\}$. Our objective is to localise potential occurrences of signs in \mathcal{S} .

Transformer training with subtitled videos. To address this task, we propose to train a sequence-to-sequence model with attention. Given a video-subtitle pair $(x_i, s_i) \in \mathcal{S}$, we train a Transformer [Vaswani et al. 2017] to predict the target text sequence $s_i = (s_i^1, s_i^2 \dots, s_i^{T_{dec}})$ from the source video sequence $x_i = (x_i^1, x_i^2, \dots, x_i^{T_{enc}})$, one token at a time. Specifically, the Transformer’s encoder transforms x_i into an encoded sequence $enc(x_i) = (e_i^1, e_i^2, \dots, e_i^{T_{enc}})$. The decoder then attends on the encoded sequence and predicts the output sequence $\hat{s}_i = (\hat{s}_i^1, \hat{s}_i^2, \dots, \hat{s}_i^{T_{dec}})$ auto-regressively, factorising its joint probability into a product of individual conditionals:

$$p(\hat{s}_i | x_i) = \prod_{t=1}^{T_{dec}} p(\hat{s}_i^t | \hat{s}_i^1, \hat{s}_i^2 \dots \hat{s}_i^{t-1}, enc(x_i)). \quad (6.1)$$

Using the target subtitles s_i as the ground truth output sequences, we train the model to maximise their log likelihoods by minimising the following loss:

$$\mathcal{L} = -\mathbb{E}_{(x_i, s_i) \in \mathcal{S}} \log p(s_i | x_i) \quad (6.2)$$

Note that we assume access to a sparse collection of automatic sign annotations, $\mathcal{N} = \{(x_k, v_k) : k \in \{1, \dots, K\}, v_k \in \mathcal{V}_{\mathcal{L}}, x_k \in \mathcal{X}_{\mathcal{L}}, \exists (x_i, s_i) \in \mathcal{S} \text{ s.t. } x_k \subseteq x_i\}$, using mouthing cues [Albanie et al. 2020] and dictionaries [Momeni et al. 2020b]. In practice, we restrict the Transformer training on a subset of videos $\mathcal{S}_A \subseteq \mathcal{S}$, containing at least one of these annotations within the subtitle timestamps, formally $\mathcal{S}_A = \{(x_a, s_a) : a \in \{1, \dots, A\}, x_a \in \mathcal{X}_{\mathcal{L}}, \exists (x_k, s_k) \in \mathcal{N} \text{ s.t. } x_k \subseteq x_a\}$. This ensures approximate alignment between the source video and target subtitle. For arbitrary sequences in \mathcal{S} this is not guaranteed due to imperfect synchronisation between subtitles (corresponding to audio) and sign language interpretation. The goal of our training is therefore to exploit the knowledge of the unannotated words in the subtitles in \mathcal{S}_A in order to discover a new collection of (x, v) sign-video pairs (that is not included in \mathcal{N}) in the entire set \mathcal{S} .

Localising new sign instances with attention. Next, we describe how we use the Transformer model to look for new sign instances (see Fig. 6.2). After inputting the video sequence x_i into the trained model, we use a decoding strategy (e.g., greedy) to predict the output sequence \hat{s}_i and corresponding attention vectors $a_i = (\mathbf{a}_i^1, \mathbf{a}_i^2, \dots, \mathbf{a}_i^{T_{dec}}) \in R^{T_{dec} \times T_{enc}}$. We iterate over the predicted sequence \hat{s}_i and localise new sign instances *only* for the tokens predicted correctly (i.e., appearing in subtitle s_i); the video location is determined by the index at which the corresponding attention vector is maximised, to yield sets of (location, sign) pairs of the form: $\{(\arg\max_{j \in \{1, 2, \dots, T_{enc}\}} \mathbf{a}_i^t(j), s_i^t) : \hat{s}_i^t = s_i^t, t \in \{1, 2, \dots, T_{dec}\}\}$.

Implementation details. We represent the input video x_i with features extracted using a pretrained spatio-temporal convolutional neural network model, applied in a sliding window manner with a 4-frame stride. In particular, we train an I3D architecture [Joao Carreira and Zisserman 2017] on an extended set of automatic annotations \mathcal{N} that we obtain by combining the methods of [Albanie et al. 2020] and [Momeni et al. 2020b], to spot signs via mouthing cues and sign language dictionaries, respectively. We train with a single-sign classification ob-

jective and follow the same hyperparameters (e.g., 16-frame inputs) of the sign language recognition models in [Albanie et al. 2020]. The 1024-dimensional video features from I3D are used as input to the Transformer encoder.

To construct ground-truth text labels for our Transformer training, we stem the words in every subtitle under the assumption that variations of a written word could map to the same sign. We note that the many-to-many mapping between words and signs is a complex problem, which we do not explicitly deal with in this work. To establish a tractable problem, we define a vocabulary of 11,515 stems based on their frequency and occurrence within the automatic annotations \mathcal{N} . This is reduced from an original set of 40K words appearing in the full set of subtitles S . We further remove stop words for which there is often no sign correspondence. This approach resembles *glossing* sign language data, i.e., representing sign sequences with word sequences, without spoken language grammar.

Following common practice in the sequence-to-sequence literature [Vaswani et al. 2017], we train the model with teacher forcing [Williams and Zipser 1989], i.e. at every decoding step we provide the previous-step’s ground truth as input to the decoder. During inference we experiment with three different decoding strategies: auto-regressive greedy decoding, left-to-right beam search, and teacher forcing. With greedy decoding, we iterate over the available sequences and for each one, we select as new spottings all the words in the predicted hypothesis that appear in the reference subtitle. For beam search, we iterate over the predictions which overlap with the reference from the multiple returned hypotheses, and select for each predicted word the location with maximum attention score. We show results for another variant of beam search where we choose the hypothesis with the highest recall in the appendix. With teacher forcing, we do not use the token predictions of the model, but only the attention scores, which we associate with the next ground-truth word in the subtitle at every decoding step. Since we consider all words in the subtitles, this strategy provides good yield but no notion of the model’s confidence. In order to obtain a confidence score we use the following heuristic: For every sequence, a word found in the subtitle is automatically annotated if the attention peak for the corresponding decoding step is higher than a threshold τ .

When using Transformers with multiple attention heads, we obtain single attention scores by averaging the attention vectors of the individual heads. In Sec. 6.4.3 we

discuss results on combining attention from different decoder layers.

6.4 Experiments

This section is structured as follows: We first present the datasets used as well as the various training and evaluation protocols that we follow in our experiments (Sec. 6.4.1). Next, we show how we choose our pretrained input video features (Sec. 6.4.2). Then, we evaluate our Transformer models trained with these features and discuss different strategies for mining new instances to obtain an automatically annotated training set (Sec. 6.4.3). We show that, when adding our newly mined training samples, we outperform the previous state of the art on sign language recognition (Sec. 6.4.4). Finally, we provide qualitative results on two datasets (Sec. 6.4.5) and discuss limitations (Sec. 6.4.6).

6.4.1 Data and evaluation protocols

Datasets. We use BSL-1K [Albanie et al. 2020], a large-scale, subtitled and sparsely annotated dataset (for a vocabulary of 1,064 signs) of more than 1000 hours of continuous signing from sign language interpreted BBC television broadcasts. The programs cover a wide range of genres: from medical dramas and nature documentaries to cooking shows. In Sec. 6.4.5, we show qualitative examples on the RWTH-PHOENIX [N. C. Camgoz et al. 2018] dataset, which is significantly smaller in size and from weather broadcasts only, restricting the domain of discourse.

Transformer training and evaluation on $\text{Test}_{7K}^{\text{Loc}}$. To form the video-subtitle training data pairs, we sample 183K (\mathcal{S}_A) out of 685K subtitles from the BSL-1K training set (\mathcal{S}), in which there exists at least 1 automatic annotation (with a confidence score above 0.7) from the annotations collection \mathcal{N} . \mathcal{N} is formed by applying the method of [Albanie et al. 2020] on a large vocabulary of words beyond 1K to find signs via mouthing cues and applying the method of [Momeni et al. 2020b] to find signs via automatic dictionary spotting. See appendix for details on this step. Subtitles originally contain 9.8 words from the initial 40K words vocabulary on average, which is reduced to 4.4 words per subtitle from the 11K stems vocabulary after stemming and filtering. Corresponding videos are tightly

		Test _{2K} ^{Rec} [Albanie et al. 2020]				Test _{37K} ^{Rec}			
		2K inst. / 334 cls.				37K inst. / 950 cls.			
Training	#ann.	per-instance		per-class		per-instance		per-class	
		top-1	top-5	top-1	top-5	top-1	top-5	top-1	top-5
M [Albanie et al. 2020]§	169K	76.6	89.2	54.6	71.8	26.4	41.3	19.4	33.2
D	510K	70.8	84.9	52.7	68.1	60.9	80.3	34.7	53.5
M+D	678K	80.8	92.1	60.5	79.9	62.3	81.3	40.2	60.1

Table 6.1: **A new recognition test set Test_{37K}^{Rec} and an improved I3D model:** We employ the method of [Momeni et al. 2020b] to find signs via automatic dictionary spotting (D), significantly expanding the training and testing data obtained from mouthing cues by [Albanie et al. 2020] (M). We also significantly expand the test set by manually verifying these new automatic annotations from the test partition (Test_{2K}^{Rec} vs Test_{37K}^{Rec}). By training on the extended M+D data, we obtain state-of-the-art results, outperforming the previous work of [Albanie et al. 2020] and providing strong I3D features for the subsequent steps of our method. §The slight improvement in the performance of [Albanie et al. 2020] over the original results reported in that work is due to our denser test-time averaging when applying sliding windows (8-frame vs 1-frame stride).

extracted according to the subtitle timestamps, and are on average 3.52 seconds long.

For evaluating the localisation capability of the proposed method, we use the automatic annotations \mathcal{N} in the BSL-1K test set whose confidence scores are above 0.9, resulting in 7497 subtitle-video pairs with a total of 7661 annotations, referred to as Test_{7K}^{Loc}. We measure the localisation accuracy for the annotated words in each subtitle and only on the correct predictions: we consider a correct prediction to be also correctly localised if its predicted location lies within 8 frames of the annotation time. We also report recall and precision of the model’s predictions for each sequence by measuring the percentage of words in the subtitle that are predicted (recall) and the percentage of predicted words which appear in the subtitle (precision). For all three metrics, we report the average over all sequences in the test set.

Single-sign recognition benchmark. In order to justify the value of our automatic annotation approach with the Transformer model, we evaluate on the proxy task of single-sign recognition on trimmed videos by using our localised sign instances from the training set as labels for classification training. Similar to [Albanie et al. 2020; Joze and Koller 2019; D. Li et al. 2019], we adopt top-1 and

Tr.	Recall	Prec.	Loc. Acc. (GD)		Loc. Acc. (TF)	
			Att. layer 1/2/3	[avg]	Att. layer 1/2/3	[avg]
1L	15.8	36.4		65.9 [65.9]		44.8 [44.8]
2L	16.5	37.2		63.9/57.8 [66.1]		51.1/37.6 [44.5]
3L	16.5	36.9		62.5/60.8/16.4 [65.3]	51.4 /38.4/15.7	[46.4]

Table 6.2: **Localisation performance of attention layers.** We evaluate the performance of Transformers on $\text{Test}_{7K}^{\text{Loc}}$ for different number of encoder/decoder layers in the training (different rows). We report the localisation accuracy for the encoder-decoder attention scores from every layer, as well as the average over layers, for both teacher forcing (TF) and greedy decoding (GD) modes.

top-5 accuracy metrics reported with and without class-balancing.

We use the BSL-1K manually verified recognition test set with 2K samples [Albanie et al. 2020], which we denote with $\text{Test}_{2K}^{\text{Rec}}$, and significantly extend it to 37K samples as $\text{Test}_{37K}^{\text{Rec}}$. We do this by collecting new annotations from human annotators using the VIA tool [Dutta and Zisserman 2019] with a verification task as in [Albanie et al. 2020]. This extended test set reduces the bias towards signs with easily spotted mouthing cues (since we also include dictionary spottings [Momeni et al. 2020a]) and spans a larger fraction of the training vocabulary, i.e. 950 out of 1064 sign classes (vs 334 classes in the original benchmark $\text{Test}_{2K}^{\text{Rec}}$ of [Albanie et al. 2020]).

6.4.2 Comparison of video features

We first conduct experiments to determine which I3D video features are best suited as input to the Transformer model as described in Sec. 6.3. In Tab. 6.1, we demonstrate the benefits of combining annotations from both mouthing (M) [Albanie et al. 2020] and dictionary spottings (D) [Momeni et al. 2020b]. We show that our sign classification training using 678K automatic annotations obtains state-of-the-art performance on $\text{Test}_{2K}^{\text{Rec}}$, as well as our new and more challenging test set $\text{Test}_{37K}^{\text{Rec}}$. We therefore use this M+D model for the rest of our experiments. Note that all three models in Tab. 6.1 (M, D, M+D) are pretrained on Kinetics [Joao Carreira and Zisserman 2017], followed by video pose distillation as described in [Albanie et al. 2020]. We observed no improvements when initialising M+D training from M-only pretraining.

6.4.3 Mining training examples through attention

Next, we ablate different design choices for the Transformer model.

Which attention layer for sign-video alignment? Similarly to [Garg et al. 2019], we conduct an investigation into which decoder layer gives attention scores that are more useful for localising signs. We train three models, with 1, 2 and 3 encoder and decoder layers and report the localisation accuracy when using the attention from each layer separately, or an average of all layers. The results on $\text{Test}_{7K}^{\text{Loc}}$ in Tab. 6.2 suggest that averaging the attention scores over all layers gives the best localisation when using greedy auto-regressive decoding, while using the attention scores from the first decoder layer works best with teacher forcing. We note that this finding stands in contrast to those of [Garg et al. 2019] which concluded that the penultimate layer works better for word alignment in a machine translation task. We conjecture that the difference results from the different nature of the two domains, i.e., video versus text inputs. In terms of precision and recall, all three models perform similarly with rates at 37% and 16%, respectively. We continue with a 2-layer Transformer model for the rest of the experiments and given the observations in Tab. 6.2, we use the layer-averaged attention with greedy decoding and the first layer attention with teacher forcing.

Incorporating sparse annotations. As explained in Sec. 6.3, we make use of the available sparse annotations \mathcal{N} to restrict the training subtitles to those with at least 1 annotation. When removing this constraint, the model does not train as well, and reaches a recall of only 6.8% (vs 16.5%).

Here, we also report some of our findings by employing three additional strategies to improve the Transformer training using the sparse annotations \mathcal{N} . In all three cases, we observe no or minor gains (on $\text{Test}_{7K}^{\text{Loc}}$), at the cost of a more complex method and the need for annotations. Therefore, we do not integrate them in our final model and provide detailed results in appendix.

Alignment loss on sparse annotations: We investigate whether the sparse annotations \mathcal{N} could be used for supervising the sign-video alignment explicitly (similar to [Garg et al. 2019] in NLP). To this end, we define an additional loss that operates on the encoder-decoder attention to enforce a high response whenever there is known location information. We achieve this via an additional L2 loss term

between a 1D gaussian centered around the annotated time frame and the corresponding attention vector. While the localisation performance with teacher-forcing increases (58.7% vs 51.1%), it still remains lower compared to the corresponding greedy decoding result and we observe no significant gains for other metrics measured on the predictions.

Curriculum learning with sparse annotations: To provide warmup for the model training, we start by temporally trimmed video inputs around known sign locations \mathcal{N} . We gradually increase the number of annotations from 1 to 3, before we fully input the subtitle duration to the Transformer. We only observe minor improvements: 16.0% vs 15.8% recall with the 1-layer architecture.

Subtitle alignment through active signer detection and sparse annotations: To overcome the alignment noise present in the data, we apply an algorithm that combines a pose-based active signer detection [Albanie et al. 2021a] and the knowledge of sparse annotations \mathcal{N} . Specifically, we apply temporal shifts to subtitles such that their temporal midpoint aligns with the average time of any annotated signs they contain. We then apply affine transformations to the subtitles without annotations such that they fill the regions between those with annotations, subject to the hard constraint that the expansions do not overlap periods of inactive signing. This approach increases the amount of training subtitles with annotations to 230K; however, training with this new set does not improve recall (15.4% vs 16.5% with 2-layers).

Which decoding mechanism? To form a new annotated set for sign recognition training, we apply the trained Transformer models on the whole 685K training video-subtitle pairs of the BSL-1K dataset. In Tab. 6.3 we summarise and compare the yield of new training samples mined with the different decoding strategies we discussed in Sec. 6.3. We report the number of previously unannotated subtitles, for which the attention mechanism is able to localise signs, to demonstrate the benefits of our approach. We also report the amount of new annotations for both the full 11K vocabulary and the 1064-subset which is used for the proxy recognition evaluation. We observe that a significant number of new automatic sign annotations are obtained with our approach.

To compare the different decoding strategies, we train recognition models on the

Spotting mode	#subtitles unannot.	#ann. 11K	#ann. 1K	top-1 per-inst	top-1 per-cls
TF ($\geq .2$)	114K	290K	97K	22.2	4.7
TF ($\geq .1$)	408K	1.7M	545K	37.3	13.4
TF ($\geq .05$)	457K	2.3M	754K	38.7	14.4
TF ($\geq .05$) (align. loss)	457K	2.3M	757K	38.8	14.6
BS (10 best)	109K	329K	166K	49.6	22.7
GD (no subtitle filtering)	480K	1.4M	910K	50.6	22.6
GD (align. loss)	53K	188K	108K	53.6	24.8
GD	53K	188K	107K	53.9	<u>24.7</u>

Table 6.3: **Automatically annotating the training data:** We show the yield obtained from various decoding strategies in terms of number of additional annotations (left). Training models only with these annotations, we evaluate the recognition accuracy on $\text{Test}_{37K}^{\text{Rec}}$. Greedy decoding (GD) obtains better results than teacher forcing (TF) even when not filtering the predictions against the ground-truth subtitles. Neither including 10 best predictions from beam search (BS) nor using the model trained with the alignment loss influences the recognition evaluation significantly.

resulting training sets containing the new annotations and evaluate them on the proxy sign recognition task. Note that for faster training, we learn a 4-layer MLP architecture on top of the pre-extracted I3D video features (architecture and optimisation details are given in the appendix).

We observe that greedy decoding with the simple filtering mechanism (checking against ground truth) gives best downstream recognition performance on $\text{Test}_{37K}^{\text{Rec}}$. Teacher forcing, beam search and no filtering all yield larger but noisier training sets that result in lower performance. However, we note that the “no subtitle filtering” experiment assumes no access to ground-truth subtitles during annotation mining and uses all the predictions, while providing competitive recognition performance (50.6% vs 53.9%).

6.4.4 Comparison with other automatic annotations

In this section, we train for sign recognition on BSL-1K [Albanie et al. 2020] on various label sets, comparing different automatic annotation methods and showing that our new sign instances are complementary when added to training data, achieving state of the art. As in the previous experiments, we use the MLP architecture on frozen I3D features to compare the different annotation sets. This time we perform 3 trainings per model with different random seeds and report the

Training	#ann.	per-instance		per-class	
		top-1	top-5	top-1	top-5
A	107K	54.0 \pm 0.08	67.9 \pm 0.10	24.8 \pm 0.10	35.5 \pm 0.20
M [Albanie et al. 2020] [†]	169K	40.8 \pm 0.17	62.2 \pm 0.07	21.7 \pm 0.19	38.5 \pm 0.29
M+A	276K	58.5 \pm 0.17	75.5 \pm 0.02	30.4 \pm 0.04	45.9 \pm 0.26
D [Momeni et al. 2020b] [†]	510K	62.1 \pm 0.24	80.8 \pm 0.10	35.1 \pm 0.38	54.3 \pm 0.11
D+A	276K	64.2 \pm 0.08	81.7 \pm 0.07	36.0 \pm 0.26	54.0 \pm 0.32
M+D	678K	63.5 \pm 0.28	82.1 \pm 0.04	37.2 \pm 0.12	56.4\pm0.17
M+D+A	786K	65.0\pm0.14	82.6\pm0.02	37.9\pm0.07	56.3 \pm 0.02

Table 6.4: **Sign recognition on BSL-1K Test_{37K}^{Rec}**: We evaluate our 4-layer MLP classification models trained on video feature inputs for 1064-sign recognition for various training label sets: mouthing (M), dictionary (D), and our proposed attention (A) spottings. We obtain state-of-the-art results, by consistently improving over previous works when including our attention localisations. [†]The results are obtained from our MLP trained with the annotations from [Albanie et al. 2020] and our application of [Momeni et al. 2020b].

average and standard deviation.

Tab. 6.4 summarises the results on Test_{37K}^{Rec}. We first note that the MLP performance of M+D annotations matches and slightly outperforms that of I3D from Tab. 6.1 (63.5% vs 62.3%), validating the suitability of MLP for efficiently comparing annotation set quality. When compared to the visual keyword spotting through mouthing (M) [Albanie et al. 2020], our automatic attention localisations (A) show significant improvements. Furthermore, we observe consistent improvements when combining our new annotations with either the mouthing (M+A) or dictionary (D+A) annotations. Combining all available annotations (M+D+A), we achieve state-of-the-art performance (65%) outperforming previous work of [Albanie et al. 2020] (M: 40.8%), as well as a new much stronger baseline (D: 62.1%) that we establish in this work, which uses the new annotations obtained using sign language dictionaries for sign spotting [Momeni et al. 2020b]. Our final recognition model can be interpreted as distilling information from multiple sources (mouthing, dictionary, attention), each of which has access to a large training set.

We also evaluate the performance of our MLP trained on M+D+A annotations on the BSL-1K sign spotting benchmark proposed by [Albanie et al. 2020], following their protocol, and achieve a score of 0.174 mAP, outperforming the previous state-of-the-art performance of 0.170 mAP [Momeni et al. 2020b] and 0.159 mAP [Albanie et al. 2020].

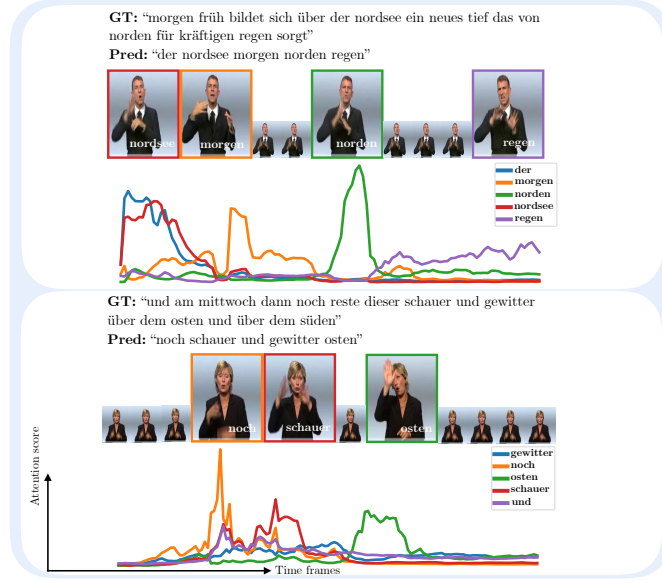


Figure 6.3: **Qualitative analysis on the RWTH-PHOENIX:** We show example sign localisation results on the test set of RWTH-PHOENIX 2014T. For each video clip, we show the ground-truth sentence as well as the predicted words from the Transformer model of [N. C. Camgoz et al. 2020b] which overlap with the target sentence. We plot attention scores over time frames for these predicted words and show the frame index at which the corresponding attention vector is maximised for a subset of the correctly predicted words.

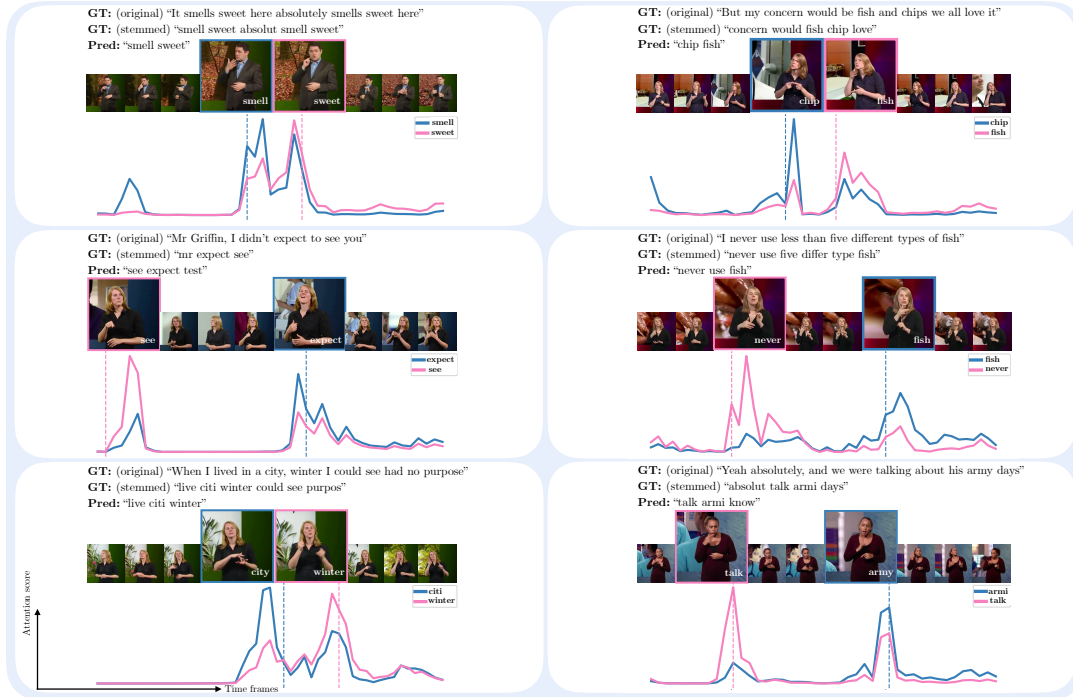


Figure 6.4: **Qualitative analysis on BSL-1K:** We show example sign localisation results on the BSL-1K test set ($\text{Test}_{7K}^{\text{Loc}}$). For each video clip, we show the original subtitle, the ground-truth stemmed and filtered to 11K vocabulary version, and the prediction of our Transformer model. We plot attention scores over time frames for the predicted words which overlap with the subtitle and for which we have annotated sign times in \mathcal{N} (shown by vertical dashed lines). We highlight the frame at which the corresponding attention vector is maximised.

6.4.5 Qualitative analysis

We demonstrate the potential of our Transformer model to localise sign instances through its attention mechanism. Fig. 6.4 shows qualitative examples of localising multiple signs, by plotting attention scores over video time frames for predicted words that occur in corresponding subtitles of the BSL-1K test set ($\text{Test}_{7K}^{\text{Loc}}$). We observe close alignment with the automatic annotations \mathcal{N} . One potential limitation of this approach for localisation is that the attention vector does not peak only at the corresponding sign location, but also on other signs suggesting that the predictions use context (e.g., “smell” and “sweet” in Fig. 6.4, top-left).

We also investigate whether this localisation ability extends to other datasets. In particular, we reproduce the translation method of Camgöz et al. [N. C. Camgoz et al. 2020b] on RWTH-PHOENIX 2014T [N. C. Camgoz et al. 2018] and similarly to [N. C. Camgoz et al. 2018], we visualise the attention score plots for predicted words in Fig. 6.3. We are unable to compute the localisation accuracy as sign annotation times are not available for RWTH-PHOENIX 2014T; however, we observe correct signs when indexing the frame at which the corresponding attention vector is maximised. This suggests that alignment emerges from the attention mechanism also for a full translation system.

6.4.6 Discussion

From our investigations in this work, we believe there are important and challenging problems to be solved before achieving large-vocabulary sign language *translation* from videos to spoken language. First, significantly expanding the coverage of the *vocabulary* of both languages is necessary, and the current state of the art only covers about 3K spoken language and 1K sign language vocabularies [N. C. Camgoz et al. 2020b]. In preliminary experiments, we found that a direct application of [N. C. Camgoz et al. 2020b] to translation on the significantly broader vocabulary of 40K contained within the subtitles of BSL-1K failed to converge to meaningful results (for more details see appendix). In this work, we have extended to an 11K spoken language vocabulary, but the NLP literature typically works with much larger vocabularies (e.g. a few hundred thousand words [Dai et al. 2019]). Our attempts to move to 40K words did not obtain sufficient-quality results. Second, the *alignment* between text and video is far from perfect in large-

scale sign language datasets which inserts significant amount of noise in training. Our automatic alignment attempts in this work did not obtain improvements. Relying on sparse annotations for approximate alignments limits the amount of data. Third, most of the works, including ours, focus on *interpreted* data, which has certain biases. In fact, the act of interpreting can cause a simplification in signing style and vocabulary, and even lead to a reduction in speed for comprehension [Bragg et al. 2019]. Datasets of native signers should be built to train strong, robust models that generalise at scale and in the wild. Given these observations, we believe that future work that specifically targets translation systems will benefit from addressing these challenges. We refer to the appendix for a discussion of broader impact.

6.5 Conclusions

We have presented an approach to localise signs in continuous sign language videos with weakly-supervised subtitles by leveraging the attention mechanism of a Transformer model trained on a video-to-text sequence prediction task. We find that state-of-the-art translation models have very low recall on a large-vocabulary dataset, but a satisfactory localisation accuracy through attention that allows us to annotate sign timings. We automatically annotate hundreds of thousands of new signing instances through our learned attention and validate their quality by using them to train a sign language recognition model that surpasses the state of the art on the BSL-1K benchmark as well as a more robust sign language benchmark which is 18 times larger. Future work can leverage our automatic annotations and recognition model for large-vocabulary sign language translation.

Acknowledgements. This work was supported by EPSRC grant ExTol and a Royal Society Research Professorship. We thank Cihan Camgöz, Himel Chowdhury and Abhishek Dutta for their help.

Chapter 7

Automatic dense annotation of large-vocabulary sign language videos

The paper has been accepted for publication at the European Conference on Computer Vision (ECCV), 2022.

Automatic dense annotation of large-vocabulary sign language videos

Liliane Momeni^{1*} Hannah Bull^{2*} K R Prajwal^{1*}
Samuel Albanie³ Gül Varol² Andrew Zisserman¹

¹ Visual Geometry Group, University of Oxford, UK

² LIGM, École des Ponts, Univ Gustave Eiffel, CNRS, France

³ Department of Engineering, University of Cambridge, UK

Abstract

Recently, sign language researchers have turned to sign language interpreted TV broadcasts, comprising (i) a video of continuous signing and (ii) subtitles corresponding to the audio content, as a readily available and large-scale source of training data. One key challenge in the usability of such data is the lack of sign annotations. Previous work exploiting such weakly-aligned data only found *sparse* correspondences between keywords in the subtitle and individual signs. In this work, we propose a simple, scalable framework to *vastly* increase the *density* of automatic annotations. Our contributions are the following: (1) we significantly improve previous annotation methods by making use of synonyms and subtitle-signing alignment; (2) we show the value of pseudo-labelling from a sign recognition model as a way of sign spotting; (3) we propose a novel approach for increasing our annotations of *known* and *unknown* classes based on *in-domain exemplars*; (4) on the BOBSL BSL sign language corpus, we increase the number of confident automatic annotations from 670K to 5M. We make these annotations publicly available to support the sign language research community.

*Equal contribution.

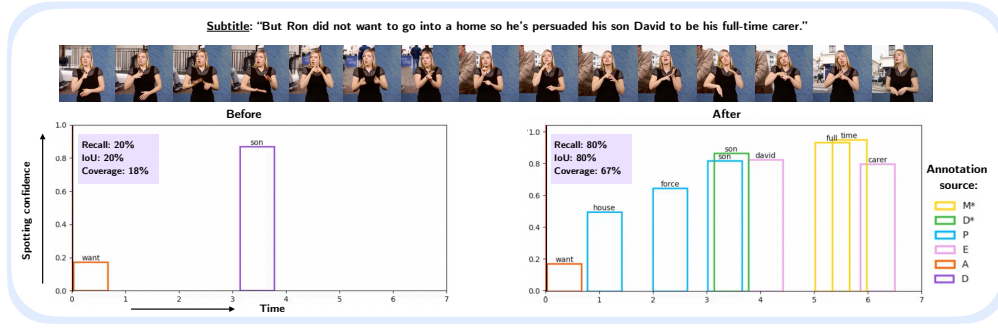


Figure 7.1: **Densification**: For continuous sign language, we show automatic sign annotation timelines, along with their confidence and annotation source, *before* and *after* our framework is applied. M, D, A refer to automatic annotations from previous methods from mouthings [Albanie et al. 2020], dictionaries [Momeni et al. 2020b] and the Transformer attention [Varol et al. 2021]. M*, D*, P, E, N refer to new and improved automatic annotations collected in this work. Annotation methods are compared in the appendix.

7.1 Introduction

Sign languages are visual-spatial languages that have evolved among deaf communities. They possess rich grammar structures and lexicons that differ considerably from those found among spoken languages [Rachel Sutton-Spence and Woll 1999]. An important factor impeding progress in automatic sign language recognition – in contrast to automatic speech recognition – has been the lack of large-scale training data. To address this issue, researchers have recently made use of sign language interpreted TV broadcasts, comprising (i) a video of continuous signing, and (ii) subtitles corresponding to the audio content, to build datasets such as Content4All [N. Camgoz et al. 2021] (190 hours) and BOBSL [Albanie et al. 2021b] (1460 hours).

Although such datasets are orders of magnitude larger than the long-standing RWTH-PHOENIX [N. C. Camgoz et al. 2018] benchmark (9 hours) and cover a much wider domain of discourse (not restricted to only weather news), the supervision they provide on the signed content is limited in that it is *weak* and *noisy*. It is weak because the subtitles are temporally aligned with the audio content and not necessarily with the signing. The supervision is also noisy because the presence of a word in the subtitle does not necessarily imply that the word is signed; and subtitles can be signed in different ways. Recent works have shown that training automatic sign language translation models on such *weak* and *noisy* supervision leads to low performance [N. Camgoz et al. 2021; Varol et al. 2021; Albanie et al.

2021b].

In an attempt to increase the value of such interpreted datasets, multiple works [Albanie et al. 2020; Momeni et al. 2020b; Varol et al. 2021] have leveraged the subtitles to perform lexical *sign spotting* in an approximately aligned continuous signing segment – where the aim is to determine *whether* and *when* a subtitle word is signed. Methods include using visual keyword spotting to identify signer mouthings [Albanie et al. 2020], learning a joint embedding with sign language dictionary video clips [Momeni et al. 2020b], and exploiting the attention mechanism of a transformer translation model trained on weak, noisy subtitle-signing pairs [Varol et al. 2021]. These works leverage the approximate subtitle timings and subtitle content to significantly reduce the correspondence search space between temporal windows of signs and spoken language words. Although such methods are effective at automatically annotating signs, they only find *sparse* correspondences between keywords in the subtitle and individual signs.

Our goal in this work is to produce *dense* sign annotations, as shown in Fig. 7.1. We define densification in two ways: (i) reducing gaps in the timeline so that we have a densely spotted signing sequence; and also (ii) increasing the number of words we recall in the corresponding subtitle. This process can be seen as automatic annotation of lexical signs. Automatic dense annotation of large-vocabulary sign language videos has a large range of applications including: (i) *substantially* improving recall for retrieval or intelligent fast forwards of online sign language videos; (ii) enabling *large-scale* linguistic analysis between spoken and signed languages; (iii) providing *supervision* and *improved alignment* for continuous sign language recognition and translation systems.

In this paper, we ask the following questions: (1) Can we improve current methods to improve the yield of automatic sign annotations whilst maintaining precision? (2) Can we increase the vocabulary of annotated signs over previous methods? (3) Can we ‘fill in the gaps’ that current spotting methods miss? The answer is yes, to all three questions, and we demonstrate this on the recently released BOBSL dataset of British Sign Language (BSL) signer interpreted video.

We make the following four contributions: (1) we significantly improve previous methods by making use of synonyms and subtitle-signing alignment; (2) we show

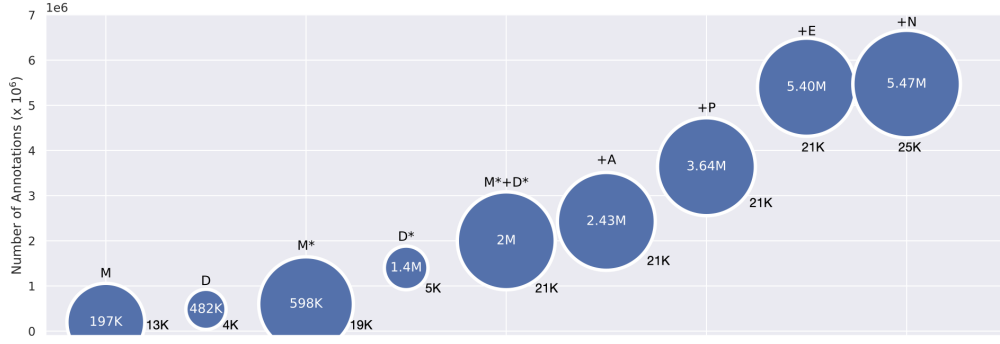


Figure 7.2: **Yield of automatic annotations and vocabulary size:** We highlight the increase in the number of automatic annotations and vocabulary size at each stage in our proposed framework. M, D, A refer to annotations from previous methods. M*, D*, P, E, N refer to new and improved annotations collected in this work. The number of annotations is shown within each circle. The vocabulary size is reported below each circle and also represented by the circle diameter.

the value of pseudo-labelling from a sign recognition model as a way of sign spotting; (3) we propose a novel approach for increasing our annotations of *known* and *unknown* sign classes based on in-domain exemplars; (4) we will make all 5 million automatic annotations publicly available to support the sign language research community. Our increased yield and vocabulary size is shown in Fig. 7.2. Our final vocabulary of 24.8K represents the vocabulary of English words (including named entities) from the subtitles which have been automatically associated to a sign instance; different words may have the same sign.

We note that this work is focused on *interpreted* data, which can differ from *conversational* signing in terms of style, vocabulary and speed [Bragg et al. 2019]. Although our long-term aim is to move to conversational signing, learning good representations of signs from interpreted data can be a ‘stepping stone’ in this direction. Moreover, non-lexical signs, such as a pointing sign and spatially located signs, are essential elements of sign language, but our method is limited to the annotation of lexical signs associated to words in the text.

7.2 Related Work

Our work relates to several themes which we give a brief overview of below.

Sign Spotting. One line of research has focused on the task of *sign spotting*, which seeks to detect signs from a given vocabulary in a target video. Early efforts

for sign spotting employed lower-level features (colour histograms and geometric cues) in combination with Conditional Random Fields [H.-D. Yang et al. 2008], Hidden Markov Models (HMMs) [Viitaniemi et al. 2014] and Sequential Interval Patterns [Ong et al. 2014] for temporal modelling. A related body of work has sought to localise signs while leveraging weak supervision from audio-aligned subtitles. These include the use of external dictionaries [D. Li et al. 2020b; Momeni et al. 2020b; T. Jiang et al. 2021] and other localisation cues such as mouthing [Albanie et al. 2020] and Transformer attention [Varol et al. 2021]. The performance of these approaches depends on the quality of the visual features, keywords, and the search window. In this work, we show improved yield of existing sign spotting techniques by employing automatic subtitle alignment techniques to adjust the time window and incorporating synonyms when forming the keywords. Going further beyond the spotting task explored in prior work, we use the automatic spottings to initiate additional algorithms for sign discovery based on *in-domain* exemplar matching (7.3.1). This is similar to dictionary-based sign spotting techniques [Momeni et al. 2020b; T. Jiang et al. 2021] except we do not source the exemplars from external dictionaries, avoiding the domain gap issue. Besides *in-domain sign* exemplars as in [T. Jiang et al. 2021], we explore the weak *subtitle* exemplars with unknown sign locations.

A recent progress in mouthing-based keyword spotting was presented by *Transpotter* [K. Prajwal et al. 2021]. This architecture comprises a transformer joint encoder of visual features and phoneme features that is trained to regress both the presence and location of the target keyword in a sequence from mouthing patterns. Preliminary small-scale experimental results reported by Prajwal et al. [K. Prajwal et al. 2021] demonstrated that Transpotter can perform visual keyword spotting in signing footage. Here, we showcase its suitability for the large-scale annotation regime, and further train it on sign language data to obtain a greater density of sign annotations.

In this work, we demonstrate the additional value of *pseudo-labelling* [Yarowsky 1995; Lee et al. 2013] with a sign classifier as an effective mechanism for sign spotting. While pseudo-labelling has been explored previously for category-agnostic sign segmentation [Renz et al. 2021b] and temporal alignment of glosses [Koller et al. 2017; K. L. Cheng et al. 2020] to the best of our knowledge, this is the

first use of pseudo-labelling for sign spotting by directly leveraging the predictions of a sign classifier in combination with a pseudo-label filter constructed from the subtitles themselves.

Sign Language Recognition. Efforts to develop visual systems for sign recognition stretch back to work in 1988 from Tamaura and Kawasaki [Tamura and Kawasaki 1988], who sought to classify signs from hand location and motion features. There were later efforts to design hand-crafted features for sign recognition [Charayaphan and Marble 1992; Starner 1995; Vogler and Metaxas 1997; Vogler and Metaxas 1998; Ong et al. 2012]. Deep convolutional neural networks then came to dominate sign representation [Koller et al. 2016], particularly via 3D convolutional architectures [Joze and Koller 2019; D. Li et al. 2019; Albanie et al. 2020; D. Li et al. 2020b] with extensions to focus model capacity around human skeletons [J. Huang et al. 2018a] and non-manual features [H. Hu et al. 2021b].

In the domain of continuous sign language recognition, in which the objective is to infer a sequence of sign glosses, prior work has explored HMMs [Bauer and Hienz 2000; Koller et al. 2015a] in combination with Dynamic Time Warping (DTW) [Jihai Zhang et al. 2014], RNNs [Cui et al. 2017] and architectures capable of learning effectively from CTC losses [H. Zhou et al. 2020b; K. L. Cheng et al. 2020]. Recently, sign representation learning methods inspired by BERT [Devlin et al. 2019] have shown the potential to learn effective representations for both isolated [H. Hu et al. 2021a] and continuous [Zhenxing Zhou et al. 2021] recognition. Koller [Koller 2020] provides an extensive survey of the sign recognition literature, highlighting the extremely limited supply of datasets with large-scale vocabularies suitable for continuous sign language recognition. In our work, we aim to take a step towards addressing this gap by developing “densification” techniques for constructing such datasets automatically.

Sign Language Translation. The task of translating sign language video to spoken language sentences was first tackled with neural machine translation by Camgöz et al. [N. C. Camgoz et al. 2018], who also introduced the PHOENIX-Weather-2014T dataset to facilitate research on this topic. Several frameworks have been proposed to employ transformers for this task [N. C. Camgoz et al. 2020b; K. Yin and Read 2020], with extensions to improve temporal modelling [D. Li et al. 2020a], multi-channel cues [N. C. Camgoz et al. 2020a] and signer inde-

pendence [Jin and Z. Zhao 2021]. Related work has also sought to contribute to progress on this task by exploiting monolingual data [H. Zhou et al. 2020a] and gloss sequence synthesis [Moryossef et al. 2021; D. Li et al. 2021]. To date, various works have shown promise on the PHOENIX-Weather 2014T [N. C. Camgoz et al. 2018] and CSL Daily [H. Zhou et al. 2020a] benchmarks. However, sign language translation has not yet been demonstrated for a large vocabulary across multiple domains of discourse. Differently from the works above, this paper focuses on developing methods that are applicable to large/open vocabulary regimes.

Weakly-supervised Object Discovery and Localisation. Our approach is also related to the rich body of literature on object cosegmentation [Rother et al. 2006; Joulin et al. 2010; G. Kim et al. 2011; Rubinstein et al. 2013], weakly supervised object localisation [M. H. Nguyen et al. 2009; Deselaers et al. 2010; Z. Shi et al. 2013; C. Wang et al. 2014; Gokberk Cinbis et al. 2014], object colocalisation [Tang et al. 2014; Joulin et al. 2014] and unsupervised object discovery and localisation [Cho et al. 2015; Vo et al. 2021]. Here, we propose an algorithm for discovering and localising novel signs (i.e. for which we have no labelled examples), but instead have weak supervision in the form of subtitles containing keywords of interest. Moving beyond initial work that sought to learn from subtitles in an aligned setting [Ali Farhadi and David Forsyth 2006], classical approaches for sign discovery using subtitles have included Multiple Instance Learning where the subtitles are considered as positive and negative bags for a particular keyword [Buehler et al. 2009; Kelly et al. 2010; Pfister et al. 2013] and a priori mining [Cooper and Bowden 2009]. Differently from these works, we first bootstrap our sign discovery process with sign spotting to both obtain initial candidates and learn robust sign representations, then propagate these examples across video data by leveraging the similarities between the resulting representations together with noisy constraints imposed by the subtitle content.

7.3 Densification

Our goal is to leverage several ways of sign spotting to achieve dense annotation on continuous signing data. To this end, we introduce both new sources of automatic annotations, and also improve the existing sign spotting techniques. We start

by presenting two new spotting methods using in-domain exemplars: to mine more sign instances with individual *exemplar signs* (Sec. 7.3.1) and to discover novel signs with weak *exemplar subtitles* (Sec. 7.3.2). We also show the value of pseudo-labelling from a sign recognition model for sign spotting (Sec. 7.3.3). We then describe key improvements to previous work which substantially increase the yield of automatic annotations (Sec. 7.3.4). Finally, we present our evaluation framework to measure the quality of our sign spottings in a large-vocabulary setting (Sec. 7.3.5). The contributions of each source of annotation are assessed in the experimental results.

7.3.1 Mining more Spottings through In-domain Exemplars (E)

The key idea is: given a continuous signing video clip and a set of exemplar clips of a particular sign, we can use the exemplars to search for that sign within the video clip. In our case, the exemplars are obtained from other *automatic* spotting methods (M^* , D^* , A , P), described in Sec. 7.3.3 and Sec. 7.3.4, and come from the same *domain* of sign language interpreted data, i.e. the same training set. We hypothesise that signs from the same domain are more likely to be signed in a similar way and in turn help recognition; in contrast, for example, to signs from a different domain such as dictionaries.

Formally, suppose we have a reference video V_0 in which we wish to localise a particular sign w , whose corresponding word occurs in the subtitle. We also have N video exemplars V_1, \dots, V_N of the sign w . For each video, V_i , let \mathcal{C}_i denote the set of possible temporal locations of the sign w and let $c = (f, p) \in \mathcal{C}_i$ denote a candidate with features f at temporal location p . We compute a score map between our reference video V_0 and each exemplar V_1, \dots, V_N by computing the cosine similarity between each feature at each position in $c_0 \in \mathcal{C}_0$ and $(c_1, c_2, \dots, c_n) \in \mathcal{C}_1 \times \dots \times \mathcal{C}_N$. This results in N score maps of dimension $|\mathcal{C}_0| \times |\mathcal{C}_i|$ for $i = 1 \dots N$. We then apply a max operation over the temporal dimension of the exemplars, giving us N vectors of length $|\mathcal{C}_0|$, which we call M_1, \dots, M_N .

We subsequently apply a voting scheme to find the location of the common sign w in V_0 . Specifically, we let $L = \frac{1}{N} \sum_{i=1}^N \mathbb{1}_{(M_i > h)}$ for a threshold h , where the

vector $\mathbb{1}_{(M_i > h)}$ takes the value 1 for entries of M_i which are greater than h and 0 otherwise. The candidate location of w in V_0 is then $c = (f, p) \in \mathcal{C}_0$ where p corresponds to the position of the maximum non-zero entry in the vector L (see Fig. 7.3 for a visual illustration). If there are multiple maxima, we assign p to be the midpoint of the largest connected component. If all entries in L are zero, we conclude w is not present. We perform two variants of this approach using mean and max pooling of the score maps (instead of voting); these are described in the appendix. We note that for a given signing sequence, we only focus on finding signs for words in the subtitle that have *not* been annotated by other methods.

7.3.2 Discovering Novel Sign Classes (N)

One limitation of our proposed method in Sec. 7.3.1 is that we are only able to collect more sign instances from a *closed* vocabulary, determined by sign exemplars obtained from other methods (described in Sec. 7.3.3 and Sec. 7.3.4). Here, we extend our approach to localise *novel* signs, for which we have no exemplar signs but whose corresponding word appears in the subtitle text. We follow our approach described in Sec. 7.3.1, computing score maps between our reference video and exemplar subtitles (instead of exemplar signs, see Fig. 7.3). We note that by ‘exemplar subtitle’, we are referring to the video frames corresponding to the subtitle timestamps. Non-lexical signs, such as pointing signs or pause gestures, are very common in sign language. To avoid annotating such non-lexical signs as the common sign across V_0 and V_1, \dots, V_N , we also choose N^- negative subtitle exemplars $U_1 \dots U_{N^-}$ presumed to not contain w (due to the absence of w in the subtitle). We compute L^+ and L^- using the score maps from positive exemplars V_1, \dots, V_N and negative exemplars U_1, \dots, U_{N^-} respectively. We then let $L = L^+ - L^-$. Implementation details on the number of positive and negative exemplars used can be found in the appendix.

7.3.3 Pseudo-labelling as a Form of Sign Spotting (P)

We propose to re-purpose a pretrained large-vocabulary sign classification model (see vocabulary expansion in Sec. 7.3.5) for the task of sign spotting. Specifically, we predict a sign class from a fixed vocabulary for each time step in a continuous signing video clip. We subsequently filter the predicted signs to words which occur

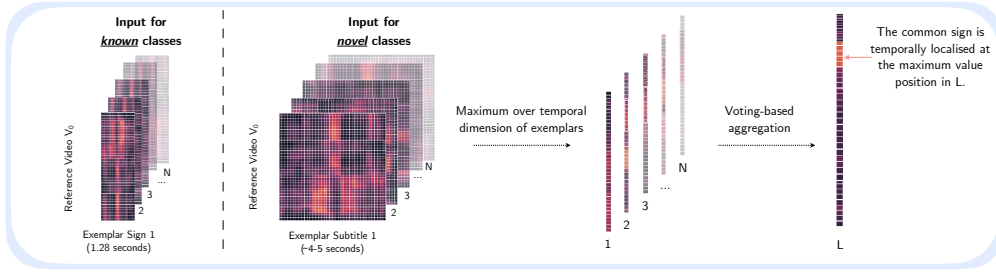


Figure 7.3: **Sign spotting through exemplars to find instances of known classes (E) and novel classes (N)**: By comparing a reference video V_0 to a set of exemplars (either sign exemplars for known sign class instances or weak subtitle exemplars for novel sign class instances), we can find the common lexical sign in the collection. We (1) form a set of score maps by calculating the cosine similarities between reference and exemplar representations; (2) we perform a maximum operation over the temporal dimension of exemplars; (3) we apply a voting-based aggregation to find the temporal location of the common sign in V_0 . The duration of exemplar signs is fixed.

in the corresponding English subtitle. Similarly to [Varol et al. 2021], here the task is to recognise the sign from scratch, without a query keyword. The subtitle is only used as a post-processing step to filter out signs which are less likely performed (due to absence in the subtitle).

7.3.4 Improving the Old (M^* , D^*)

Here, we briefly describe our improvements over the existing sign spotting techniques, additional details are provided in the appendix.

Better Mouthings with an Upgraded KWS from Transpotter [K. Prajwal et al. 2021]. In previous work [Albanie et al. 2021b], an improved BiLSTM-based visual-only keyword spotting model of Stafylakis et al. [Stafylakis and Tzimiropoulos 2018] from [Momeni et al. 2020a] (named “P2G [Stafylakis and Tzimiropoulos 2018] baseline”) is used to automatically annotate signs via mouthings. In this work, we make use of the recently proposed transformer-based *Transpotter* architecture [K. Prajwal et al. 2021], provided by the authors, that achieves state-of-the-art results in visual keyword spotting on lipreading datasets. We follow the procedure described in [Albanie et al. 2020; Albanie et al. 2021b] to query words in the subtitle in continuous signing video clips.

Finetuning KWS on Sign Language Data through Bootstrapping. The visual keyword spotting Transpotter architecture in [K. Prajwal et al. 2021] is trained on silent speech segments, which differ considerably from signer mouthings. In fact,

signers do not mouth continuously and sometimes only partly mouth words [Boyes Braem and RL Sutton-Spence 2001]. In order to reduce this severe domain gap, we propose a dual-stage finetuning strategy. First, we extract high-confidence mouthing annotations using the pre-trained Transpotter from [K. Prajwal et al. 2021] on the BOBSL training data. We query for the words in the subtitle and obtain the temporal localization of the word in the video. We finetune on this pseudo-labeled data using the same training pipeline of [K. Prajwal et al. 2021], where the spotted mouthings (word-video pairs) act as positive samples. For the negative samples, we pair a given word with a randomly sampled video segment from the dataset. As we observe the Transpotter to predict a large number of false positives, we remedy this by sampling a larger number of negative pairs in each batch. We also do a second round of fine-tuning by training on the pseudo-labels from the finetuned model of the first stage. We did not achieve significant improvements with further iterations.

Better Search Window with Subtitle Alignment with SAT [Bull et al. 2021a]. One challenge in using sign language interpreted TV broadcasts is that the original subtitles are not aligned to the signing, but to the audio track. In [Albanie et al. 2021b], a signing query window is defined as the audio-aligned subtitle timings together with padding on both sides to account for the misalignment. We automatically align spoken language text subtitles to the signing video by using the SAT model introduced in [Bull et al. 2021a], trained on manually aligned and pseudo-labelled subtitles as described in [Albanie et al. 2021b]. By using subtitles which are better aligned to the signing, we reduce the probability of missing spottings.

Better Keywords with Synonyms and Similar Words. To determine whether a keyword belongs to a subtitle, previous works [Albanie et al. 2021b] check whether the raw form, the lemmatised form, or the text normalised form (e.g. *two* instead of *2*) appears in the subtitle text. We notice that this is sub-optimal as multiple words may correspond to the same sign, often due to (i) English synonyms, (ii) identical signs for similar words, or (iii) ambiguities in spoken language. For example, *dad* and *father* or *today* and *now* can be the same signs in BSL. In this work, we investigate whether the automatic annotation yield could be improved by querying words beyond the subtitle, by querying synonyms and similar

words to the words in the subtitle. We collect the additional words to query through (i) English synonyms from WordNet [Feinerer and Hornik 2020], (ii) the metadata present in online sign language dictionaries such as SignBSL¹ [Momeni et al. 2020b] and BSL Sign-Bank² which provide a set of ‘related words’ for each sign video entry; (iii) words with GloVe [Pennington et al. 2014] cosine similarity above 0.9 to account for ambiguities in spoken language.

7.3.5 Evaluation Framework

Our framework consists of three stages: (a) a costly end-to-end classification training to learn sign category aware video features given an initial set of sign-clip annotation pairs; (b) a lightweight classification training given pre-extracted video features for a large number of annotations; (c) a sliding window evaluation of the trained lightweight model by comparing dense sign predictions against the subtitles (see Sec. 7.4.1). These stages are illustrated in Fig. 7.4. Note that the *annotations* we refer to are always *automatically* localised sign spottings from continuous videos using subtitle information. The motivation for the video backbone and lightweight classifier is purely related to computational costs. Unlike traditional video recognition datasets, we work with untrimmed video data of 1400 hours, where the set of sign-clip pairs is not fixed. Instead, our goal is to increase the number of sign-clip pairs within the continuous stream, and assess the quality of the expanded annotation yield on the proxy task of continuous sign language recognition. Next, we describe the training stages for the video backbone and the lightweight classifier.

Improving the I3D Feature Extractor through Vocabulary Expansion.

Following previous works [Joze and Koller 2019; D. Li et al. 2019; Albanie et al. 2020; Albanie et al. 2021b], we use the I3D spatio-temporal convolutional architecture to train an end-to-end sign recognition model. We input 16 consecutive RGB frames and output class probabilities. The details about optimisation are provided in the appendix. As explained above, this model forms the basis of sign video representation which corresponds to the spatio-temporally pooled latent embedding before the classification layer. The prior work of [Albanie et al. 2021b]

¹www.signbsl.com

²bslsignbank.ucl.ac.uk

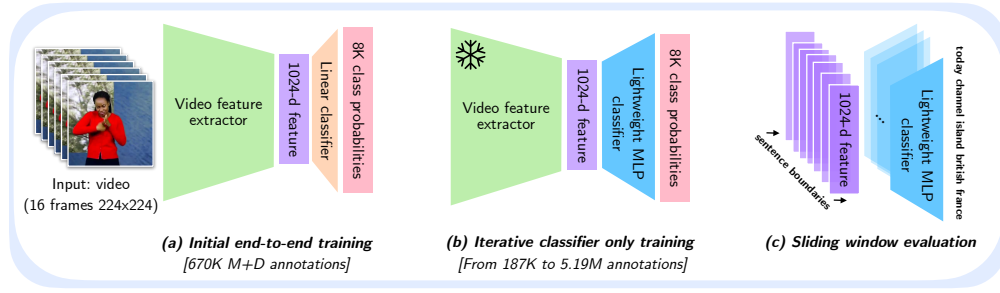


Figure 7.4: **Evaluation framework:** (a) Video features are obtained by training an I3D architecture end-to-end given $M + D$ annotations from [Albanie et al. 2021b]. The I3D ingests 16-frames of video and has a linear classifier for 8K sign categories. The end-to-end training is a costly procedure which is not affordable to repeat for each set of our new sign spottings that are on the order of several million training samples. (b) As new sets of spottings are generated, a light weight MLP classifier is trained on the pre-extracted I3D features. This relatively inexpensive training procedure means that we benefit from new annotations without the expense of end-to-end training. (c) The MLP is applied in a sliding window fashion to the signing sequence to generate sign predictions.

trains this classifier on the BOBSL dataset (see Sec. 7.4.1) with 2K categories obtained through the vocabulary of mouthing spottings. As a first step, we perform a vocabulary expansion and construct a significantly increased vocabulary of 8K categories. This is achieved by including each sign that has at least 5 training spottings above 0.7 confidence from both mouthing (M) and dictionary (D) annotations. The confidence for the mouthing annotation corresponds to the probability that a text keyword (corresponding to the sign) is mouthed at a certain time frame, as computed in [Albanie et al. 2020]. The confidence for the dictionary annotation corresponds to the cosine similarity (normalised between 0-1) between the representations of a dictionary clip of the sign and the continuous signing at each time frame, as in [Momeni et al. 2020b]. The resulting M+D training set comprises 670K annotations, with a long-tailed distribution. Furthermore, we note that the categories are noisy where multiple categories may correspond to the same sign, and vice versa. Despite this noise, we empirically show that this model provides better performance than its 2K-vocabulary counterpart. We use our improved I3D model for two purposes: as the frozen feature extractor and as the source of pseudo-labelling for sign spotting (see Sec. 7.3.3).

Lightweight Sign Recognition Model. Following [Varol et al. 2021], we opt for a 4-layer MLP module (with one residual connection) to assess the quality of different sets of annotations. Given pre-extracted features, this model is trained

for sign recognition into 8K categories. We note that we do not train on a larger vocabulary to avoid the presence of many singletons in the training set. The efficiency of the MLP allows faster experimentation to analyse the value of each of our sign spotting sets. The input is one randomly sampled feature around the sign spotting location (the receptive field of one feature 16 frames). The MLP weights are randomly initialised. Additional training and implementation details are given in the appendix.

7.4 Experiments

We start by describing our dataset and evaluation metrics (Sec. 7.4.1). We then present experimental results on the contribution of each source of annotation and show qualitative examples (Sec. 7.4.2).

7.4.1 Data and Evaluation Protocol

BOBSL [Albanie et al. 2021b] is a public dataset consisting of British Sign Language interpreted BBC broadcast footage, along with English subtitles corresponding to the audio content. The data contains 1,962 episodes, which have a total duration of 1,467 hours spanning 426 different TV shows. BOBSL has a total 1,193K subtitles covering a total vocabulary of 78K words. We note that in this work we use the word *subtitle* to refer to the processed BOBSL sentences from [Albanie et al. 2021b] as opposed to the raw subtitles. There are a total of 39 signers in the dataset. Further dataset statistics can be found in [Albanie et al. 2021b]. For a subset of 36 episodes in BOBSL, referred to as SENT-TEST in [Albanie et al. 2021b], the English subtitles have been manually aligned *temporally* to the continuous signing video. We make use of this test set to evaluate the quality of our predicted automatic annotations. SENT-TEST covers a total duration of 31 hours and contains 20,870 English subtitles. The total vocabulary of English words is 13,641, of which 5,604 are singletons. The 3 signers in SENT-TEST are different to the signers in the training set, this enables signer-independent BSL recognition to be evaluated.

Evaluation protocol. Given an English subtitle and the *temporally* aligned

Subtitle:	I hope they taste really good!	Recall = 0.75 (MLP predicts 3 out of 4 words in subtitle)
Lemmatise, no stopwords (L+NS):	hope taste really good	
MLP predictions:	do hope miss mouth taste delicious delicious good do do do	IoU = 0.5 (intersection=3, union=6)
L+NS+combine <i>synonym</i> classes:	hope miss mouth taste good	
Subtitle:	So one of the first indicators of spring?	Recall = 0.75 (MLP predicts 3 out of 4 words in subtitle)
Lemmatise, no stopwords (L+NS):	one first indicator spring	
MLP predictions:	receive green year grow sell true start start start spring spring spring spring one one fast charles	IoU = 0.27 (intersection=3, union=11)
L+NS+combine <i>synonym</i> classes:	receive green year spring sell true first one fast charles	

Figure 7.5: **Evaluation illustration on sample prediction:** We illustrate the processing applied to the predicted sign sequence from the MLP predictions and corresponding English subtitle for calculating our metrics. As the MLP model predicts one sign per time-step, some predictions are repeated and irrelevant words appear at transition periods between signs, decreasing the IoU. Some signs are not predicted as they are not signed, showing the limitations of using the subtitle to measure performance.

continuous signing video clip, we evaluate our predicted signs for the clip using (i) *intersection over union* (IoU); (ii) *recall* between signs and the English word sequence; and (iii) *temporal coverage*: this is defined as the proportion of frames in the clip assigned to signs that occur in the word sequence, where a sign is given a fixed duration of 16 frames (for 25Hz video). Note that none of these metrics depend on the word order of the English subtitle, only the words it contains. All metrics are rescaled from the range 0-1 to 0-100 percentage for readability.

For this evaluation, stop words are filtered out since often they are not signed. This reduces the number of test subtitles from 20,870 to 20,547: subtitles such as “is it?”, “Oh!”, “but no” are removed. The sign and word sequences are also lemmatised. We also remove repetitions from the predicted sign sequence and allow the prediction of synonyms of words in the English subtitle. This processing is highlighted in Fig. 7.5, where the IoU and recall are computed for a pair of predicted signs and English text. While this evaluation is suboptimal due to the simplified word-sign correspondence assumption, it tests the capacity of the sign recognition model in a large-vocabulary scenario, necessary for open-vocabulary sign language technologies.

Note, the predicted signs for a clip can be produced in two ways. In the first way, the signs are obtained from the automatic annotations using knowledge of the content of the English subtitle – we refer to these as *Spottings*. In the second, signs are predicted directly from the clip using the MLP sign predictions, without access to the corresponding English subtitle. These are referred to as *MLP predictions*. Spottings are evaluated using all the words; this metric is important to monitor how dense we can automatically annotate the data. The MLP evaluation is limited

Table 7.1: **Comparison of I3D video features:** We highlight the improved performance of I3D on the test set (SENT-TEST) when trained on a larger vocabulary (8K instead of 2K) with more samples (670K instead of 426K).

Annot. source	Num. I3D train annot.	Vocab. size	I3D predictions (subtitle independent)		
			Recall	IoU	Coverage
M [Albanie et al. 2020]+D [Momeni et al. 2020b]	426K	2K	25.5	6.4	15.5
M [Albanie et al. 2020]+D [Momeni et al. 2020b]	670K	8K	26.3	7.9	16.3

Table 7.2: **Improved mouthing and dictionary spottings:** We evaluate different sets of spottings and their respective MLP predictions. M [K. Prajwal et al. 2021] shows our finetuned version for all the rows in the last block. We quantify the effects of subtitle alignment and querying synonyms. We also show the oracle performance and a translation baseline.

Annotation source	Subtitle alignment	Synonyms	Training set			Spottings [full] (subtitle dependent)			MLAP predictions [8K] (subtitle independent)		
			full vocab	#ann. [full]	#ann. [8K]	Recall	IoU	Coverage	Recall	IoU	Coverage
Oracle			-	-	-	-	-	-	86.7	86.3	55.2
Translation baseline [Albanie et al. 2021b]			-	-	-	-	-	-	11.7	8.3	7.6
M [Albanie et al. 2020]			13.6K	197K	187K	2.5	2.2	1.3	15.1	3.2	8.7
M [K. Prajwal et al. 2021] (no finetuning)			21.5K	725K	661K	9.4	8.3	4.9	20.4	4.8	11.9
M [K. Prajwal et al. 2021]			18.6K	445K	412K	7.1	6.5	3.9	23.6	4.8	13.8
M [K. Prajwal et al. 2021] (M*)	✓		19.6K	598K	552K	8.9	8.2	4.9	27.4	6.3	16.7
M [K. Prajwal et al. 2021]	✓	✓	19.6K	1.38M	1.25M	11.8	10.4	6.1	25.3	6.2	16.3
D [Momeni et al. 2020b]			4.4K	482K	482K	6.5	6.3	3.7	24.0	7.2	15.1
D [Momeni et al. 2020b]	✓		4.5K	535K	535K	7.0	6.9	4.0	24.2	7.3	15.3
D [Momeni et al. 2020b] (D*)	✓	✓	5.0K	1.40M	1.39M	12.5	11.6	7.0	26.0	7.3	16.9
M*+ D*	✓	✓ (D-only)	20.9K	2.00M	1.94M	19.0	17.6	10.5	29.0	7.9	18.4
M*+ D*+ A [Varol et al. 2021]	✓	✓ (D-only)	20.9K	2.43M	2.37M	21.9	20.1	11.8	29.6	9.1	19.0

to the fixed classification vocabulary (of size 8K in our experiments). We note that when different annotations are combined, the sign spotting methods are applied independently.

7.4.2 Results

Comparison of Video Features. By finetuning our Kinetics pretrained I3D model on BOBSL M+D annotations from [Albanie et al. 2021b] using an 8K vocabulary instead of a 2K vocabulary, we improve predictions on the test set, as shown in 7.1. We increase the recall from 25.5 to 26.3 and the coverage from 15.5 to 16.3. We therefore use the 8K M+D model for the rest our experiments as the frozen feature extractor. We note that we restrict the M+D annotations to the high-confidence ones (over 0.8 threshold) used for the I3D baseline in [Albanie et al. 2021b], as these present an appropriate signal-to-noise ratio. We use the same threshold for subsequent automatic annotations unless stated otherwise.

Oracle. As the MLPs are trained on a restricted 8K vocabulary, it is not possible to predict the full vocabulary of 13,641 words present in the test set subtitles. Furthermore, not all words in the subtitle are signed and vice versa. This means

a recall, IoU and coverage of 100% is not achievable between predicted signs and English subtitle words. However, we propose an oracle in Tab. 7.2 whereby we measure the recall and IoU assuming each word in the subtitle, which either falls within the 8K vocabulary or corresponds to a synonym of a word in the 8K vocabulary, is signed and correctly predicted. The oracle achieves a recall of 86.7 and IoU of 86.3. For the coverage metric, we assume each correctly predicted sign has a duration of 16 frames and no signs overlap. The resulting oracle coverage is 55.2. This low coverage is partly due to the signer pausing within subtitles and also due to the presence of non-lexical signs. In fact, the percentage of fully lexical signs in three other sign language corpora (Auslan [Johnston 2012], ASL [Johnston 2012] and LSF [Belissen et al. 2020b]) is estimated to be only 70-85% of total signing.

Translation Baseline. Although the goal in this work is not translation, but achieving dense annotations, we can nevertheless compare our MLP predictions to the translation baseline in [Albanie et al. 2021b]. Using the test set translation predictions from this model, we perform the same processing as highlighted in Fig. 7.5 to calculate our metrics. As shown in Tab. 7.2, all our simple MLP models clearly outperform the transformer-based translation model used in [Albanie et al. 2021b], demonstrating that we are able to recognise more signs in the English subtitle.

Improving Mouthing and Dictionary Spottings. As shown in Tab. 7.2, by using the Transpotter [K. Prajwal et al. 2021] for spotting mouthings M, our yield of total annotations triples from 197K to 725K. The quality of these new annotations is reflected in the increased performance of the MLP: the recall increases from 15.1 to 20.4 and the coverage from 8.7 to 11.9. Finetuning the keyword spotter on sign language data through pseudo-labelling also helps considerably despite the drop in the number of training annotations since there are less false positives; recall increases from 20.4 to 23.6 and coverage from 11.9 to 13.8. Subtitle alignment improves the yield of both mouthing and dictionary annotations, as shown in Tab. 7.2. This translates to a significant boost for mouthings on the MLP performance; the recall increases from 23.6 to 27.4 and the coverage from 13.8 to 16.7. For dictionary annotations, the improvement by using aligned subtitles is less striking. By querying synonyms when searching for mouthings, the yield more than doubles. However, these additional annotations seem to be quite noisy as

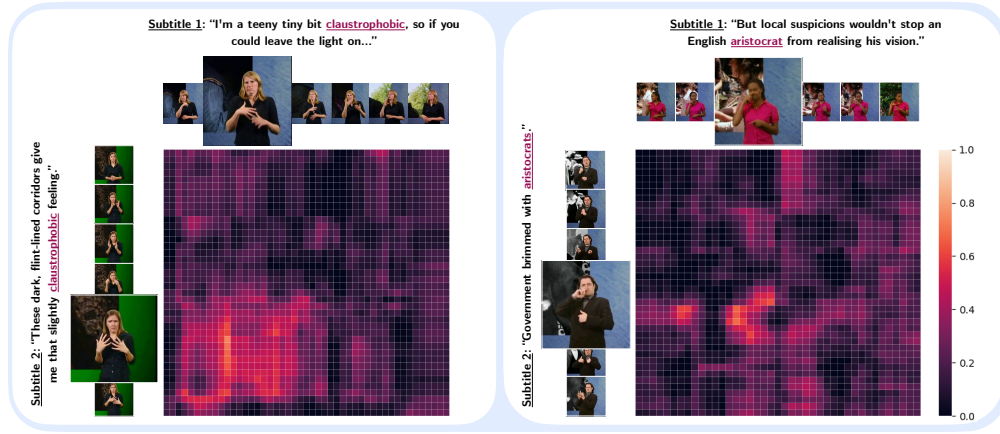


Figure 7.6: **Discovering novel sign classes (N)**: For two pairs of continuous signing sentences, we plot the score maps (as described in Sec. 7.3.1) between their feature sequences. We highlight the ability of our approach to spot novel sign classes.

they decrease the performance of our MLP. Due to the nature of sign language interpretation, it is possible that signers are far more likely to mouth a word which is actually in the written subtitle than a synonym of that word. We therefore do not query synonyms for mouthing spottings. For dictionary spottings, we observe the opposite effect. By incorporating synonyms, the yield of dictionary spottings more than doubles and the recall of the MLP predictions also increases from 24.2 to 26.0. We denote our best performing mouthing and dictionary spottings with M^* and D^* , respectively. Adding attention spottings from [Albanie et al. 2021b] (with a threshold of 0) adds around 400K additional annotations and boosts the MLP performance; increasing recall from 29.0 to 29.6 and coverage from 18.4 to 19.0, compared to the oracle recall of 86.7 and coverage of 55.2.

Sign Recognition as a Form of Pseudo-labelling. Pseudo-labels P are a source of over 1M new annotations (when using a threshold of 0.5) on top of our best M^* , D^* , A spottings. As shown in Tab. 7.4, they greatly increase the spottings recall from 21.9 to 25.4 and coverage from 11.8 to 13.9, while only marginally increasing the recall and coverage for MLP predictions. As the pseudo-labels come from our 8K I3D model in Tab. 7.1 whose frozen features are also used for training the MLP, P may not be providing additional information for our downstream evaluation. Nevertheless, they provide a great source of additional spottings (not found by previous methods) for our goal of dense annotation.

Mining more Examples of Known and Novel Sign Classes with In-

Table 7.3: **Ablation on mining exemplar-based spottings for known signs E:** We perform different ablations for mining known signs which have been unannotated by previous methods (M*, D*, A, P). We experiment with the source of exemplar data (same episode, same signer, all data), the confidence of exemplar signs (0,0.5,0.8), the number of samples of exemplar data (5,10,20) and the pooling mechanism (average, max, vote). We evaluate on the test set (SENT-TEST).

Ann. src.	ex. data	ex. thres	ex. #	ex. pooling	Training set			Spottings [full] (subtitle dependent)			MLP predictions [8K] (subtitle independent)		
					full vocab	#ann. [full]	#ann. [8K]	Recall	IoU	Coverage	Recall	IoU	Coverage
E	same ep.	0	var	avg	11.6K	869K	833K	10.4	9.6	5.8	25.1	6.9	15.3
E	same signer	0	20	avg	15.9K	505K	421K	7.8	7.5	4.4	23.1	5.6	14.2
E	all	0	20	avg	16.7K	351K	252K	5.7	5.7	3.3	21.5	5.1	13.4
E	all	0.5	20	avg	16.6K	370K	261K	5.9	5.8	3.4	21.9	5.2	13.5
E	all	0.8	20	avg	16.6K	458K	358K	7.4	7.3	4.3	25.2	6.2	15.7
E	all	0.8	20	max	15.4K	1.48M	1.38M	20.2	18.6	10.8	27.6	8.4	17.7
E	all	0.8	10	max	15.4K	1.07M	982K	15.2	14.0	8.3	27.9	8.0	17.7
E	all	0.8	5	max	15.3K	740K	664K	10.7	10.0	6.0	27.6	7.6	17.4
E	all	0.8	20	vote	15.9K	1.76M	1.63M	25.8	23.3	13.5	28.4	8.5	18.1
E	all	0.8	10	vote	15.8K	1.32M	1.21M	20.0	18.1	10.7	28.4	8.3	18.1

Table 7.4: **Pseudo-label spottings P & Exemplar-based sign spottings for known E and novel classes N:** We highlight the boost in annotations by adding our pseudo-label annotations (P) as well as exemplar-based spottings of known (E) and novel (N) classes. We evaluate Spottings and MLP predictions on the test set (SENT-TEST). For the novel classes, we only show the evaluation of spottings since these are beyond the 8K training vocabulary of the MLP.

Annotation source	Training set			Spottings [full] (subtitle dependent)			MLP predictions [8K] (subtitle independent)		
	full vocab	#ann. [full]	#ann. [8K]	Recall	IoU	Coverage	Recall	IoU	Coverage
M* + D* + A [Varol et al. 2021] + P	20.9K	3.64M	3.56M	25.4	23.5	13.9	29.8	8.9	19.2
M* + D* + A [Varol et al. 2021] + P + E	20.9K	5.40M	5.19M	45.3	40.7	23.3	30.7	9.5	19.8
M* + D* + A [Varol et al. 2021] + P + E + N	24.8K	5.47M	-	45.6	40.9	23.4	-	-	-

domain Exemplars. By explicitly querying words in the subtitle text which are not present in our annotations, we can obtain significantly more annotations. Tab. 7.3 shows multiple methods to use exemplar signs to find additional annotations for these signs. The best performing method takes spotting exemplars from across the whole training set, irrespective of signer or episode, and uses the voting scheme described in Sec. 7.3.1 to localise signs. By using 20 spotting exemplars, we acquire 1.63M additional annotations. An MLP model trained *only* on these additional annotations achieves a recall of 28.4 and coverage of 18.1. Tab. 7.4 illustrates the impact of combining these additional annotations from spotting exemplars to M*, D*, A and P annotations. With the additional exemplar-based annotations E, recall increases from 29.8 to 30.7 and coverage increases from 19.2 to 19.8, where the oracle recall and coverage are 86.7 and 55.2. Furthermore, by mining instances of novel sign classes N (see Fig. 7.6), we increase our total

vocabulary to 24.8K and total number of annotations to 5.47M.

7.5 Conclusion

Progress in sign language research has been accelerated in recent years due to the availability of large-scale datasets, in particular sourced from interpreted TV broadcasts. However, a major obstacle for the use of such data is the lack of available sign level annotations. Previous methods [Albanie et al. 2020; Momeni et al. 2020b; Varol et al. 2021] only found *sparse* correspondences between keywords in the subtitle and individual signs. In our work, we propose a framework which scales the number of confident automatic annotations from 670K to 5.47M (which we make publicly available). Potential future directions for research include: (1) increasing our number of annotations by incorporating context from *surrounding* signing to resolve ambiguities; (2) investigating *linguistic* differences between spoken English and British Sign Language such as the different word/sign ordering; (3) leveraging our automatic annotations for sign language translation.

Acknowledgements. This work was supported by EPSRC grant ExTol, a Royal Society Research Professorship and the ANR project CorVis ANR-21-CE23-0003-01. LM would like to thank Sagar Vaze for helpful discussions. HB would like to thank Annelies Braffort and Michèle Gouiffès for the support.

Part III

Sequence recognition in Sign Language

Chapter 8

Weakly-supervised Fingerspelling Recognition in British Sign Language Videos

The paper has been accepted for publication at the British Machine Vision Conference (BMVC), 2022.

Weakly-supervised Fingerspelling Recognition in British Sign Language Videos

K R Prajwal^{1*} Hannah Bull^{2*} Liliane Momeni^{1*}

Samuel Albanie³ Gül Varol² Andrew Zisserman¹

¹ Visual Geometry Group, University of Oxford, UK

² LIGM, École des Ponts, Univ Gustave Eiffel, CNRS, France

³ Department of Engineering, University of Cambridge, UK

Abstract

The goal of this work is to detect and recognize sequences of letters signed using fingerspelling in British Sign Language (BSL). Previous fingerspelling recognition methods have not focused on BSL, which has a very different signing alphabet (e.g., two-handed instead of one-handed) to American Sign Language (ASL). They also use manual annotations for training. In contrast to previous methods, our method only uses weak annotations from subtitles for training. We localize potential instances of fingerspelling using a simple feature similarity method, then automatically annotate these instances by querying subtitle words and searching for corresponding mouthing cues from the signer. We propose a Transformer architecture adapted to this task, with a multiple-hypothesis CTC loss function to learn from alternative annotation possibilities. We employ a multi-stage training approach, where we make use of an initial version of our trained model to extend and enhance our training data before re-training again to achieve better performance. Through extensive evaluations, we verify our method for automatic annotation and our model architecture. Moreover, we provide a human expert annotated test set of 5K video clips for evaluating BSL fingerspelling recognition methods to support sign language research.

*Equal contribution.

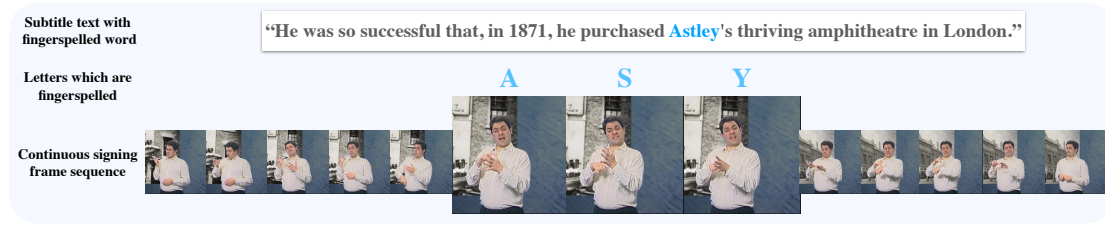


Figure 8.1: **Fingerspelling recognition.** We study the task of recognising a sequence of BSL fingerspelled letters in a continuous signing window in sign language interpreted TV broadcast data. We exploit the accompanying English subtitles to automatically collect training data. However, the task remains very challenging as (i) the signer may fingerspell only a subset of the letters – as shown above, although the subtitle contains the word ‘Astley’, only the letters ‘A’, ‘S’ and ‘Y’ are fingerspelled – and (ii) due to occlusions from the hands, different letters can visually look very similar.

8.1 Introduction

Fingerspelling in signed languages is a means to encode words from the surrounding written language into sign language via a manual alphabet, i.e. one sign per letter. Words from a written language with no known sign may be fingerspelled, such as names of people and places. Additionally, some signs are derived from finger-spelled words, for example, initialized signs in BSL such as ‘F’ for ‘father’ or ‘KK’ for ‘kitchen’ [Rachel Sutton-Spence and Woll 1999]. Padden & Gunsauls [Padden and Gunsauls 2003] estimate that signers fingerspell 12-35% of the time in ASL. Within the BSL videos used in this work [Albanie et al. 2021b], we estimate roughly 5-10% based on the duration of automatically detected fingerspelling content. Consequently, it is important to incorporate automatic fingerspelling recognition methods to be able to *exhaustively* transcribe signs in continuous sign language videos.

Some sign languages, such as ASL and LSF, use a one-handed manual alphabet, and others, including BSL and Auslan, use a two-handed manual alphabet [Schembri and Johnston 2007]. One-handed manual alphabets typically do not involve significant wrist movements, while two-handed fingerspelling resembles other lexical signs, making it relatively more difficult to detect fingerspelling beginning and end times from a longer signing sequence. The presence of two-hand movements further results in occlusions, making it challenging to differentiate between certain characters, e.g. especially vowels.

In this work, we focus on BSL, i.e. the more challenging two-handed fingerspelling.

Given a sign language video, our goal is to temporally locate the fingerspelling segments within the video (i.e. detection) and to transcribe the fingerspelled letters to text (i.e. recognition). To this end, we design a Transformer-based model that jointly performs both detection and recognition. Our key contribution lies in the data collection procedure used to automatically obtain training data for this task, which is applicable to any sign language videos that have approximately-aligned subtitle translations. We also provide the first large-scale benchmark for BSL fingerspelling recognition based on the recently released BOBSL dataset [Albanie et al. 2021b]. Our experiments on this benchmark demonstrate promising results with a 53.3 character error rate on this challenging task while only using weak supervision.

Previous works building fingerspelling datasets rely on manual annotation, either by expert annotators resulting in limited data [B. Shi et al. 2018], or by crowdsourcing noisy large-scale annotations [B. Shi et al. 2019]. In contrast to these works on ASL fingerspelling, we introduce a practical methodology to automatically annotate fingerspelling in the presence of subtitled sign language video data, allowing to scale up the data size, and potentially to be applicable to other sign languages. Starting from a small number of manually annotated fingerspelling exemplars, we use an embedding space to find numerous similar instances of fingerspelling in the corpus. To annotate these instances, we exploit the observation that signers often simultaneously mouth the words (i.e. silent speech with lip movements) which they fingerspell. We obtain an initial set of annotations by querying potential words from the subtitles, especially proper nouns, and identifying mouthing cues [Albanie et al. 2020; K. Prajwal et al. 2021] which also coincide with fingerspelling instances. This initial set of annotations are further extended and enhanced by a pseudolabeling step, also making use of subtitles (see Sec. 8.4).

A key challenge with training through automatic annotations is label noise. For example, the mouthing model [K. Prajwal et al. 2021] may spot a single word, while the fingerspelling contains multiple words, such as name-surname pairs. In some other cases, the mouthing model may fail, associating the wrong word in the subtitle with the fingerspelling segment. Also, some letters of the word may be skipped within fingerspelling as illustrated in Fig. 8.1. To account for this uncertainty, we implement a multiple-hypotheses version of the CTC loss [Graves

et al. 2006] (MH-CTC) where we consider all nouns from the subtitle, as well as bigrams and trigrams, as potential targets.

Our main contributions are: (i) Training a BSL fingerspelling detection and recognition model using only weak labels from subtitles, mouthing cues, and a small number (115) of manual fingerspelling exemplars; (ii) Employing multiple hypotheses from the subtitle words within the CTC loss to train with noisy labels; (iii) Demonstrating advantages of a pseudolabeling step incorporating the subtitle information; and (iv) Providing a large-scale manually annotated benchmark for evaluating BSL fingerspelling recognition, released for research purposes. Please check our website for more details: <https://www.robots.ox.ac.uk/~vgg/research/transpeller>.

8.2 Related work

Our work relates to four themes in the research literature: the broader topics of *sign language recognition* and *learning sign language from weak/noisy annotation*, and the more directly related literature on *spotting mouthings* and *fingerspelling recognition*.

Sign language recognition. Building on the pioneering 1988 work of Tamura and Kawasaki [Tamura and Kawasaki 1988], early approaches to automatic sign recognition made use of hand-crafted features for motion [M.-H. Yang et al. 2002] and hand shape [Fillbrandt et al. 2003; Vogler and Metaxas 2003]. To model the temporal nature of signing, there has also been a rich body of work exploring the use of Hidden Markov Models [Starner 1995; Vogler and Metaxas 2001; Fang et al. 2004; Cooper et al. 2011; Koller et al. 2016; Koller et al. 2017] and Transformers [N. C. Camgoz et al. 2020b; De Coster et al. 2020]. One notable trend in prior work is the transition towards employing deep spatiotemporal neural networks to provide robust features for recognition. In this regard, the I3D model of [Joao Carreira and Zisserman 2017] has seen widespread adoption, achieving strong recognition results on a range of benchmarks [Joze and Koller 2019; D. Li et al. 2019; Albanie et al. 2020; D. Li et al. 2020b]. In this work, we likewise build our approach on strong spatiotemporal video representations, adopting the Video-Swin Transformer [Z. Liu et al. 2022] as a backbone for our model.

Learning sign language from weak/noisy annotation. Given the paucity of large-scale annotated sign language datasets, a range of prior work has sought to leverage weakly aligned subtitled interpreter footage as a supervisory signal [Cooper and Bowden 2009; Buehler et al. 2009; Pfister et al. 2014; Momeni et al. 2020b] for sign spotting and recognition via apriori mining [Agrawal et al. 1993] and multiple instance learning [Dietterich et al. 1997]. Similarly to these works, we likewise aim to make use of subtitled signing footage. However, we do so in order to detect and recognize fingerspelling in a manner that allows for scalable training. To the best of our knowledge, this approach has not been considered in prior work.

Spotting mouthings in sign language videos. Signers often mouth the words that they sign (or fingerspell) [Rachel Sutton-Spence 2007]. The recent advancements in visual keyword spotting [K. Prajwal et al. 2021; Stafylakis and Tzimiropoulos 2018; Momeni et al. 2020a] have enabled the automatic curation of large-scale sign language datasets by spotting a set of query words using the mouthing cues and matching them with the corresponding sign segment. The state-of-the-art architecture for the visual KWS task is the Transpotter [K. Prajwal et al. 2021], which we use in Sec 8.4.1 to obtain our initial set of automatic fingerspelling annotations. Our fingerspelling architecture also partly takes inspiration from the Transpotter (and more broadly from prior works for text spotting that detect words and learn to read them via CTC [H. Li et al. 2017; Borisyuk et al. 2018]), wherein we process video features with a single Transformer encoder and then employ multiple heads to solve related tasks such as detection and classification with a $[CLS]$ token. In this work, we add also add a recognition head supervised by a novel loss function. We also show the benefits of our multi-stage training pipeline in training this model when we only have weak supervision.

Fingerspelling recognition and detection. Early work on automatic fingerspelling recognition explored the task of classification under fairly constrained settings, focusing on isolated signs and limited vocabularies (e.g. 20 words [Goh and Holden 2006], 82 words [Ricco and Tomasi 2009] and 100 words [Liwicki and Everingham 2009]). Kim et al. propose to consider instead a “lexicon-free” setting which they tackle with frame-level classifiers in combination with segmental CRFs on a newly introduced dataset, of 3,684 American Sign Language (ASL) fingerspelling instances [Taehwan Kim et al. 2017]. Moving towards more chal-

lenging data, the ChicagoFSWild (7304 fingerspelling sequences across 160 signers annotated by ASL students) [B. Shi et al. 2018] and ChicagoFSWild+ (55,232 fingerspelling sequences signed by 260 signers annotated by crowdsourcing) [B. Shi et al. 2019] datasets sourced from YouTube and Deaf social media target greater diversity and visual variation.

From a modeling perspective, Pugealt and Bowden employ random forests on depth and intensity images for real-time recognition of 24 fingerspelled letters [Pugeault and Bowden 2011]. Shi et al. demonstrate the benefits of using a signing hand detector for fingerspelling recognition without frame-level labels [B. Shi and Livescu 2017], motivating later work to attain this benefit automatically through visual attention without an explicit region detector [B. Shi et al. 2019]. Other work has explored the feasibility of using synthetic hand training data to fine-tune a CNN for isolated Irish Sign Language (ISL) fingerspelling recognition [Fowley and Ventresque 2021].

More closer to our work, several works have considered detecting the temporal location of fingerspelling in addition to recognition. This includes efforts to segment signing into sign types (classifiers, lexical signs, and fingerspelling) prior to recognition [Yanovich et al. 2016], as well as systems for fingerspelling detection supervised with segment boundaries [B. Shi et al. 2021]. The recently proposed FSS-Net learns joint embeddings to enable fingerspelling search within and across videos [B. Shi et al. 2022b]. In contrast, our work is weakly supervised with noisy, automatic annotations. It is weakly supervised in the sense that the model lacks access to ground truth fingerspelling boundaries at training, while the annotation is noisy in the sense that it is derived from subtitles from which the signing is produced as a translation, rather than a transcription.

8.3 Fingerspelling detection and recognition

In this section, we introduce Transpeller, our Transformer-based model to recognize and detect fingerspelling (Sec. 8.3.1). Our model is trained only on automatically curated data. In order to circumvent this label noise, we also propose a new loss function in Sec. 8.3.2.

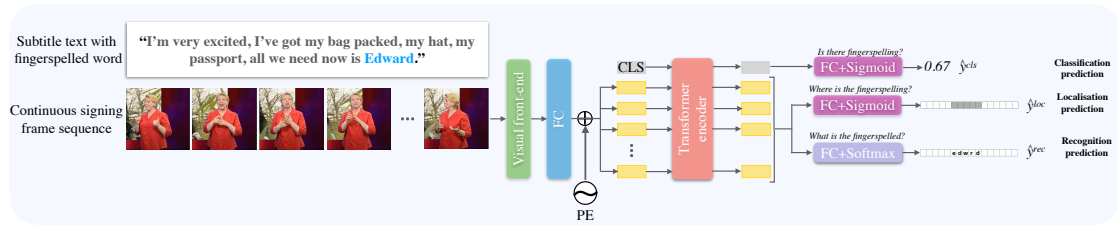


Figure 8.2: **Transpeller architecture.** Given a short clip of a signer, we extract features from a visual front-end pre-trained on the sign language recognition task. We project these features to a desired hidden dimension and add positional encodings before encoding these feature vectors using a transformer encoder. We use three heads on top of the transformer outputs to predict if the clip contains fingerspelling (classification prediction), and if so, where it is in the clip (localization prediction) and what it is (recognition prediction). The subtitle shown on the left is used for curating the training data and is shown here for illustrative purposes.

8.3.1 The Transpeller architecture

Our model ingests a video clip of a signer, encodes it with a Transformer encoder [Vaswani et al. 2017], and produces three outputs with each of its prediction heads: (i) a classification head that predicts **if** the given video clip contains fingerspelling, (ii) a localization head that produces per-frame probabilities indicating **where** the fingerspelling is in the clip, (iii) a recognition head that produces a sequence of letter probabilities indicating **what** is being fingerspelled. We illustrate this architecture in Figure 8.2.

Visual backbone. Our input is a sequence of RGB frames constituting a short clip of a signer. In order to extract the visual features, we follow prior sign-language works [Bull et al. 2021b; Momeni et al. 2022; Varol et al. 2021] and use a pre-trained sign classification model. The sign classification model used in prior works is an I3D model [Bull et al. 2021b; Momeni et al. 2022; Varol et al. 2021] pre-trained on Kinetics [Kay et al. 2017] and finetuned using a sign classification objective. We replace the I3D with a relatively modern Video-Swin-S [Z. Liu et al. 2022] architecture and finetune it for the sign classification task in a similar fashion. We pre-extract the features for all our videos and save them at a temporal stride of 4. The Transpeller operates on these feature vectors $V \in \mathbb{R}^{T \times d_f}$.

Transformer Encoder. Given a sequence of visual feature vectors $V \in \mathbb{R}^{T \times d_f}$, we first use two fully-connected layers (FC) to project the feature dimension to d , which is the hidden dimension of the transformer encoder. We add the temporal positional encodings and prepend this sequence with a learnable $[CLS]$ token em-

bedding (such as in BERT [Devlin et al. 2019] and ViT [Dosovitskiy et al. 2021]). We encode the temporal information of this sequence by passing it through a Transformer encoder consisting of N layers:

$$V_{enc} = \text{encoder}([CLS]; [FC(V) + PE_{1:T}]) \in \mathbb{R}^{(1+T) \times d}.$$

Prediction heads. The $[CLS]$ output feature vector $V_{enc(1)}$ serves as a aggregate representation for the entire input video clip. An MLP head for binary classification, f_c is attached to $V_{enc(1)}$ to predict the probability of a fingerspelling segment being present in the input video:

$$\hat{y}^{cls} = \sigma(f_c(V_{enc(1)})) \in \mathbb{R}^1,$$

where σ denotes a sigmoid activation. To localize the fingerspelling segment, we attach a second MLP head f_l that is shared across the encoded video feature time-steps $V_{enc(2:T+1)}$:

$$\hat{y}^{loc} = \sigma(f_l(V_{enc(2:T+1)})) \in \mathbb{R}^T.$$

The output y_t^{loc} at each feature time-step $t \in T$ indicates the probability of the time-step t being a part of a fingerspelling segment. In order to recognize what is being fingerspelt, a third MLP head f_r is attached in a similar fashion to f_l to predict \mathbb{C} letter probabilities at each time-step:

$$\hat{y}^{rec} = \text{softmax}(f_r(V_{enc(2:T+1)})) \in \mathbb{R}^{T \times \mathbb{C}}.$$

Loss functions. Given a training dataset \mathcal{D} consisting of video clips v , class labels y^{cls} , location labels y^{loc} and recognition labels (whenever present) y^{rec} , we define the following training objectives:

$$\mathcal{L}^{cls} = -\mathbb{E}_{(v, y^{cls}) \in \mathcal{D}} \text{BCE}(y^{cls}, \hat{y}^{cls}) \quad (8.1)$$

$$\mathcal{L}^{loc} = -\mathbb{E}_{(v, y^{cls}, y^{loc}) \in \mathcal{D}} y^{cls} \left[\frac{1}{T} \sum_{t=1}^T \text{BCE}(y_t^{loc}, \hat{y}_t^{loc}) \right] \quad (8.2)$$

$$\mathcal{L}^{rec} = -\mathbb{E}_{(v, y^{rec}) \in \mathcal{D}} \text{CTC}(y^{rec}, \hat{y}^{rec}) \quad (8.3)$$

where BCE stands for the binary cross-entropy loss and CTC stands for Connectionist Temporal Classification [Graves et al. 2006] loss. The labels y^{cls} are set to 1 when the given keyword occurs in the video and 0 otherwise; the frame labels

y^{loc} are set to 1 for the frames where the keyword is uttered and 0 otherwise. We optimize the total loss $\mathcal{L} = \mathcal{L}^{cls} + \mathcal{L}^{loc} + \lambda\mathcal{L}^{rec}$ where, $\lambda = 1$ if a given input video has a word label annotation, else, $\lambda = 0$ to only train the classification and localization heads.

8.3.2 Multiple Hypotheses (MH) CTC loss

As described earlier, our word labels for fingerspelling recognition are weak labels from subtitles and mouthing cues. The process for obtaining these labels is described in Sec. 8.4.1 & 8.4.2. Since this process is automatic, it introduces a degree of label noise. For example, the mouthing model can detect false positives; the detection boundaries can be erroneous, or multiple mouthings can be spotted for a single detection interval. We observed that our automatic detection pipeline is more accurate than our process of obtaining word labels. If we assume the fingerspelling detection is correct, then it is quite likely that the fingerspelled word is among one of the words (usually a noun) in the subtitle.

With this idea in mind, we design an improved CTC loss function, termed Multiple Hypotheses CTC (MH-CTC), that allows the model to “pick” the most correct word label from a set of possible word hypotheses. This set comprises a number of words, of which one is the correct word label for the fingerspelling sequence in the input video clip. For example, the set of word hypotheses could be the proper nouns in the subtitle corresponding to the input video clip. Given a list of word hypotheses \mathbb{H} for the input video clip v , our modified recognition loss \mathcal{L}^{rec} is:

$$\mathcal{L}^{rec} = -\mathbb{E}_{(v, \mathbb{H}) \in \mathcal{D}} \min_{\forall h \in \mathbb{H}} CTC(h, \hat{y}^{rec})$$

This corresponds to backpropagating the recognition loss for the word that achieves the minimum CTC loss among all the hypotheses. Since this allows the model to “choose” its own target, we found two strategies that help the model converge. Firstly, pretraining the model with CTC loss using Eqn 8.3.1 before using MH-CTC is essential. Secondly, when using MH-CTC, we found that randomly (with 50% chance) setting \mathbb{H} to be a single hypothesis containing the word found by our automatic annotation also prevents the model from diverging.

	Train			Val		
	#labels	vocab	avg. dur.	#labels	vocab	avg. dur.
Stage 1: Exemplar detections	149k	-	1.4s	3.0k	-	1.3s
w/ mouthing labels: nouns	59k	18k	1.8s	1.2k	0.8k	1.6s
w/ mouthing labels: pr. nouns	39k	14k	1.9s	0.9k	0.5k	1.7s
Stage 2: Transpeller detections	129k	-	1.9s	2.5k	-	1.8s
w/ mouthing labels: nouns	61k	19k	2.1s	1.3k	0.9k	1.9s
w/ mouthing labels: pr. nouns	41k	15k	2.2s	0.9k	0.5k	2.0s
w/ Transpeller labels	111k	32k	2.0s	2.2k	1.4k	1.8s

Table 8.1: **Two stages of automatic annotations.** Stage 1: We use exemplars to detect fingerspelling and mouthing cues to obtain letter labels. Stage 2: We obtain detections and letter labels using Transpeller pseudolabels.

8.4 Automatic annotations

Our model is completely trained with weak labels that are automatically curated. We perform two stages of automatic annotation. In the first stage (Sec. 8.4.1), we automatically annotate fingerspelling detections using exemplars and letter labels using mouthing cues. In the second stage (Sec. 8.4.2), we obtain detection and letter pseudolabels from the Transpeller model that has been trained on annotations from the first stage. The number and duration of these fingerspelling detections, as well as the number of detections associated with either a noun or proper noun letter label, is shown in Tab. 8.1.

8.4.1 Exemplar and mouthing annotations

Detections using exemplars. We manually annotate a small number E of single frame exemplars of fingerspelling amongst videos from different signers in the training set ($E = 115$). Using these exemplars, we search for frames in continuous signing with high feature similarity to these frames containing fingerspelling. This exemplar-based annotation technique is inspired by [Momeni et al. 2020b; Momeni et al. 2022]. We use features from [Momeni et al. 2022] and compute cosine similarity with fingerspelling features. This simple method provides approximate annotation of fingerspelling detections. We use this method for two reasons: firstly, to help our model learn fingerspelling temporal detection, and secondly, to select video segments containing fingerspelling for manual annotation. Technical details on the computation of the feature similarity can be found in the supplementary material.

Letter labels from mouthings. To obtain word annotations for the finger-spelled segments, we use mouthing cues, as fingerspelled words are often mouthed

simultaneously in interpreted data [Davis 1990]. In [Momeni et al. 2022], the authors use an improved Transpotter architecture [K. Prajwal et al. 2021] to query words from surrounding subtitles and localize corresponding mouthing cues. We consider all mouthing annotations from [Momeni et al. 2022] falling within the interval of the fingerspelling detections. As the fingerspelling detection boundaries are approximate, and the automatic mouthing annotations are not always accurate, these annotations are noisy. Tab. 8.2 shows that almost all fingerspelling annotations refer to nouns, and most refer to proper nouns. Thus, restricting mouthing annotations to nouns or proper nouns reduces noise.

8.4.2 Transpeller annotations

Improving detections with Transpeller pseudolabels. The model described in Sec. 8.3.1 outputs a per-frame localization score, predicting the presence of fingerspelling. After training this model, we can improve upon the exemplar-based detections using pseudolabels. Given localisation scores s_1, \dots, s_N for a window of N frames, we consider that the window contains fingerspelling if $\max(s_1, \dots, s_N) > t_1$. We consider that a sub-interval $[i, j]$ ($1 \leq i \leq j \leq N$) of this window contains fingerspelling if $s_i, \dots, s_j \geq t_2$, where $t_2 < t_1$. To smooth the localization scores, we take the moving maximum score amongst K consecutive scores. We let $t_1 = 0.7$, $t_2 = 0.3$ and $K = 5$.

Improving letter labels with Transpeller pseudolabels. Using pseudolabels from the model in Sec. 8.3.1, we can also improve automatic letter label annotations. After decoding the CTC outputs using beam search, we can then compute a proximity score with words from neighboring subtitles, i.e. $\text{dist}(w_1, w_2)$, where w_1 is a subtitle word and w_2 is the output of beam search decoding. The proximity score is a variant of the Levenshtein edit distance, but where deletions are not heavily penalized. This is because words such as ‘Sarah Jane’ can be reasonably fingerspelled as ‘SJ’. Details of this proximity score are provided in the supplementary material. We can use this proximity score to find the subtitle word most likely to be fingerspelled.

8.4.3 Multi-stage training

We perform a multi-stage training strategy where we start by training on exemplar-based annotations (lines 2 and 3 of Tab. 8.1) using the vanilla CTC loss to supervise

the recognition head. Upon convergence, we finetune this model further using MH-CTC, this time additionally considering proper nouns in neighboring subtitles as the hypotheses.

Using the above pre-trained model, we now extract the Transpeller annotations as detailed in Sec. 8.4.2. Using these new annotations, we repeat the process: we start with the vanilla CTC loss and then finetune this model further using MH-CTC. Given that we have pseudo-labels from the Stage 1 Transpeller model, we can restrict the MH-CTC search space. The hypotheses now only contain nouns and proper nouns from neighboring subtitles with at least one letter in common with the CTC decoded outputs from the Transpeller model of Stage 1.

8.5 Experiments

8.5.1 BOBSL Fingerspelling benchmark

We collect and release the first benchmark for evaluating fingerspelling in British Sign Language. The test set annotations are collected by adapting the VIA Whole-Sign Verification Tool [Dutta and Zisserman 2019] to the task of fingerspelling recognition. Given proposed temporal windows around the automatic exemplar and mouthing annotations, annotators mark whether there is any fingerspelling in the signing window and type out the exact letters which are fingerspelled. We use a temporal window of 2.1s before to 4s after the midpoint of the automatically detected fingerspelling instance. Since the fingerspelled letters could be a subset of the actual full word (e.g. SH for SARAH), we also obtain the corresponding full word annotations. Descriptive statistics on the test set annotations are in Tab. 8.2.

Evaluation criteria. We measure the fingerspelling recognition performance using the Character Error Rate (CER), which provides a normalized count of the substitution/deletion/insertion errors in the predicted letter sequence when compared to the ground-truth sequence. We report two CERs for the two different ground-truth annotations we have for each clip: (i) $\text{CER}_{\text{fspell}}$ - ground-truth is the actual fingerspelled letters which, as mentioned before, may only be part of a word, and (ii) CER_{full} - ground-truth is the full word annotation to which the fingerspelling refers to. Given that our model(s) are trained only with automatic weakly-supervised full-word annotations (Sec. 8.4), these two different CER scores can help us see if the model learns to pick up on the fingerspelled letters, rather

#labels	(full word) vocab.	%nouns	%pr. nouns	%full	avg. % missing
4923	3442	96%	74%	22%	34%

Table 8.2: **Statistics on test set annotations.** Most fingerspellings refer to proper nouns or nouns. Around 22% of fingerspelling instances contain all letters of the encoded word, but on average 34% of the letters of a word are omitted during fingerspelling.

Annotations	#Recogn. ex.	CER _{fspell}	CER _{full}
Exemplars + Mouthings: proper nouns	39k	58.5	62.1
Exemplars + Mouthings: nouns	59k	58.6	62.9
finetuned with MH-CTC	59k (avg. 4 hyp.)	57.6	64.3

Table 8.3: **Stage 1: Transpeller model with exemplar + mouthing supervision.** All rows use 149k fingerspelling detections for training. When assigning a word label for these detections from the subtitle, we choose to look at nouns, especially proper nouns, as they are more likely to be fingerspelt. We obtain the best results when using MH-CTC loss.

than only relying on the mouthing cues or memorizing the full word annotations.

Implementation details. We use a batch size of 32 and an initial learning rate of $5e^{-5}$, which is reduced to $1e^{-5}$ after the validation loss does not improve for 3 epochs. At test time, we decode with a beam width of 30. More implementation details are provided in the supplementary material.

8.5.2 Results

We now evaluate different variations at each stage of the Transpeller recognition pipeline.

Using exemplar detections and mouthing cue letter labels with CTC loss. Our initial set of annotations from the exemplar detections gives us 149k fingerspelling instances, out of which a fraction of them can be associated with

Annotations	#Recogn. ex.	CER _{fspell}	CER _{full}
Transpeller detect. + Mouthings	61k	57.5	63.1
Transpeller detect. + Char. labels	111k	55.4	63.0
finetuned with MH-CTC	111k + (avg. 9 hyp.)	53.3	60.1

Table 8.4: **Stage 2: Transpeller pseudolabels.** All rows use 129k fingerspelling detections for training. Using both the refined detections and word annotations from the Stage 1 model gives a clear reduction in the CER. Further, using MH-CTC gives a 2.1 CER boost. Overall, our Stage 2 achieves a final best CER of 53.3 which is 4.3 CER better than the best model of Stage 1, thus validating the impact of our multi-stage training pipeline.

word labels using mouthing cues. In Tab. 8.3, we show how our performance depends on our choice of recognition annotations. Restricting our automatic word label annotations to proper nouns gives us a cleaner training set, as they are most likely to be fingerspelt. However, this comes at the expense of having very few training samples. In row 2 of Table 8.3, we find that we can tolerate a bit of label noise and expand to all nouns. We finetune the best CTC-based model from the above using our MH-CTC loss, which further results in an improvement of 1.0 $\text{CER}_{\text{fspell}}$. This is our best model with the initial set of annotations.

Transpeller detections. We now expand our training data, by extracting pseudolabels (Sec. 8.4.2) from the best model from the previous stage. We train on these new annotations and report our results in Tab 8.4. When we use the refined detections but still use the mouthing cues for assigning word labels, we obtain a $\text{CER}_{\text{fspell}}$ of 57.5, a similar result to the corresponding model (57.6) from Stage 1.

Transpeller letter labels. The error rates drop further when we also improve the recognition annotations. We do so by using a variant of the edit distance to match the predictions of the Stage 1 Transpeller to a word in a neighboring subtitle. These results can be seen in row 2 of Tab. 8.4.

Finetuning with MH-CTC. Finally, when finetuning further with our MH-CTC loss, our final Stage 2 model gives a $\text{CER}_{\text{fspell}}$ of 53.3 which is 4.3 points better than the best model (57.6) of Stage 1. It is evident from both these tables that both MH-CTC and our multi-stage training with pseudolabelling improve our recognition performance.

Fingerspelling recognition vs. spotting mouthings. An interesting line of thought is: if we have access to the subtitles at test time, can we use a mouthing model to accurately predict words instead of doing fingerspelling recognition? To judge this, we restrict the test set to instances where there is a noun or a proper noun mouthing annotation and compute the CER between the mouthing annotation and the ground truth annotation, and we get 55.6. On this same subset, our best model obtains 53.3 CER, demonstrating that it in fact performs better than a mouthing method with access to the subtitle text.

Error analysis of the best model on the test set. In Tab. 8.5, we show a few examples of our predictions and the corresponding ground-truth sequence. We

can see our model makes reasonable errors for most examples, where it confuses between letters that are visually quite similar. In Fig. 8.4, we show how the CER varies based on the length of the ground-truth character sequence. It is evident that the model struggles the most with very short sequences (one or two letters). This is expected because it has only been supervised with full word annotations during training and has never been trained to predict one or two letters in isolation.

Lookup-based correction at inference-time. We explore the possibility of correcting the errors in the model’s outputs at test-time with the help of a pre-determined list of words, e.g. “atlanic” to “atlantic”. We first curate a list of nouns present in the subtitles of the BOBSL train set and use edit distance to match the model’s predictions to the closest noun in our list. If the edit distance is below a set threshold, i.e., a very close match, we replace the predicted character sequence with the matched word from our list. However, we found that this increases the $\text{CER}_{\text{fspell}}$ to 57.0 and CER_{full} to 61.1. Such a lookup-based correction method leads to several false matches because (i) it is done with no context of the surrounding words, (ii) not all the letters of the word are fingerspelled, (iii) fingerspelled words in the test set can also be novel and unseen.

8.5.3 Architecture ablations

Importance of joint recognition and detection. As described in Sec 8.3.1, our model contains three prediction heads for classification, localization, and recognition. All three heads are essential and are used to obtain the Stage 2 annotations as described in 8.4.2. We conduct an experiment to also demonstrate that it is beneficial to train these heads jointly. We find that joint training leads to a better recognition performance (55.4 $\text{CER}_{\text{fspell}}$) than training without the localization head (56.3 $\text{CER}_{\text{fspell}}$) or without the localization and classification heads (56.2 $\text{CER}_{\text{fspell}}$).

Sequence-to-Sequence vs. CTC-based models. We also compare our CTC-based recognition head with a sequence-to-sequence (seq2seq) encoder-decoder architecture supervised with a cross-entropy loss. We use the standard Transformer-Base [Vaswani et al. 2017] model, which contains a Transformer encoder similar to Transpeller and a 6-layer auto-regressive Transformer decoder. We compare with the CTC model trained on the Stage 2 annotations. The seq2seq network

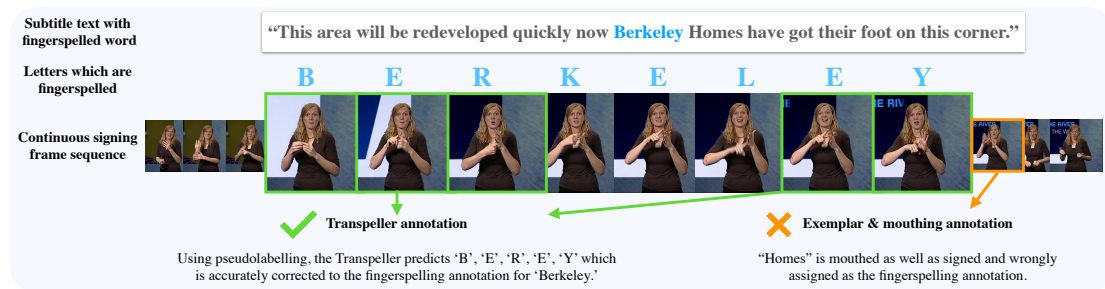


Figure 8.3: **Benefit of Transpeller annotations:** Here, we are given a finger-spelling clip of “BERKELEY”. The initial annotation with exemplars + mouthing cues assigns an incorrect word label because the spotted mouthing of the word “Homes” is temporally close to the fingerspelling location, resulting in a false word assignment. In the second annotation stage, we correct this using the Transpeller’s predicted characters “BEREY” that are matched to the subtitle word “BERKELEY”.

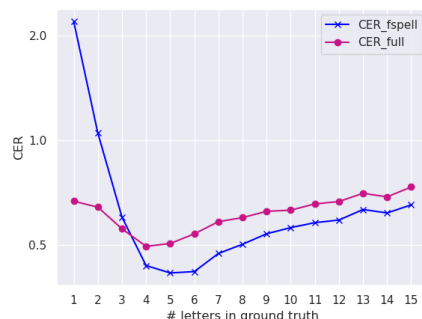


Figure 8.4: **Variation of CER vs the number of letters in the ground-truth.** Transpeller struggles to correctly predict very short fingerspelling segments, which are usually partial fingerspellings (e.g. MJ for Mary Jane). This happens because the model is only supervised with full-length words.

Ground Truth	Prediction	CER
chloride	churide	25
nurembrg	turmug	50
ivory	ener	80
elind	elinc	20
bnmm	ben	75
clove	clune	40
semnoa	samol	50

Figure 8.5: **Qualitative examples.** Model predictions on our manually verified test set. The CERs are shown for reference. We see that one of the most common error sources is confusion between letters that are visually similar: (a, e, i), (d, c), (l, n, m, v, t), (o, u).

performs worse ($57.4 \text{ CER}_{\text{fspell}}$) than the CTC model ($55.4 \text{ CER}_{\text{fspell}}$). This is expected because only 22% of the test samples contain no missing letters, i.e. full words, so conditioning on past letters can lead to errors in future letter predictions. However, conditioning on past letters can be very helpful if we want to actually estimate the full word and not determine the exact letters that are fingerspelled. This is indeed validated by the fact that the seq2seq model performs much better on CER_{full} , achieving 55.2 compared to the CTC model’s 63.0. We also note that due to the sequential decoding in the seq2seq model, it is $\approx 20\times$ slower in terms of run-time speed than the CTC model, which decodes all the characters in parallel.

8.6 Conclusion

We presented a new BSL fingerspelling recognition benchmark and our Transpeller model designed to jointly detect and recognize fingerspelled letters in continuous sign language video. Our training data is largely constructed automatically, exploiting English subtitles and mouthing cues. The evaluation data is manually curated, and we achieve promising recognition results (Tab. 8.5). However, we note some limitations. First, there remains room for improvement in the accuracy of the model to further reduce the character error rate. Second, our training and evaluation of the Transpeller is limited to the use of interpreted data and therefore is not necessarily representative of more natural, conversational signing. Addressing these limitations would be a valuable future research direction.

Acknowledgements. This work was supported by the Oxford-Google DeepMind Graduate Scholarship, EPSRC grant ExTol, a Royal Society Research Professorship and the ANR project CorVis ANR-21-CE23-0003-01. HB would like to thank Annelies Braffort and Michèle Gouiffès for the support.

Chapter 9

A Tale of Two Languages: Large-Vocabulary Continuous Sign Language Recognition from Spoken Language Supervision

The paper is a technical report presenting ongoing works, written in 2023.

A Tale of Two Languages: Large-Vocabulary Continuous *Sign Language* Recognition from *Spoken Language* Supervision

Charles Raude^{1,2*} Prajwal KR^{2*} Liliane Momeni^{2*}
Hannah Bull¹ Samuel Albanie³ Andrew Zisserman²
Gül Varol¹

¹ LIGM, École des Ponts, Univ Gustave Eiffel, CNRS, France

² Visual Geometry Group, University of Oxford, UK

³ Department of Engineering, University of Cambridge, UK

Abstract

In this work, our goals are two fold: large-vocabulary continuous sign language recognition (CSLR), and sign language retrieval. To this end, we introduce a multi-task Transformer model, CSLR², that is able to ingest a signing sequence and output in a joint embedding space between signed language and spoken language text. To enable CSLR evaluation in the large-vocabulary setting, we introduce new dataset annotations that have been manually collected. These provide continuous sign-level annotations for six hours of test videos, and will be made publicly available. We demonstrate that by a careful choice of loss functions, training the model for both the CSLR and retrieval tasks is mutually beneficial in terms of performance – retrieval improves CSLR performance by providing context, while CSLR improves retrieval with more fine-grained supervision. We further show the benefits of leveraging weak and noisy supervision from large-vocabulary datasets such as BOBSL, namely sign-level pseudo-labels, and English subtitles. Our model significantly outperforms the previous state of the art on both tasks.

*Equal contribution.

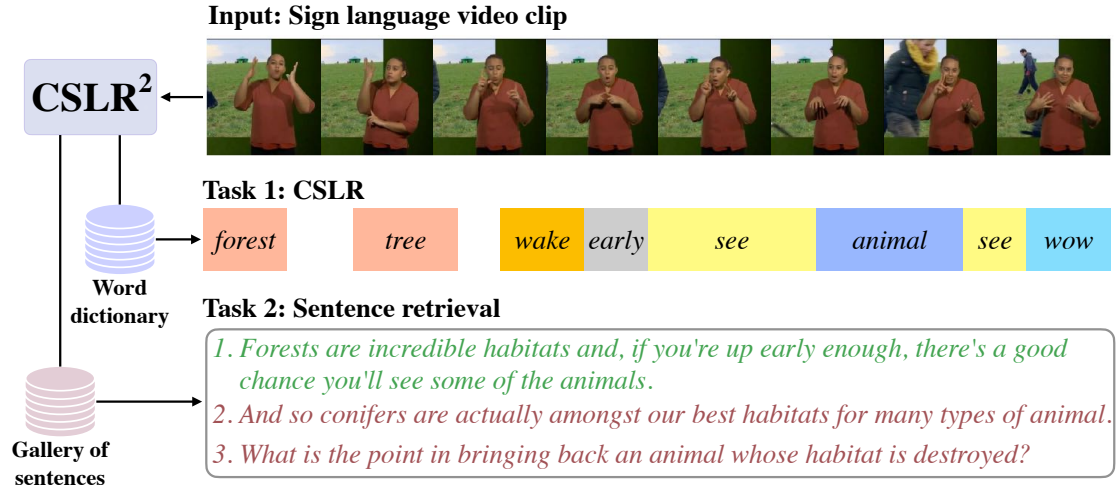


Figure 9.1: **CSLR² model**: We illustrate our multi-task model that performs both **CSLR** and sentence **R**etrieval, thanks to its joint embedding space between signed language and spoken language text.

9.1 Introduction

Recognising *continuous* and *large-vocabulary* sign language is a vital step towards enabling real-world technologies that enhance communication and accessibility for the deaf or hard of hearing. With the availability of data that depicts continuous signing from a large vocabulary of signs [Albanie et al. 2021b; N. Camgoz et al. 2021; Duarte et al. 2021], the computer vision field has recently gained momentum towards this direction, building on previous research that had largely focused on restricted settings such as recognising single signs in isolation [Joze and Koller 2019; D. Li et al. 2019] or signs covering relatively small vocabularies [Koller et al. 2015b; H. Zhou et al. 2020a].

Our goal in this paper is two-fold: first, to enable *large-vocabulary* continuous sign language recognition (CSLR) – providing time aligned and dense word predictions for each sign within a signing sequence. This is an essential first step towards translation, as English sentence-level annotations have been shown to be difficult to use directly as targets for sign language translation [Varol et al. 2021; Albanie et al. 2021b; N. Camgoz et al. 2021]. Our second goal is *sentence retrieval*, i.e. given a signing video, to retrieve the most similar sentence text or vice versa (see Fig. 9.1). This is important as indexing sign language videos to make them searchable has been highlighted as a useful application for deaf or hard of hearing [Bragg et al. 2019]. Also, video to subtitle retrieval can be seen as a proxy for translation – it is reminiscent of the pre-deep learning style of machine translation where sentences

were broken down into phrases and translation proceeded by a lookup of paired phrases in the two languages [Koehn et al. 2003].

There are several challenges to achieving these goals, primarily due to the lack of suitable data for training and evaluation. For CSLR, ideally, *each* individual sign within a continuous video should be associated to a symbolic category. However, current training supervision sources are restricted by their *weak* or *sparse* nature. For instance, in the largest dataset BOBSL [Albanie et al. 2021b], the available annotations are either (i) at sentence-level, weakly associating the entire signing video to an English sentence, rather than breaking the video into individual sign-word correspondences, or (ii) at sign-level, but sparse with gaps in the temporal timeline (despite the densification efforts in [Momeni et al. 2022] to scale up the number and vocabulary of annotations). Also, there is no evaluation benchmark with continuous ground truth sign annotations for the BOBSL dataset, so it is not possible to assess and compare the performance of large-vocabulary CSLR algorithms at scale.

In this paper, we introduce a simple Transformer encoder model [Vaswani et al. 2017] that ingests a signing video sequence and outputs tokens in a joint embedding space between signed and spoken¹ languages. The output space enables both the CSLR and sentence retrieval tasks. The Transformer architecture outputs CSLR predictions by leveraging temporal context, and a retrieval embedding through pooling. The joint embedding language space may also help to overcome yet another challenge of sign language recognition: polysemy where the same word may correspond to several sign variants, and conversely the same sign may correspond to several different words.

We train our model on both tasks by leveraging noisy supervision from the large-scale BOBSL dataset. Specifically, we use an individual sign predictor to generate continuous pseudo-labels (for training CSLR) and available weakly-aligned sentence-level annotations (for training sentence retrieval). We show that training for both tasks is mutually beneficial – in that including CSLR improves the retrieval performance, and including retrieval improves the CSLR performance. To enable CSLR evaluation, we manually collect new sign-level annotations that are continuous on the timeline. Since we focus on the large-vocabulary setting, we

¹We refer to the written form of spoken language, not the speech audio.

collect annotations on the BOBSL test set. We hope our new CSLR benchmark will facilitate further exploration in this field.

In summary, our contributions are the following: (i) We demonstrate the advantages of a single model, CSLR², that is trained jointly for both CSLR and sign language sentence retrieval with weak supervision. (ii) Thanks to our joint embedding space between spoken and signed languages, we are the first to perform sign recognition via video-to-text retrieval. (iii) We build a benchmark of substantial size for evaluating large-vocabulary CSLR by collecting continuous sign-level annotations for 6 hours of video. (iv) We significantly outperform strong baselines on our new CSLR and retrieval benchmarks, and carefully ablate each of our components. We will make our code and data available for research.

9.2 Related Work

We briefly discuss relevant works that operate on (i) continuous sign language video streams, (ii) sign language retrieval, and (iii) CSLR benchmarks.

Ingesting continuous sign language video streams. In the recent years, the community has started to move beyond isolated sign language recognition (ISLR) [Joze and Koller 2019; D. Li et al. 2019], which only seeks to assign a category (typically also expressed as a word) to a short video segment trimmed around a single sign without context. Besides CSLR, several tasks that require ingesting a continuous video stream exist. These include sign spotting [Albanie et al. 2020; Momeni et al. 2020b; Varol et al. 2021; Momeni et al. 2022], sign tokenization [Renz et al. 2021a; Renz et al. 2021b], translation [N. C. Camgoz

		segmented	#sentences	hours	vocab.	#glosses	source
train	PHOENIX-2014 [Koller et al. 2015b]	✗	6K	11	1K	65K	TV
	CSL-Daily [H. Zhou et al. 2020a]	✗	18K	21	2K	134K	lab
	BOBSL [Albanie et al. 2021b]	✗	993K	1220	72K*	5.5M*	TV
test	PHOENIX-2014 [Koller et al. 2015b]	✗	629	0.99	500	7089	TV
	CSL-Daily [H. Zhou et al. 2020a]	✗	1176	1.41	1345	9002	lab
	BOBSL CSLR-TEST	✓	4451	5.93	4462	30172	TV

Table 9.1: **Recent CSLR training and evaluation sets:** Our manually-curated BOBSL CSLR-TEST set is larger in number of annotated signs and vocabulary, compared to other CSLR test sets from the literature. In addition, it also comes with sign segmentation annotations. *Note that the BOBSL training has different vocabulary sets of varying sizes: 72K words spanned by subtitles and 25K words spanned by 5.5M automatic annotations generated in [Momeni et al. 2022].

et al. 2018; N. C. Camgoz et al. 2020b; N. C. Camgoz et al. 2020a; Benjia Zhou et al. 2023; A. Yin et al. 2023], subtitle alignment [Bull et al. 2021a], subtitle segmentation [Bull et al. 2020], text-based retrieval [Duarte et al. 2022; Y. Cheng et al. 2023] and fingerspelling detection [K R Prajwal et al. 2022b]. Our work is related to some of these works in that they also operate on a large-vocabulary setting [Albanie et al. 2020; Momeni et al. 2020b; Varol et al. 2021; Bull et al. 2021a; Momeni et al. 2022]; however, they do not tackle CSLR, mainly due to lack of continuous sign annotations. While [Duarte et al. 2022] addresses retrieval, their method is not suitable for CSLR – our work differs in that we perform both tasks jointly.

State-of-the-art CSLR methods have so far focused on PHOENIX-2014 [Koller et al. 2015b] or CSL-Daily [H. Zhou et al. 2020a] benchmarks, where the performances are saturated. These methods typically consider a fully-supervised setting, and train with RNN-based [J. Huang et al. 2018b; Cui et al. 2019; Huaiwen Zhang et al. 2023], or Transformer-based [N. C. Camgoz et al. 2020b] models. Due to lack of sign segmentation annotation (i.e., the start and end times of signs are unknown), many works use the CTC loss [N. C. Camgoz et al. 2020b; K. L. Cheng et al. 2020; Jiao et al. 2023; F. Wei and Y. Chen 2023; Zuo and Mak 2022]. Our work differs from these previous works on several fronts. We consider a weakly-supervised setting, where the training videos are *not* annotated for CSLR purposes, but are accompanied with weakly-aligned spoken language translation sentences. We also study the benefits of joint training with CSLR and retrieval objectives. In a similar spirit, the works of [N. C. Camgoz et al. 2020b; Zuo and Mak 2022] jointly train CSLR with sentence-level objectives (translation in [N. C. Camgoz et al. 2020b], margin loss for gloss-sequence text retrieval in [Zuo and Mak 2022]), but in significantly different settings (e.g., $8\times$ smaller vocabulary, and with manually-annotated CSLR labels for training).

Sign language retrieval. Early works focused on query-by-example [Athitsos et al. 2010; S. Zhang and Bo Zhang 2010], where the goal is to retrieve individual sign instances for given sign examples. The release of continuous sign video datasets, like BOBSL [Albanie et al. 2021b], How2Sign [Duarte et al. 2021], and CSL-Daily [H. Zhou et al. 2020a] with (approximately) aligned spoken language subtitles, has shifted the interest towards spoken language to sign language re-

trieval (and vice-versa). The first work in this direction is the recent method of [Duarte et al. 2022], which focuses on improving the video backbone that is subsequently used for a simple retrieval model using a contrastive margin loss. CiCo [Y. Cheng et al. 2023] also focuses on improving video representations, specifically, by designing a domain-aware backbone. In contrast, our main emphasis is on (i) the use of weakly-supervised data, and (ii) the joint training with CSLR.

Our work naturally derives lessons from the large number of efforts in the parent task of sign language retrieval, i.e., video-text retrieval [Bain et al. 2021b; Gabeur et al. 2020; S. Liu et al. 2021; Y. Liu et al. 2019; H. Luo et al. 2022; C. Sun et al. 2019; Y. Yu et al. 2018]. Works such as CoCa [Jiahui Yu et al. 2022] and JSFusion [Y. Yu et al. 2018] have shown that jointly training with a cross-modal retrieval objective can help in other tasks such as captioning and question-answering. Our approach is in the same vein as these works: we show that jointly training for retrieval and CSLR improves performance for both tasks.

CSLR benchmarks. Early works with continuous signing videos provided very small vocabularies in the order of several hundreds (104 signs in Purdue RVL-SLLL [Wilbur and Kak 2006] and BOSTON104 [Dreuw et al. 2008] ASL datasets, 178 in CCSL [J. Huang et al. 2018b], 310 in GSL [Adaloglou et al. 2020], 455 in the SIGNUM DGS dataset [von Agris et al. 2008], and 524 in the KETI KSL dataset [Ko et al. 2019b]). BSL Corpus [Schembri et al. 2013] represents a large-vocabulary collection; however, it is mainly curated for linguistics studies, and has not been used for CSLR.

Relatively large collections made it possible to train CSLR methods based on neural networks (see Tab. 9.1). Most widely used RWTH-PHOENIX-Weather 2014 [Koller et al. 2015b] dataset contains around 11 hours of videos sourced from weather forecast on TV. CSL-Daily [H. Zhou et al. 2020a] provides 20K videos with gloss and translation annotations from daily life topics, covering a 2K sign vocabulary, and 23 hours of lab recordings. In Tab. 9.1, we provide several statistics to compare against our new CSLR benchmark, mainly on their evaluation sets (bottom). While being larger, we also provide sign segmentation annotations.

Recently released large-vocabulary continuous datasets (such as BOBSL [Albanie et al. 2021b], How2Sign [Duarte et al. 2021], Content4All [N. Camgoz et al. 2021],

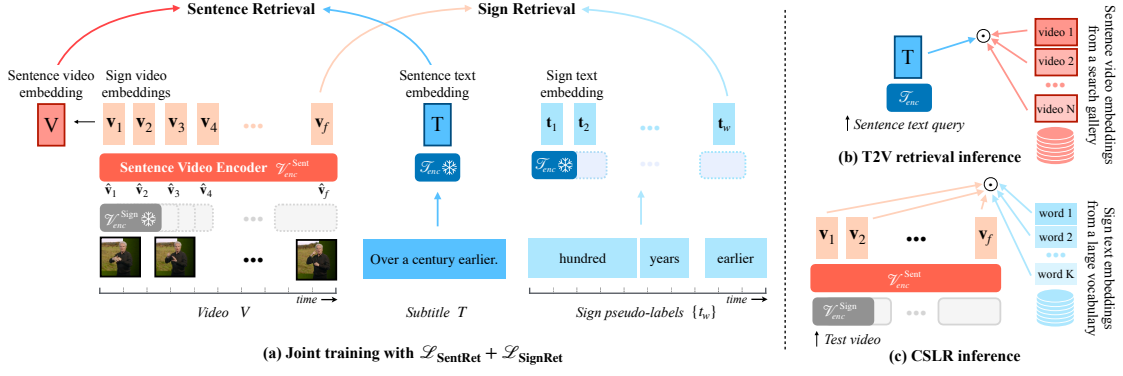


Figure 9.2: **Method overview:** (a) We show a simplified view for our model architecture which consists of both video and text streams. On the video side, features are extracted from a signing video clip V by running \mathcal{V}_{enc}^{Sign} in a sliding window fashion and passed through a Transformer model \mathcal{V}_{enc}^{Sent} . A video embedding V and sign video embeddings $\{\mathbf{v}_f\}$ are subsequently extracted. On the text side, we input an English subtitle sentence T and sign pseudo-labels $\{t_w\}$ to the text encoder \mathcal{T}_{enc} and obtain sentence and sign text embeddings ($T, \{t_w\}$), respectively. While we illustrate only one triplet data point $(V, T, \{t_w\})$, in practice, we operate on a minibatch of triplets, and employ two contrastive losses to jointly train on sentence retrieval $\mathcal{L}_{SentRet}$ and sign retrieval $\mathcal{L}_{SignRet}$. (b) For text-to-video retrieval inference, we simply extract a sentence text embedding given a text query, and rank the sentence video embeddings corresponding to gallery videos according to their cosine similarities. (c) For CSLR inference, each sign video embedding is matched to the top-ranked word from a large vocabulary of size 8K. A post-processing strategy is applied on frame-level predictions to produce final outputs. For visibility, we omit linear layers which project embeddings into the learnt joint-space. See Sec. 9.3.1 for a detailed description of the architecture and inference procedure.

and OpenASL [B. Shi et al. 2022a]) do not provide sign-level gloss annotations due to the prohibitive costs of densely labeling within the open-vocabulary setting. In this work, we leverage an isolated sign recognition model to generate continuous pseudo-labels for training CSLR, and available weakly-aligned sentence-level supervision for retrieval.

9.3 Joint Space for Signed and Spoken Languages

We start by describing the model design that goes from raw sign language video pixels to a *joint embedding space* with spoken language text (Sec. 9.3.1). We then present the losses of our joint training framework with sentence-level and sign-level objectives (Sec. 9.3.2). Next, we detail our supervision which consists of (noisy) sign-level pseudo-labels and weakly-aligned subtitles (Sec. 9.3.3). Finally, we provide model implementation details (Sec. 9.3.4).

9.3.1 Model overview and inference

Our model, shown in Fig. 9.2, consists of three main components: (i) a sign video encoder $\mathcal{V}_{enc}^{\text{Sign}}$ based on a pretrained Video-Swin [Z. Liu et al. 2022], (ii) a sentence video encoder $\mathcal{V}_{enc}^{\text{Sent}}$ based on a randomly initialised Transformer encoder, and (iii) a text encoder \mathcal{T}_{enc} based on a pretrained T5 model [Raffel et al. 2020]. Given raw RGB video frame pixels V for a signing sentence, we obtain a sequence of isolated sign video embeddings from $\mathcal{V}_{enc}^{\text{Sign}}$ as $\{\hat{\mathbf{v}}_f\} = \mathcal{V}_{enc}^{\text{Sign}}(V)$. In practice, such a sequence is obtained by feeding 16 consecutive frames to the sign video encoder in a sliding window fashion, with a stride of 2 frames. These are then fed through the sentence video encoder $\mathcal{V}_{enc}^{\text{Sent}}$ to have *context-aware* sign video embeddings $\{\mathbf{v}_f\}$, as well as a single sentence video embedding \mathbf{V} , denoted $(\{\mathbf{v}_f\}, \mathbf{V}) = \mathcal{V}_{enc}^{\text{Sent}}(\{\hat{\mathbf{v}}_f\})$. Similarly, for the text side, we embed the sentence T into $\mathbf{T} = \mathcal{T}_{enc}(T)$. Additionally, we define sign-level text embeddings for each sign in the sentence as $\{\mathbf{t}_w\}_{w=1}^W = \{\mathcal{T}_{enc}(t_w)\}_{w=1}^W$, where W is the number of signs in the sentence. In practice, we get these embeddings by independently feeding the word(s) corresponding to each sign to the text encoder.

CSLR inference. Sign-level recognition predictions are obtained by using the sequence of sign video embeddings in \mathbf{v}_f that lies in the same space as spoken language. To associate each feature frame f to a word (or phrase), we perform nearest neighbour classification by using a large text gallery of sign category names, as illustrated in Fig. 9.2c. In our experiments, we observe superior performance of such retrieval-based classification over the more traditional cross-entropy classification [M. Wang et al. 2021], with the advantage that it is potentially not limited to a closed vocabulary. In order to go from per-feature classification to continuous sign predictions, we perform a post-processing strategy detailed in Sec. 9.3.3. We note that this same post-processing strategy is used for obtaining our sign-level pseudo-labels for training.

Retrieval inference. For sign-video-to-text (V2T) retrieval, the video sentence embedding \mathbf{V} is matched to a gallery of text sentence embeddings, ranking text sentences by their cosine similarities. Symmetrically, text-to-sign-video (T2V) retrieval is performed in a similar manner, as shown in Fig. 9.2b.

9.3.2 Training with sentence- and sign-level losses

We train the Transformer-based model, that operates on sentence-level sign language videos, to perform two tasks, namely, CSLR and sign language Retrieval (CSLR²). As illustrated in Fig. 9.2a, we employ two retrieval losses: (i) a sentence-level objective, supervised with weakly-aligned subtitles, and (ii) a sign-level objective, supervised with pseudo-labels obtained from a strong ISLR model [K R Prajwal et al. 2022b]. We next formulate each objective individually before introducing our joint framework that leverages both sentence-level and sign-level information.

Sentence-level objective: sign language sentence retrieval (SentRet).

We explore the task of retrieval as a means to obtain supervision signal from the subtitles. Following the success of vision-language models building a cross-modal embedding between images and text [Radford et al. 2021; J. Li et al. 2022], we employ a standard contrastive loss, and map sign language videos to spoken language text space.

Sign language sentence retrieval is made of two symmetric tasks, that is, V2T and T2V retrievals. For the former, given a query signing video V , the goal is to rank a gallery of text samples (here subtitles) such that the content of V matches the content in the highest ranked texts. Symmetrically, in the latter, given a text query T , the goal is to rank a gallery of signing videos.

Formally, given a dataset $\mathcal{D} = \{(V_i, T_i)\}_{i=1}^N$ of video-subtitle pairs, the goal is to learn two encoders ϕ_V, ϕ_T mapping each signing video V and subtitle T into a joint embedding space. In the following, $\mathbf{V}_i = \phi_V(V_i)$ and $\mathbf{T}_i = \phi_T(T_i)$ denote the video and text embeddings, respectively. The encoders are trained using a recently proposed Hard-Negative variant of InfoNCE [van den Oord et al. 2018], HN-NCE [Radenovic et al. 2023], that re-weights the contribution of each element in the computation of the contrastive loss. Let $\{(\mathbf{V}_i, \mathbf{T}_i)\}_{i=1}^B$ be a batch of encoded video-subtitle pairs and $S_{ij} = \mathbf{V}_i^T \mathbf{T}_j$ be the similarity between the pair (i, j) . For the sake of visibility, we only detail the equations for V2T:

$$\mathcal{L}_{\text{HN-NCE, V2T}} = -\frac{1}{B} \sum_{i=1}^B \log \frac{e^{S_{ii}/\tau}}{\alpha \cdot e^{S_{ii}/\tau} + \sum_{j \neq i} w_{ij}^{V2T} \cdot e^{S_{ij}/\tau}}, \quad (9.1)$$

with w_{ij}^{V2T} weights defined as

$$w_{ij}^{V2T} = \frac{(B-1) \cdot e^{\beta S_{ij}/\tau}}{\sum_{k \neq i} e^{\beta S_{ik}/\tau}}, \quad (9.2)$$

where the temperature $\tau > 0$, $\alpha \in (0, 1]$, and $\beta \geq 0$ are hyperparameters. By training to maximise the similarity between correct pairs of video and subtitle embeddings, while minimising the similarity between negative pairs, HN-NCE serves as a proxy for the retrieval by ranking that we perform at inference.

Sign-level objective: sign classification via sign retrieval (SignRet). Given a continuous signing video, the goal of CSLR is to recognise a sequence of individual signs. The continuous video is encoded into a sequence of context-aware sign video embeddings $\{\mathbf{v}_f\}_{f=1}^F$, with F the number of video frames. Again, since these embeddings are in the same joint space as the text embeddings, a contrastive loss can be used as a proxy for sign retrieval.

Similarly to the sentence-level retrieval, we use the HN-NCE contrastive formulation defined in Eq.(9.1) for the sign retrieval (SignRet) loss. However, instead of the *full sentence* video-text embedding pair (\mathbf{V}, \mathbf{T}) , we map *individual* sign video-word embedding pairs (\mathbf{v}, \mathbf{t}) (see Fig. 9.2c).

Overall loss. Our model is trained jointly using a weighted sum of the two retrieval terms:

$$\mathcal{L} = \lambda_{\text{SentRet}} \mathcal{L}_{\text{SentRet}} + \lambda_{\text{SignRet}} \mathcal{L}_{\text{SignRet}}$$

with $\mathcal{L}_{\text{SentRet}}, \mathcal{L}_{\text{SignRet}}$, i.e. two contrastive losses for sentence and sign retrieval, respectively. The training details including the batch size, learning rate, and other hyperparameters can be found in the supplementary materials.

9.3.3 Sources of supervision

Leveraging weak and noisy text labels for the CSLR and retrieval training constitutes one of the key contributions of this work. Next, we present our two sources of text supervision, namely, sign-level pseudo-labels and sentence-level weakly-aligned subtitles.

Sign-level pseudo-labels. We start with (V, T) video-subtitle pairs that do not contain sign-level annotations. In order to obtain sign-level supervision to train for CSLR, we perform sign-level pseudo-labelling. Specifically, we apply an ISLR model in a sliding window fashion with a stride of 2 frames, and perform post-processing of sign predictions as an attempt to reduce noise. Our post-processing strategy consists of 3 steps: (i) we first combine confidence scores of synonym categories for the Top-5 predictions from the ISLR model (using the synonym list defined in [Momeni et al. 2022]); (ii) we then filter out low confidence predictions (below a threshold value of $\theta = 0.6$); (iii) finally, we remove non-consecutive predictions – since each sign spans several video frames, we expect repetitions from the ISLR model (we keep predictions with at least $m = 6$ repetitions).

In practice, for each subtitle, we define a sentence-level video (on average 3.4 seconds) by trimming the episode-level video (~ 1 h duration) using the subtitle timestamps. The sentence-level video is further broken down into frame-sign correspondences based on pseudo-label timestamps. The sign-level loss is then only computed on frames associated to a pseudo-label after post-processing.

Weakly-aligned subtitles. The source of our large-scale video-subtitle pairs is from sign language interpreted TV shows, where the timings of the accompanying subtitles correspond to the audio track, but not necessarily to signing [Albanie et al. 2021b]. For better sign-video-to-text alignments, we use automatic signing-aligned subtitles from [Bull et al. 2021a] (described in [Albanie et al. 2021b]) to train our models. We restrict our training to subtitles spanning 1-20 seconds, resulting in 689K video-subtitle training pairs.

9.3.4 Implementation details

In the following, we detail each component of our model.

Sign video encoder ($\mathcal{V}_{enc}^{\text{Sign}}$). Similar to [K R Prajwal et al. 2022b], our sign video features are obtained by training a Video-Swin model [Z. Liu et al. 2022], on ISLR. The network ingests a short video clip (16 frames, < 1 second) and outputs a single vector $\hat{\mathbf{v}} \in \mathbb{R}^d$ ($d = 768$), followed by a classification head to recognise isolated signs. Specifically, we finetune the Video-Swin-Tiny architecture, pretrained on Kinetics-400 [Joao Carreira and Zisserman 2017], using automatic annotations

released in [Momeni et al. 2022]. These annotations provide individual sign labels along with timestamps of where they occur in the video. Note that the annotations have been *automatically* obtained with the help of subtitles (by exploiting cues such as mouthing), and can thus be noisy. Once trained for ISLR, we freeze the parameters of this relatively expensive backbone, and extract isolated sign video embeddings $\hat{\mathbf{v}}$ in a sliding window manner with a stride of 2 frames. Note we use RGB-based embeddings, instead of body keypoint estimates, due to their more competitive performance in the large-vocabulary setting, where sign differences are subtle and nuanced [Albanie et al. 2021b].

Sentence video encoder ($\mathcal{V}_{enc}^{\text{Sent}}$). We adopt a Transformer encoder architecture, similar to BERT [Devlin et al. 2019], with 6 encoder layers, 8 attention heads and 768 hidden dimensionality. It ingests the sign sentence video as isolated sign video embeddings $\{\hat{\mathbf{v}}_f\}$ and outputs context-aware sign video embeddings $\{\mathbf{v}_f\}$. We obtain a single sentence video embedding by simply max-pooling over the temporal dimension, i.e. $\mathbf{V} = \text{MaxPool}_f(\{\mathbf{v}_f\}_{f=1}^F)$, with F video features, and experimentally validate this choice. We restrict our training to video clips shorter than 20 seconds. The parameters of the Transformer encoder are learnt using the sentence retrieval loss $\mathcal{L}_{\text{SentRet}}$ between sentence text T and video V embeddings, and the sign retrieval loss $\mathcal{L}_{\text{SignRet}}$ between sign text embeddings \mathbf{t}_w and the corresponding sign video embeddings \mathbf{v}_f , as described in Sec. 9.3.2.

Text encoder (\mathcal{T}_{enc}). We use the encoder part of a pre-trained T5 [Raffel et al. 2020] (t5-large), and keep its weights frozen. Note we do not use its decoder. The output text embeddings have dimensionality 1024.

Projection heads. We additionally learn projection layers, mainly to reduce the joint embedding dimensionality to 256 before contrastive loss computations. Specifically, we have a total of 4 projection heads: two for reducing the *text* dimensionality ($1024 \rightarrow 256$) with a separate projection for sign categories and sentences, two for reducing the *video* embedding dimensionality ($768 \rightarrow 256$) separately for sign and sentence video embeddings.

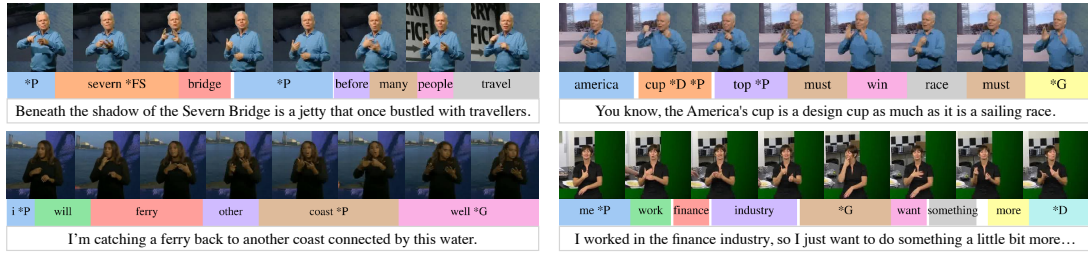


Figure 9.3: **Annotation examples from the CSLR-Test dataset:** As well as assigning the English word(s) corresponding to a sign (i.e. ‘gloss’), the annotators indicate the type of sign when appropriate. For example, ‘*P’ for pointing, ‘*FS’ for fingerspelling, ‘*G’ for gesture sign.

9.4 A New CSLR Evaluation Benchmark

In this section, we describe the new continuous sign annotations that we collected for evaluating CSLR. We first describe what the CSLR-TEST is, and then how it was annotated.

CSLR-Test. The continuous annotations are provided for a subset of the SENT-TEST partition of BOBSL [Albanie et al. 2021b]. SENT-TEST is a 31 hour subset of the BOBSL test set where the BSL signing sequences have been manually aligned temporally with their corresponding English subtitle sentences.

The CSLR-TEST annotations consist of a time aligned sequence of sign ‘glosses’², where each sign is annotated with its temporal interval, the type of the sign, and its word equivalent if that exists. In addition to lexical signs (i.e. signs that have an English word equivalent), a wide range of sign types such as fingerspelling, pointing, depicting or no-signing are annotated. These are marked with special characters such as *FS and *P for fingerspelling and pointing, respectively.

Note, that there exists no universally accepted writing system for sign languages today [Filhol 2020], though attempts have been made with descriptive languages such as HamNoSys [Hanke 2004] and SignWriting [Sutton 1990]. Also, careful glossing that is linguistically consistent (e.g., enumerating each sign variant [Schembri et al. 2017]) is a tedious process which hinders scaling up. For these reasons, we make a compromise when annotating for CSLR and use English words for the glosses, assigning ‘any’ reasonable English word for a sign segment, but prioritising words in the surrounding subtitle. For example, if the ‘natural’ English word for the

²We abuse the gloss terminology, despite our sign-level annotations *not* being careful linguistic glosses, but rather free-form sign-level translations.

sign is ‘laugh’ but a synonym word such as ‘giggle’ is in the subtitle, then the gloss would be ‘giggle’ (with ‘laugh’ also provided as a more general translation). However, one should keep in mind that associating words to signs is a lossy and error-prone process in any case.

Fig. 9.3 shows example ground truth gloss annotations for CSLR-TEST. In total, we curate these continuous labels for 5.93 hours of video, comprising 30,172 individual signs from a vocabulary of approximately 4,462 glosses. The CSLR-TEST annotations evenly cover all 35 episodes in SENT-TEST. Additional statistics for the dataset are given in Tab. 9.1. We note that we also annotate a small subset of the BOBSL training and validation subtitle-aligned splits. All annotations will be publicly released.

Dataset annotation. The annotation procedure uses a web based annotation tool that is built from the VIA video annotation software [Dutta and Zisserman 2019]. Annotators are provided with a video sequence with 10 time aligned subtitle sentences. Annotators enter free-form text for each sign token, taking into account the context by watching the full video around a given subtitle. The subtitle is also displayed on the video and the annotators are encouraged to prioritise assigning words that appear in the corresponding subtitle. The annotators additionally assign sign types when appropriate (see Fig. 9.3) and temporally align each gloss to the duration of the sign.

We facilitate faster annotation iterations by incorporating several strategies. We adopt a semi-automatic labeling technique where we initialise the sign boundaries by using an automatic sign segmentation method [Renz et al. 2021a]. Annotators are thus given an initial set of sign intervals, and are instructed to refine these sign boundaries (or add/remove sign intervals) if necessary. We further show a dropdown menu for each sign and prioritise at the top of the list words from the corresponding subtitle. Two native BSL users worked on this task for over one year. Further details on the annotation procedure are provided in the supplementary material.

9.5 Experiments

In this section, we first present evaluation protocols used in our experiments (Sec. 9.5.1) and describe baselines (Sec. 9.5.2). Next, we provide ablations to assess the contribution of important components (Sec. 9.5.3). We then report CSLR and retrieval performance, comparing to the state of the art (Sec. 9.5.4), and illustrate qualitative results (Sec. 9.5.5).

9.5.1 Data and evaluation protocol

BOBSL [Albanie et al. 2021b] consists of about 1500 hours of video data accompanied with approximately-aligned subtitles. A 200-hour subset is reserved for testing. We reuse existing manually-aligned validation and test sets (**Sent-Val** [Albanie et al. 2021b], **Sent-Test** [Albanie et al. 2021b]) for our sentence retrieval evaluation (20,870 and 1,973 aligned sentences respectively). We perform our retrieval ablations on the validation set, and report the final model on both evaluation sets. For CSLR evaluation, we use our manually annotated test set (**CSLR-Test**) as described in Sec. 9.4, which corresponds to 4950 unseen test subtitles.

For retrieval evaluation, we report both T2V and V2T performances using standard retrieval metrics, namely **recall at rank k ($R@k$)** for $k \in \{1, 5\}$. For CSLR evaluation, given a video sequence defined by **CSLR-TEST**, we compute our predicted gloss sequence (after post-processing the raw per-frame outputs with the optimal θ and m heuristics – see Sec. 9.3.3 for post-processing strategy) and compare against its corresponding ground-truth gloss sequence using several metrics. We note that we filter out sign types and signs that are not associated to lexical words from the ground-truth sequence. In addition, if several annotation words are associated to one sign (e.g. ‘giggle’ and ‘laugh’), predictions are considered to be correct if one of these words is predicted correctly.

As in other CSLR benchmarks [Koller et al. 2015b; H. Zhou et al. 2020a], we report word error rate (**WER**) as our main performance measure. We also monitor **mIoU** (mean intersection over union) between predicted and ground truth sequences’ words, without considering any temporal aspect. In all metrics, similar to [Momeni et al. 2022], we do not penalise output words if they are synonyms.

In order to assess the model’s ability to correctly predict sign segments at the right temporal location, we also report the F1 score. We define a segment to be correctly predicted if (i) the predicted sign matches, up to synonyms, the ground-truth gloss, and (ii) the IoU between the predicted segment’s boundaries and the ground truth gloss segment’s boundaries is higher than a given threshold. We compute the F1 score as the harmonic mean of precision and recall, based on this definition of correct segment detection. We report the F1 score at different thresholds values, namely, $F1@\{0.1, 0.25, 0.5\}$

9.5.2 Baselines

Subtitle-based automatic annotations CSLR baseline. The first baseline for CSLR is obtained using spotting methods that search for signs corresponding to words in the spoken language subtitles. The initial set of sparse sign annotations along with timestamps was released with the BOBSL dataset [Albanie et al. 2021b], using sign spotting methods from [Albanie et al. 2020; Momeni et al. 2020b; Varol et al. 2021], followed by denser spottings in [Momeni et al. 2022]. We evaluate these existing sequences of spottings, in particular the ones corresponding to our CSLR-TEST subtitles. In practice, we filter these automatic annotations using the same sets of thresholds as in [Albanie et al. 2021b; Momeni et al. 2022], respectively (see supplementary material for details). Note that these spottings make use of the weakly-aligned subtitles (and cannot go beyond words in the subtitles), and therefore cannot be used as a true CSLR method. They are also point annotations, without precise temporal extent, thus we omit F1 scores.

ISLR baselines for CSLR. This set of baselines uses ISLR models in a sliding window fashion to obtain continuous frame-level predictions. We aggregate the sliding window outputs by performing the post-processing strategy described in Sec. 9.3.3 (for optimal θ and m parameters, tuned on unseen manually annotated CSLR sequences from the validation set). We use the I3D [Joao Carreira and Zisserman 2017] and Video-Swin [Z. Liu et al. 2022] models trained from spotting annotations of [Albanie et al. 2021b; Momeni et al. 2022]. We build on prior works for these baselines: we use the I3D weights released in [Albanie et al. 2021b] and train a Video-Swin-Tiny model in a similar fashion as in [K R Prajwal et al. 2022b].

InfoNCE retrieval baseline. Our baseline for sentence retrieval is the standard

SentRet	Sign-level loss	WER ↓	mIOU ↑	F1@{0.1, 0.25, 0.5} ↑		
ISLR BASELINE		71.3	30.0	51.8	50.5	42.4
✗	CTC	74.1	28.4	-	-	-
✗	CE	73.5	29.5	46.5	45.8	39.0
✗	SignRet	70.8	30.0	50.0	49.4	43.1
✓	CE	71.8	27.6	49.0	48.5	42.9
✓	SignRet	65.3	35.3	54.0	53.2	47.1

Table 9.2: **CSLR ablations on CSLR-Test:** Only using CTC, cross entropy (CE), or SignRet does not perform well, remaining below the ISLR baseline. We observe best results when incorporating joint sentence retrieval training.

contrastive training [van den Oord et al. 2018] employed by many strong vision-language models [Radford et al. 2021; J. Li et al. 2022]. We train this vanilla model, without the CSLR objective, on the automatically-aligned subtitles from [Bull et al. 2021a]. Sentence embeddings, obtained by feeding subtitle text into \mathcal{T}_{enc} , are compared against a learnable `cls` token on the video side which pools the video embeddings as done in [Dosovitskiy et al. 2021; Radford et al. 2021].

9.5.3 Ablation study

CSLR components. In Tab. 9.2, we experiment with the choice of training objectives for CSLR performance. In particular, we train with the standard CTC or cross-entropy (CE) losses, as well as our sign retrieval (SignRet) loss alone. Without an additional sentence retrieval loss, i.e. if we only optimise for a sign-level objective, we observe that the performance is worse than the strong ISLR baseline (i.e. with Video-Swin) for CTC and CE, and comparable for SignRet. In the final two rows, we observe clear gains by combining the SentRet loss with either (i) our SignRet loss or (ii) the standard CE loss. While the joint training with the CE loss brings a performance boost, from 73.5 to 71.8 WER, it does not surpass the competitive ISLR baseline. Our model CSLR², which jointly trains sentence and sign retrieval, brings a major improvement by reducing the WER by 6 points, from 71.3 to 65.3.

Retrieval components. In Tab. 9.3, we compare design choices for the retrieval task: (i) choice of the contrastive loss function – InfoNCE vs. HN-NCE; (ii) choice of pooling the temporal features – using a `cls` token [Devlin et al. 2019; Dosovitskiy et al. 2021] vs. max-pooling. We observe a clear boost in all metrics by using HN-NCE with our weakly-aligned data, which gives more weight to the hard-negatives

Pool.	SentRet loss	Sign-level loss	T2V			V2T		
			R@1 ↑	R@5 ↑	R@10 ↑	R@1 ↑	R@5 ↑	R@10 ↑
c1s	InfoNCE	X	38.9	62.1	69.1	39.2	61.0	68.1
c1s	HN-NCE	X	48.9	68.3	73.9	46.5	67.2	72.8
max	InfoNCE	X	43.4	64.9	71.0	42.7	64.8	70.7
max	HN-NCE	X	50.5	69.5	75.1	49.7	69.7	74.7
max	HN-NCE	CE	50.0	69.1	74.4	48.7	68.7	74.3
max	HN-NCE	SignRet	51.7	69.9	75.4	50.2	69.1	74.7

Table 9.3: **Retrieval ablations on Sent-Val:** We experiment with the choice of the contrastive sentence retrieval (SentRet) loss (standard InfoNCE vs. HN-NCE), the visual encoder pooling (c1s vs max), the addition and choice of sign-level losses (cross entropy CE vs. contrastive sign retrieval SignRet). The last two rows correspond to the joint models evaluated for CSLR in Tab. 9.2.

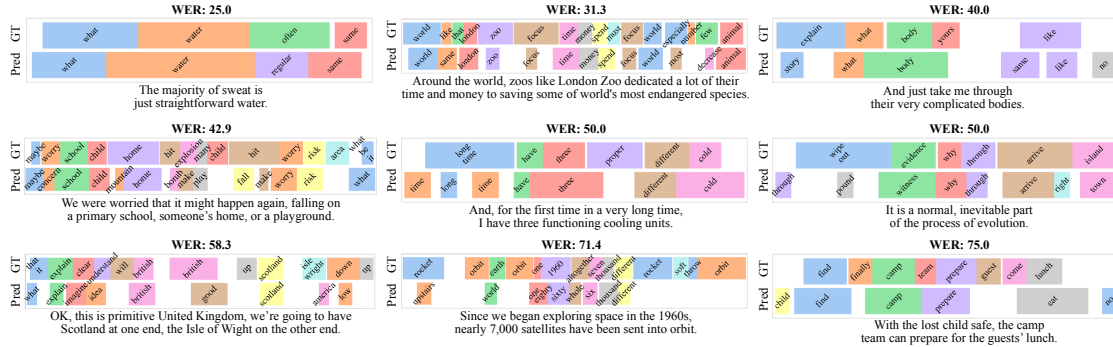


Figure 9.4: **Qualitative CSLR results:** We compare our model’s predictions (Pred) against the ground truth (GT), providing examples from several error ranges (sorted by WER). The subtitles displayed below each example are not used by the model. While we observe that our model correctly predicts a large portion of signs, handling both English synonyms as well as sign language polysemy (two visually similar signs with different meanings) makes the CSLR task challenging. Synonyms are depicted with the same color coding, e.g. ‘earth’ and ‘world’ in 3rd row, middle.

when computing the contrastive loss: there is a minimum improvement of +7 R@1 for T2V comparisons. We further observe that max-pooling the visual Transformer encoder outputs, instead of using a learnable c1s token, consistently gives better results. The joint training of retrieval and CSLR, with SignRet, also improves retrieval performance: R@1 for T2V increases from 50.5 to 51.7. More importantly, joint training enables a single, strong model which can perform both tasks.

9.5.4 Comparison to the state of the art

We compare to the current state-of-the-art approaches, both for large-vocabulary CSLR and sentence retrieval, in Tab. 9.4.

CSLR performance. First, in terms of the baselines, it can be seen from Tab. 9.4, that our post-processing strategy significantly strengthens the original ISLR I3D-

CSLR		CSLR-TEST				
Model		WER ↓	mIOU ↑	F1@{0.1, 0.25, 0.5} ↑		
Subtitle-based spotting [Albanie et al. 2021b]		93.6	7.2	-	-	-
Subtitle-based spotting [Momeni et al. 2022]		81.6	19.8	-	-	-
ISLR I3D-2K [Albanie et al. 2021b]		453.0	8.7	11.5	9.5	6.2
ISLR I3D-2K [Albanie et al. 2021b] †		82.4	17.9	46.5	44.5	35.2
ISLR I3D-8K [Momeni et al. 2022] †		74.6	27.0	49.5	47.9	39.1
ISLR Swin-8K [K R Prajwal et al. 2022b] †		71.3	30.0	51.8	50.4	42.4
CSLR ² (OURS) †		65.3	35.3	54.0	53.2	47.1

Retrieval	SENT-VAL (2K)		SENT-TEST (20K)			
	T2V	V2T	T2V		V2T	
Model	R@1 ↑	R@1 ↑	R@1 ↑	R@5 ↑	R@1 ↑	R@5 ↑
InfoNCE	38.9	39.2	19.5	35.1	18.9	33.8
CSLR ² (OURS)	51.7	50.2	29.4	45.2	28.1	44.9

Table 9.4: **Comparison to the state of the art:** Our joint model significantly outperforms both CSLR (top) and retrieval (bottom) baselines. For CSLR, note that the automatic spotting annotations [Albanie et al. 2021b; Momeni et al. 2022] have *access to the subtitles* at inference (unlike our fully automated approach). We also compare to raw ISLR outputs from sign classification models from [Albanie et al. 2021b; Momeni et al. 2022; K R Prajwal et al. 2022b] with various backbones (I3D or Swin) and with various vocabularies (2K or 8K categories). Our optimal filtering and post-processing strategy at inference is denoted with † (see. 9.3.3). We note that for the ISLR I3D-2K baseline without †, we still remove consecutive repetitions.

2K [Albanie et al. 2021b] performance by removing significant noise (with/without †) – we reduce the WER by more than a factor of 5 (453.0 vs 82.4). Also, our post-processing strategy combined with the 8K vocabulary ISLR models, delivers models of higher performance (by more than 6 WER) than all subtitle-based spottings methods, even though the ISLR models do not have access to the subtitles. Second, our joint model, CSLR², outperforms all CSLR baselines by a significant margin on all metrics. Indeed, CSLR² surpasses the best subtitle spotting method by 16.3 WER and the strongest ISLR baseline by 6 WER. Please refer to the supplementary material for a breakdown of performance based on different sign types.

Retrieval performance. As Tab. 9.4 shows, our joint CSLR² model outperforms a standard InfoNCE [van den Oord et al. 2018] baseline for retrieval on all reported metrics, with gains in R@1 for both T2V and V2T of almost 10 points. On the more challenging SENT-TEST gallery of 20k video-subtitle pairs, our CSLR² model achieves a Top-5 accuracy of 45.2% for T2V retrieval. We observe that for cases where the target sentence is not the Top-1, the top-retrieved results usually exhibit

semantic similarities with the correct sentence, with multiple common words (see the qualitative examples in the supp. mat.).

9.5.5 Qualitative analysis

In Fig. 9.4, we show several qualitative examples of our CSLR predictions (Pred rows) against the corresponding ground truth (GT rows) on CSLR-TEST. Note that we display the corresponding ground truth subtitles below each example to give context to the reader, but they are not given as input to the model. We illustrate examples from several error ranges, sorting them by WER per sample (reported at the top). These timelines show that our model is able to predict a large proportion of the annotations, in the correct order, with approximate sign segmentation. For instance, even though the bottom-right example in Fig. 9.4 has a high error rate of 75 WER, our predictions correctly identify 4 out of the 8 ground truth words, and catch the meaning of the sentence.

However, we also observe several challenges: (i) our model has difficulty predicting several words for a single sign, as the 8K training vocabulary of pseudo-labels primarily comprises of individual words (e.g. the phrase ‘long time’ is associated to a single sign in the 2nd row - middle - but our model predicts two separate signs ‘long’ and ‘time’, leading to an extra insertion) (ii) our model performance is sensitive to the synonyms list, which must be carefully constructed to not unfairly penalise predictions (e.g. in the top left example, ‘regular’ is counted as a substitution since ‘often’ is not present in ‘regular’s synonyms list) (iii) our model still struggles with visually similar signs in BSL which correspond to different English words (e.g. ‘upstairs’ and ‘rocket’ signs are visually similar, both signed by pointing upwards, 3rd row - middle); (iv) finally, our model is more likely to fail in recognising names of places and people as these are often fingerspelled in BSL and may therefore not be in our 8K sign vocabulary of pseudo-labels (e.g. ‘isle wright’ is predicted as ‘america’ in the bottom left). Future directions include addressing such limitations.

9.6 Conclusion

In this work, we demonstrate that jointly training for CSLR and sign language retrieval is mutually beneficial. We collect a large-vocabulary CSLR benchmark, consisting of 6 hours of continuous sign-level annotations. By leveraging weak supervision, we train a single model which outperforms strong baselines on both our new CSLR benchmark and existing retrieval benchmarks. While our approach shows substantial improvements, future work includes increasing the vocabulary size beyond 8K and modeling non-lexical signing classes such as pointing and gesture-based signs.

Societal impact. The two sign language understanding tasks we address can have positive implications by bridging the gap between spoken and sign languages. These tasks can enable more seamless communication, content creation and consumption by breaking down the language barriers that are prevalent today. At the same time, the ability to automatically search a large volume of signing videos can lead to risks such as surveillance of signers. We believe that the positives outweigh the negatives.

Acknowledgements. This work was granted access to the HPC resources of IDRIS made by GENCI. The authors would like to acknowledge the ANR project CorVis ANR-21-CE23-0003-01.

Part IV

Enhancing verb representations

Chapter 10

Verbs in Action: Improving verb understanding in video-language models

The paper has been accepted for publication at the International Conference on Computer Vision (ICCV), 2023.

Verbs in Action: Improving verb understanding in video-language models

Liliane Momeni¹ Mathilde Caron² Arsha Nagrani²

Andrew Zisserman¹ Cordelia Schmid²

¹ Visual Geometry Group, University of Oxford, UK

² Google Research

Abstract

Understanding verbs is crucial to modelling how people and objects interact with each other and the environment through space and time. Recently, state-of-the-art video-language models based on CLIP have been shown to have limited verb understanding and to rely extensively on nouns, restricting their performance in real-world video applications that require action and temporal understanding. In this work, we improve verb understanding for CLIP-based video-language models by proposing a new Verb-Focused Contrastive (VFC) framework. This consists of two main components: (1) leveraging pretrained large language models (LLMs) to create hard negatives for cross-modal contrastive learning, together with a calibration strategy to balance the occurrence of concepts in positive and negative pairs; and (2) enforcing a fine-grained, verb phrase alignment loss. Our method achieves state-of-the-art results for *zero-shot* performance on three downstream tasks that focus on verb understanding: video-text matching, video question-answering and video classification. To the best of our knowledge, this is the first work which proposes a method to alleviate the verb understanding problem, and does not simply highlight it. Code and model available at [scenic/projects/verbs_in_action](https://github.com/oxford-vision-lab/scenic/projects/verbs_in_action).

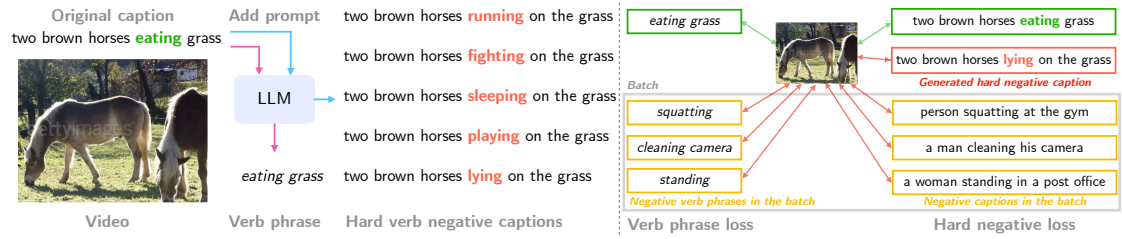


Figure 10.1: **Verb-Focused Contrastive (VFC) learning:** (Left): Given a video and its corresponding caption, we leverage a Large Language Model (LLM) to output (1) hard negative captions, where only the verb has been changed while keeping the remaining context, and (2) verb phrases which succinctly describe the action in the video. (Right): To encourage better verb reasoning, we subsequently enforce (1) a *calibrated* hard negative loss, using our generated hard negative captions and other captions in the batch, and (2) a fine-grained, verb phrase loss. We show that VFC improves verb understanding of video-language models compared to the standard contrastive loss.

10.1 Introduction

Large-scale visual-language models (VLMs) such as CLIP [Radford et al. 2021] have shown strong performance on multiple video-language tasks such as text-to-video retrieval [H. Luo et al. 2022], video question-answering, and open-set action recognition [Z. Lin et al. 2022]. These models perform surprisingly well on these tasks in a zero-shot setting, despite being trained only on image-language pairs (with no access to temporal data), even outperforming strong video-specific models [Bain et al. 2021b; S. Yan et al. 2022].

A recently highlighted and well-documented problem with such models, however, is their strong *noun* or *object* bias, as evidenced by their lower performance in distinguishing between *verbs* in natural language descriptions [Hendricks and Nematzadeh 2021; Park et al. 2022; Yuksekgonul et al. 2023]. This was first studied in images alone by the SVO-Probes benchmark [Hendricks and Nematzadeh 2021], which shows that *image*-language models struggle to distinguish between different verbs, and often rely on the nouns instead. This problem persists with *video*-language models that inherit these VLMs, even after they are fine-tuned on video-text datasets [J. Xu et al. 2016; Rohrbach et al. 2017]. For example, Park et al. [Park et al. 2022] similarly propose evaluation sets with hard verb negatives, and show that CLIP-based models, even when fine-tuned on video datasets, have difficulties discriminating verbs in a multi-choice setting where the context remains unchanged. Yuksekgonul et al. [Yuksekgonul et al. 2023] further highlight limita-

tions of vision-language models at understanding attribute, relationship, and order information. This deficiency in verb understanding limits the model’s applicability for real-world tasks. Verbs encapsulate how people and objects interact with each other, and the environment, via actions in space and time.

We believe that there are two probable causes for this deficiency, even after fine-tuning on video-text data: (i) existing visual-text datasets have a strong bias towards single-frame concepts such as *objects* and *backgrounds* as well as *static* actions [Sevilla-Lara et al. 2021; Buch et al. 2022; Lei et al. 2022]. Models are hence less incentivized to understand dynamics and temporal actions [Sevilla-Lara et al. 2021], biasing them towards noun understanding; and (ii) the limitations of the cross-modal contrastive pretraining objective used by most current vision-language models [Yuksekgonul et al. 2023]. In contrastive learning, the model is trained to distinguish correct video-caption pairs from incorrect ones. Since it is unlikely that existing datasets contain many examples with captions of *similar* context but *different* verbs, the task can be solved by taking little verb information into account. This relates to shortcut learning in deep neural networks [Geirhos et al. 2020].

In an attempt to mitigate this problem, we propose a novel training framework for tackling the task of verb understanding in vision-language models. Our framework, called **Verb-Focused Contrastive** pretraining (VFC), consists of two novel technical modifications to the contrastive learning framework. We first introduce a method to automatically generate negative sentences for training where only the verb has changed, keeping the context the same. This is done using LLMs [Raffel et al. 2020; et al 2022], in an automatic and scalable manner. Note that we *generate* hard negative captions, unlike works that simply mine hard negatives from an existing paired dataset [Radenovic et al. 2023], or change the order of words [Yuksekgonul et al. 2023]. For example, given the caption ‘*two brown horses eating grass*’, we generate the negative caption ‘*two brown horses running on the grass*’ (see Fig. 10.1). While this improves performance on some downstream tasks, we find that introducing concepts simply in *negative* examples can also lead to an imbalance in the contrastive objective, favouring certain concepts in the feature space. To solve this, we propose a simple but effective *calibration strategy* to balance the occurrence of verbs in both positive and negative captions.

Secondly, inspired by recent works on *grounding* concepts in vision-language learning [Kamath et al. 2021; M. Cao et al. 2022], we also introduce a verb phrase loss that explicitly isolates the verb from a caption for more focused training. For example, we extract the verb phrase ‘*eating grass*’ from the caption ‘*two brown horses eating grass*’ (see Fig. 10.1). We find that this helps particularly for zero-shot performance on downstream tasks that do not use long sentences in their evaluation [Ghadiyaram et al. 2019]. Verb phrases are also extracted from sentences using LLMs.

We then train a CLIP-based model [H. Luo et al. 2022] on a video-language dataset with this novel training framework. We show that a *single model* trained in this way transfers well to diverse downstream tasks that focus particularly on verb understanding, including three video benchmarks (multiple choice video-text matching on MSR-VTT [J. Xu et al. 2016], video question answering on Next-QA [Xiao et al. 2021], action recognition on Kinetics [Joao Carreira and Zisserman 2017]) and one image benchmark (SVO-probes [Hendricks and Nematzadeh 2021]), achieving state-of-the-art performance compared to previous works in *zero-shot* settings (and often with fine-tuning as well); while maintaining performance on noun-focused settings. On Kinetics, we also introduce a verb split of the data which specifically highlights classes that are challenging to distinguish without fine-grained verb understanding (‘*brushing hair*’ vs ‘*curling hair*’) and show that our model particularly improves performance on this split.

10.2 Related works

LLMs for video-text tasks. LLMs have been used for various vision applications, for example to initialise vision-text models [Seo et al. 2022; Jun Chen et al. 2022; Z. Luo et al. 2022]. Recent works further use frozen LLMs via prompting for tackling vision-language tasks [Alayrac et al. 2022; Tsimpoukelli et al. 2021; A. Zeng et al. 2023; A. Yang et al. 2022; Zhenhailong Wang et al. 2022; Zhengyuan Yang et al. 2022; et al. 2023]. LLMs have also been used in creative ways to obtain better supervision for training for various tasks [A. Yang et al. 2021; Zellers et al. 2022; X. Lin et al. 2022; Y. Zhao et al. 2022; Santurkar et al. 2022]. For example, [A. Yang et al. 2021] use LLMs to generate question-answer pairs from transcribed video narrations, while [Zellers et al. 2022] use LLMs to rephrase questions into

sentences. [X. Lin et al. 2022] use LLMs to match noisy speech transcriptions to step descriptions of procedural activities. [Nagrani et al. 2020] train BERT [Devlin et al. 2019] to predict action labels from transcribed speech segments and use this to scale up training data for action classification. [Y. Zhao et al. 2022] use pretrained LLMs conditioned on video to create automatic narrations. Recent works [Y. Zhao et al. 2022; Santurkar et al. 2022] also show the benefits of using LLMs to paraphrase captions for data augmentation for video-language pretraining. [M. Li et al. 2022] use LLMs to generate negative captions by manipulating event structures. Our work differs to [M. Li et al. 2022] in that we focus specifically on verb negatives, and videos instead of images. Most closely related to our work, [Park et al. 2022] construct a test set for verb understanding by leveraging T5 [Raffel et al. 2020] and highlight the poor performance of current video-language models. Our work is substantially different: (i) we automatically construct hard negative captions for *training* (not testing), (ii) we compare the use of different LLMs, (iii) we show that training with such negative captions can improve verb understanding on various verb-focused benchmarks.

Hard negatives for contrastive pretraining. Hard negatives have been used to improve performance in metric representation learning and contrastive learning [Kalantidis et al. 2020; Harwood et al. 2017; C. Wu et al. 2017]. Recent works mine hard negatives from an existing paired dataset [Radenovic et al. 2023; Hu Xu et al. 2021; J. Yang et al. 2021]. In comparison, in our work, we *generate* hard negative captions and propose a careful calibration mechanism for training effectively with such unpaired data. We also verify here the benefit of the HardNeg-NCE loss [Radenovic et al. 2023] when training with generated hard negative captions. [Yuksekgonul et al. 2023] construct hard negative captions by shuffling words from the original caption to improve order and compositionality understanding. Our work differs by (i) focusing specifically on *verb* reasoning, as opposed to object-attribute relationships, (ii) using LLMs to construct hard verb text negatives as opposed to perturbing the word order, (iii) focusing on *video*-language models.

Learning from parts-of-speech in video. Recent works use parts-of-speech (PoS) tags for video understanding [Sadhu et al. 2021; Wray and Damen 2019; Falcon et al. 2022; Ghadiyaram et al. 2019; Ran Xu et al. 2015]. [Wray and Damen 2019] learn multi-label verb-only representations, while other works fo-

cus on learning adverb representations [Doughty et al. 2019; Doughty and Snoek 2022]. [Alayrac et al. 2016] use verb-noun pairs for unsupervised learning with instructional videos, while [Falcon et al. 2022] leverage such pairs to generate data augmentations in the feature space. Other works exploit PoS for fine-grained or hierarchical alignment between video and text [Bowen Zhang et al. 2018; S. Chen et al. 2020]. [Wray et al. 2019] learn a separate multi-modal embedding space for each PoS tag and then combine these embeddings for fine-grained action retrieval. [S. Chen et al. 2020] construct a hierarchical semantic graph and use graph reasoning for local-global alignments. Most closely related to our work, [J. Yang et al. 2021] use a PoS based token contrastive loss. Our work differs in that: (i) we apply a verb phrase contrastive loss, as opposed to separate verb and noun losses; (ii) we extract verb phrases using a LLM and show this performs better than PoS tagging with NLTK [Bird et al. 2009] (Tab. 10.5); (iii) we evaluate our methods on verb-focused downstream tasks. Similarly to [Ghadiyaram et al. 2019], we find that training with verb phrase supervision helps for zero-shot performance on tasks with shorter sentences.

Temporal understanding in videos. A long term goal in computer vision is temporal understanding in videos [Joao Carreira and Zisserman 2017; Goyal et al. 2017; Diba et al. 2019; Schindler and van Gool 2008; C.-Y. Wu et al. 2019; Bolei Zhou et al. 2018; Sigurdsson et al. 2017]. However, current training and test datasets have a strong visual bias towards *objects* and *backgrounds* as well as *static* actions [Sevilla-Lara et al. 2021; D.-A. Huang et al. 2018], with some works [Buch et al. 2022; Lei et al. 2022] demonstrating strong results with a *single* frame. Despite these challenges, many recent works in video-only self-supervised learning propose pretext tasks for improving temporal modelling [D. Kim et al. 2019; Ahsan et al. 2019; Pickup et al. 2014; D. Wei et al. 2018; Price and Damen 2019; Benaim et al. 2020; Jiangliu Wang et al. 2020; Yuan Yao et al. 2020; Dorkenwald et al. 2022; Misra et al. 2016; Liang et al. 2021; Jue Wang et al. 2022; Behrmann et al. 2021; Recasens et al. 2021; Dave et al. 2022]. Unlike these works that use only video, [Y. Sun et al. 2022; M. Cao et al. 2022] focus on fine-grained temporal video-text alignment via localization of text sub-tokens. [Bagad et al. 2023] also leverage before/after relations in captions to create artificial training samples for video-text. Differently to these works (which create augmented video negatives

or positives), we approach the problem of improving *verb understanding* in video-language models from the language side, by leveraging the strong generalization capabilities of LLMs.

10.3 Method

Our goal is to adapt large-scale vision-language pretrained models (such as CLIP) to understand *verbs*. We aim to do this without requiring such models to be re-trained from scratch, but by simply fine-tuning them on a video-language dataset. However, given the pitfalls with using the standard video-text contrastive setup [Radford et al. 2021] on existing video-language datasets, we propose a new framework which we call **Verb-Focused Contrastive pretraining (VFC)**. It consists of two components, both using the power of LLMs: (i) a novel calibrated hard negative training method where we train with synthetic verb-focused hard negative captions, and (ii) an additional verb phrase loss where videos are contrasted against isolated verb phrases as opposed to the entire caption. Note that a ‘verb phrase’ can be a single verb or verb-noun pair depending on the caption (see Fig. 10.1).

10.3.1 Preliminaries

Large Language Models (LLMs) are generative text models with impressive capacities, in particular for few-shot or prompt-based learning [et al 2022]. In our work, we design prompts to instruct a LLM to (i) create verb-focused hard negative captions and (ii) isolate verb phrases from the captions of a dataset. LLMs allow scalability and generalisation, and as we show in the ablations (see Tab. 10.2 and 10.5), are preferable to manual or rule based methods (eg. NLTK [Bird et al. 2009]). In particular, we use PaLM [et al 2022], a state-of-art autoregressive model, throughout this paper. However, our framework is model agnostic and other LLMs can be used (see Tab. 10.2).

Video-language contrastive pretraining works by learning to distinguish between aligned and non-aligned video-text pairs. Given a dataset of N pairs $\{(V_i, T_i)\}_{i \in N}$ with video V_i and caption text T_i , we extract normalised feature representations v_i and t_i by using a video encoder f and text encoder g : we have $v_i = f(V_i)$ and $t_i = g(T_i)$. We use the InfoNCE loss [van den Oord et al. 2018] to make aligned (‘positive’) pairs close in feature space and all other pairwise

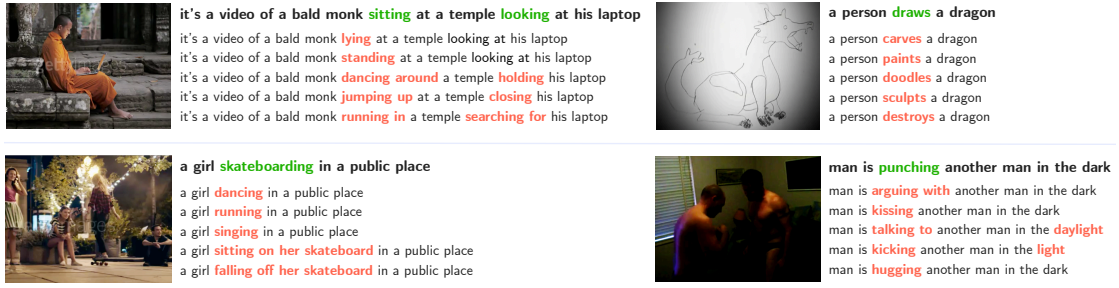


Figure 10.2: **Qualitative examples of hard negatives generated by PaLM.** We show a single frame per video and the corresponding caption in bold, with the verb highlighted in green. We see that PaLM can effectively generate hard negatives where the verb has changed (changes in red). When there are several verbs in the caption (see top left), PaLM may replace one or all verbs. As a failure case (bottom right), we show an example where PaLM can change more than just the verb, which could make it an easier negative (replacing ‘punching’ by ‘talking’ but also ‘dark’ by ‘daylight’).

combinations in the batch further apart [Radford et al. 2021]. We optimize for video-to-text L^{v2t} and text-to-video L^{t2v} alignments:

$$L_i^{t2v} = -t_i^\top v_i / \sigma + \log \sum_{j=1}^B \exp(t_i^\top v_j / \sigma) \quad (10.1)$$

where B is the batch size and σ a temperature parameter controlling the sharpness of the distribution. L^{v2t} is obtained by inverting v and t in Eq. 10.1.

Adapting image-text models to videos. We leverage CLIP [Radford et al. 2021] for video-language tasks following the CLIP4CLIP ‘seqTrans’ protocol [H. Luo et al. 2022]. Both single-modal encoders (video f and text g) are initialized with CLIP weights, with four additional temporal frame aggregation transformer blocks stacked on top of the image encoder (see [H. Luo et al. 2022] for more details). Our approach is agnostic to model architecture and so any state-of-the-art video-language architecture could be potentially used.

10.3.2 Verb-Focused Contrastive Pretraining (VFC)

We describe both our calibrated hard negative training (Sec. 10.3.2) and the proposed verb phrase loss (Sec. 10.3.2).

Calibrated Hard Negative training

In regular contrastive learning, given a video-caption pair, other captions in the batch are simply pushed further in the feature space. Since it is unlikely that existing datasets contain many examples with captions of similar context but different *verbs*, the task can be solved by paying little attention to verbs. Instead,

Name	Video-to-text alignment loss	R_ω
Baseline	$-v_i^\top t_i/\sigma + \log \sum_{j=1}^B \exp(v_i^\top t_j/\sigma)$	$\frac{(B-1)S_\omega}{S_\omega} \perp \omega$
HN	$-v_i^\top t_i/\sigma + \log \left(\sum_{j=1}^B \exp(v_i^\top t_j/\sigma) + \sum_{j=1}^B \sum_{k=1}^{N^{\text{hard}}} \exp(v_i^\top t_{jk}^{\text{hard}}/\sigma) \right)$	$\frac{(B-1)S_\omega + BG_\omega}{S_\omega} \propto B \frac{G_\omega}{S_\omega}$
Calibrated HN	$-v_i^\top t_i/\sigma + \log \left(\sum_{j=1}^B \exp(v_i^\top t_j/\sigma) + \sum_{k=1}^{N^{\text{hard}}} \exp(v_i^\top t_{i_k}^{\text{hard}}/\sigma) \right)$	$\frac{(B-1)S_\omega + G_\omega}{S_\omega} \propto \frac{G_\omega}{S_\omega}$ with $G_\omega \approx S_\omega$

Table 10.1: **Different choices for video-to-text alignment** when training with additional hard negatives (HN). R_ω is the ratio of the number of times a given verb phrase ω is used as a negative versus the number of times it is used as a positive. We note that for the regular contrastive loss (**Baseline**), R_ω only depends on the batch size B , however when training with generated hard negatives (**HN**), it depends on the verb phrase ω . We minimise this effect using our proposed **Calibrated HN** loss, which we denote as L_i^{CHN} . See details in Section 10.3.2.

our goal is to encourage the video-language model to focus on verb reasoning. We do so by tasking a LLM to generate hard negative captions where only the verb(s) in the captions change. Second, we train with these additional negative captions. We find that naive training with additional data leads to imbalances affecting the resulting video-text feature space. We propose a simple but effective calibration mechanism to solve this.

Generating verb-focused hard negatives with PaLM. Given a caption T_i , we task PaLM to replace the verbs with other verbs that convey a different action, but still form a linguistically and semantically viable sentence (which may not be guaranteed with random verb replacements – see qualitative examples in the appendix). For example, in the caption ‘*a man washes his face*’, the verb ‘*washes*’ should not be replaced with ‘*jumps*’ or ‘*plays*’. The generated caption is then a negative match for the corresponding video V_i (albeit a *hard* negative, as the nouns and context remain the same). We experiment with different handcrafted prompts, and find our best performing prompt to be the following: ‘*In this task, you are given an input sentence. Your job is to tell me 10 output sentences with a different meaning by only changing the action verbs*’. We also add four input-output pair examples to the prompt, which increases the quality of PaLM’s predictions (see in the appendix). We use one PaLM forward pass per caption T_i to generate ten verb-focused hard negatives for that caption (qualitative examples of the generated captions can be seen in Fig. 10.2). During training, we randomly sample N^{hard} generated captions for each pair (V_i, T_i) in the minibatch, which we denote $(T_{i_k}^{\text{hard}})_{k \in [1, N^{\text{hard}}]}$. Importantly, note that a $T_{i_k}^{\text{hard}}$ is a new generated text caption, or an *unpaired* data sample, meaning that it does not come with a corresponding matching (‘positive’) video.

Calibration. Interestingly, we observe that naively adding in negative captions into training with a contrastive loss leads to harmful feature space distortions, as some concepts are only seen in negative captions but never in positives. This is observed by careful analysis of downstream performance (see Tab. 10.3 and 10.4). We next describe a calibration mechanism to avoid such distortions: we first denote the vocabulary of all verb phrases in the original and generated captions as Ω . For each verb phrase ω (or ‘concept’) in Ω , we denote S_ω as the number of times it appears in the captions of the original dataset and G_ω as the number of times it appears in the PaLM-generated captions. We then derive equations for R_ω (see Tab. 10.1), which we define as the ratio of the number of times a verb phrase ω is used as a negative versus as a positive during training, for different choices of the video-to-text contrastive loss (note L^{t2v} is unchanged).

Contrastive training with paired data (Baseline). We first note that *the ratio R_ω is independent of the verb phrase ω* in regular contrastive learning (paired data only). It simply depends on the batch size B , as S_ω is cancelled from both the numerator and denominator. This means that the number of times a concept is used as a positive versus negative sample is the same regardless of the considered verb phrase. This naturally balances training, and is a great property of the contrastive framework.

Adding generated unpaired negative captions (HN). However, when training with unpaired captions, this ratio is proportional to G_ω/S_ω and therefore becomes *dependent* on the considered verb phrase ω . This can have significant consequences for the video-text feature representations. The model can learn to either ignore or always predict some concepts based on the average concept occurrences in positive or negative pairs during training.

Hard negatives with calibration (Calibrated HN). In order to make R_ω as ω -agnostic as possible, we introduce an ensemble of two techniques which we refer to as ‘calibration’. First, we ignore the hard negative captions from the other elements of the batch (see row 3 in Tab. 10.1), which allows us to mitigate the influence of G_ω/S_ω by not amplifying it by the batch size B (equal to 256). Second, we filter the generated PaLM captions to have $G_\omega \approx S_\omega$. In practice, we discard some generations so that the number of times a verb phrase appears in the set of kept generations is equal to the number of times it is originally present in the

dataset. We denote our video-to-text loss (text-to-video is unchanged) as L_i^{CHN} for calibrated hard negative training.

Video mining. An alternative to avoid imbalances due to the addition of negative captions would be to avoid training with unpaired data at all, by mining a matching video $V_{i_k}^{\text{hard}}$ for each generated caption $T_{i_k}^{\text{hard}}$. We attempt this via CLIP-based text-to-video retrieval in a large video database but found that finding a video matching a detailed, long caption is challenging, as such a precise video may not exist in a given corpus (see in the appendix for examples).

The verb phrase loss

In order to further encourage our model to focus on verbs, we introduce a contrastive ‘verb phrase’ loss. We use PaLM to extract the verb phrase T_i^{verb} in a caption T_i with the following prompt: *‘In this task, you are given an input sentence. Your job is to output the action verb phrases.’* While multiple parts-of-speech (PoS) tagging tools exist, we use a LLM for the following reasons: (i) we would like to isolate verb phrases, which may correspond to single verbs or verb-noun pairs depending on the caption, (ii) LLMs deal better with ambiguous cases (see qualitative examples in the appendix). We show the benefits experimentally via an ablation in Tab. 10.5. During training, we minimize the loss:

$$L_i^{\text{verb-phrase}} = -v_i^\top t_i^{\text{verb}} / \sigma + \log \sum_{j=1}^B \exp(v_i^\top t_j^{\text{verb}} / \sigma)$$

where the negative verb phrase representations t_j^{verb} simply come from other captions in the batch. Note that we do not require the calibration mechanism described in Section 10.3.2 since all verb phrases T_i^{verb} have a positive video match V_i (i.e. the video aligned with T_i).

Overall, our verb-focused contrastive (VFC) pretraining optimizes the sum of three objectives:

$$L^{\text{VFC}} = \frac{1}{B} \sum_{i=1}^B \left(\lambda_1 L_i^{\text{t2v}} + \lambda_2 L_i^{\text{CHN}} + \lambda_3 L_i^{\text{verb-phrase}} \right)$$

with parameters λ_1 , λ_2 and λ_3 weighting the contribution of the different terms. We learn the parameters of f and g via back-propagation.

10.3.3 Implementation details

Spoken Moments in Time (SMiT) pretraining dataset. The SMiT [Monfort et al. 2021] training set consists of 481K pairs of 3 seconds video clips with

corresponding captions. It is a subset of Moments in Time (MiT) [Monfort et al. 2019]. Our work falls under the umbrella of transfer learning: we pretrain on SMiT and then use the resulting features to solve different downstream tasks in a zero-shot or fine-tuned manner. Pretraining is either done as in regular contrastive learning (‘baseline’) or with our VFC framework. We find that the baseline already performs competitively on our benchmarks, despite the relatively small size of SMiT compared to other datasets such as HowTo100M [Miech et al. 2019], due to the quality and diversity of the manually annotated captions. We encourage the community to consider SMiT as a powerful pretraining dataset.

PaLM. We use PaLM-540B [et al 2022] with beam size 4, output sequence length 512, and temperature of 0.7. The negative captions are generated in an autogressive way and are therefore of arbitrary length. We post-process them by removing text after any newline character and by filtering out candidates which contain the same verbs as the original caption.

Training details. Most hyper-parameters follow CLIP4CLIP [H. Luo et al. 2022]. We initialise our model with CLIP ViT/B-32 and train with VFC for 100 epochs with a batch size of 256, base learning rate of $1e-7$, weight decay of $1e-2$, temperature of $5e-3$ and weights $\lambda_1 = 2$, $\lambda_2 = \lambda_3 = 1$ which we empirically find to work best in our experiments. Indeed, this balances the video-to-text and text-to-video loss terms. We also normalise each loss term by its value obtained from a random uniform prediction in order to have all loss terms in the same range (loss always equal to 1 for a random uniform prediction). We sample 32 frames per video at 25fps, with a 2 frame stride. See in the appendix for implementation and evaluation details.

10.4 Experiments

We curate a suite of benchmarks from existing works to evaluate verb understanding in Sec. 10.4.1. Then we ablate various components of VFC in Sec. 10.4.2. Finally, we demonstrate improved performance on our set of downstream tasks in Sec. 10.4.3, and compare to the state of the art.

Method	Hard negatives	Verb _H	K-400
Baseline	\emptyset	69.9	55.6
<i>w/o LLM</i>			
	Random verb	73.6 (+3.7)	55.0 (-0.6)
	Antonym verb	72.4 (+2.5)	55.4 (-0.2)
<i>w/ LLM</i>			
	T5 [Raffel et al. 2020]	75.1 (+5.2)	55.8 (+0.2)
Ours	PaLM [et al 2022]	78.0 (+8.1)	55.8 (+0.2)

Table 10.2: **Hard negatives generation.** We explore both LLM based and non LLM-based methods to obtain hard negative captions. Although PaLM LLM captions achieve the best performance, other LLMs (T5) achieve good results too. All methods are evaluated with calibration.

10.4.1 Verb-Focused Benchmarks

MSR-VTT multiple choice (MC) is a benchmark of 10K videos of length 10–30 secs. We evaluate on the standard 3k split and on Verb_H from [Park et al. 2022]. In this setting, the task is to associate each video to the right caption among five choices. While the four wrong captions are randomly chosen from other videos in the standard 3k split, one of them is replaced by a *hard verb negative* in Verb_H [Park et al. 2022].

Video question answering on NEXT-QA The train (resp. val) split contains 3870 (resp. 570) videos with 32K (resp. 5k) questions. There are three types of questions: causal (C), temporal (T) and descriptive (D). We consider the standard setting as well as ATP_{hard} [Buch et al. 2022], a subset automatically constructed with questions that are non-trivially solved with a single frame. ATP_{hard} is designed to be a better benchmark for the model’s true causal and temporal understanding which we believe is strongly related to verb reasoning.

Kinetics-400 is a video classification dataset with 400 human action classes. We report top-1, top-5 and their average classification accuracy. We follow [Radford et al. 2021] to evaluate classification in an open-set, zero-shot manner. This benchmark allows to assess transfer ability to *action* classification, which requires strong verb understanding (given actions are usually described with verb phrases).

SVO-probes dataset is a benchmark designed to measure progress in verb understanding of image-text models [Hendricks and Nematzadeh 2021]. It contains image-caption pairs with 421 different verbs. We simply replicate the image multiple times as input to our video model. We report Average Precision (AP) on the

Method	R_ω	# HN	Verb _H	K-400
Baseline	$\perp \omega$	0	69.9	55.6
w/o calibration	$\propto B \frac{G_\omega}{S_\omega}$	8.7M	80.5 (+10.6)	54.5 (-1.1)
w/ calibration	$\propto \frac{G_\omega}{S_\omega}, G_\omega \approx S_\omega$	0.9M	78.0 (+ 8.1)	55.8 (+0.2)

Table 10.3: **Importance of the calibration mechanism when training with hard negative captions.** The model trained without calibration suffers from a drop of performance on Kinetics.

	w/o calibration					w/ calibration				
$R_\omega \propto$	37	12	78	53	27	1	1	1	1	1
braiding hair										
brushing hair	38	14	1	1	5	47	2	5	2	7
curling hair	2	51	0	0	2	4	41	3	1	7
dying hair	1	33	31	0	5	4	7	55	3	3
fixing hair	0	12	1	36	4	1	1	4	45	6
	6	23	1	1	9	6	15	7	3	12
	braiding hair	brushing hair	curling hair	dying hair	fixing hair	braiding hair	brushing hair	curling hair	dying hair	fixing hair

Table 10.4: **Confusion matrix for the hair classes on Kinetics.** Without proper calibration, the verb phrase ‘brushing hair’ becomes highly attractive in the video-text feature space. This deteriorates the performance on all the ‘hair’ related classes. Our calibration mechanism alleviates this issue by making the ratio R_ω independent of verb phrases (see details in Sec. 10.3.2). More examples are shown in the appendix.

entire dataset as well as the verb-focused setting (details about our evaluation are provided in the appendix).

10.4.2 Ablation Study

In this section, we analyze our different design choices. We report results when transferring the models on two of our benchmarks: MSR-VTT multi-choice verb split (‘Verb_H’) and Kinetics-400 video classification (‘K-400’). We chose these two benchmarks as they have very different properties: the first involves captions, while the second involves action labels. We note that $N^{\text{hard}} = 1$ for all ablations unless otherwise specified.

PaLM captions:	Verb _H K-400		Verb isolation:	Verb _H K-400	
\emptyset	69.9	55.6	\emptyset	69.9	55.6
Positive	69.3	55.4	MiT labels	69.9	57.0
Negative	78.0	55.8	NLTK [Bird et al. 2009]	70.1	56.4
			PaLM [et al 2022]	70.3	57.6

Table 10.5: (left): **Generating negative *versus* positive captions with PaLM.** (right): **Verb phrase isolation methods.**

Hard negative captions generation. In Tab. 10.2, we ablate the technique used to obtain additional negative captions: we compare two LLMs (T5 [Raffel et al. 2020] and PaLM [et al 2022]) and two non LLM-based methods: (i) ‘random verb’: we replace verbs by random verbs from the UPenn XTag verb corpus and (ii) ‘antonym verb’: we replace verbs with their antonyms, using the NLTK [Bird et al. 2009] package. We see in Tab. 10.2 that ‘random verb’ and ‘antonym verb’ already give moderate performance gains on Verb_H compared to the baseline. However, using LLM-based generations improves the results by a large margin compared to the non LLM-based methods. This is likely due to the fact that (i) random or antonym replacements often create non semantically or linguistically plausible negative captions; (ii) some verbs do not have antonyms in NLTK (see qualitative examples in the appendix). Finally, we see in Tab. 10.2 that T5 generations work very well in our framework too, which demonstrates that our framework is LLM-agnostic. We observe that the best performance is achieved using PaLM, with a substantial gain over the baseline on MSR multi-choice (+8.1%) and a moderate gain on Kinetics (+0.2%).

Hard negative captions: the importance of calibration. We demonstrate the effect of the calibration mechanism described in Section 10.3.2 for training with unpaired captions. Tab. 10.3 shows the performance of hard negative training with (‘w/’) *versus* without (‘w/o’) calibration. First, we observe that the performance boost on MSR-VTT compared to the baseline is slightly stronger without calibration than with calibration. We believe this is because calibrating the PaLM generations reduces their number. However, we see that training with hard negatives without calibration deteriorates a lot the performance on Kinetics (−2.0% compared to the baseline). We hypothesize that this is due to some verb phrases being seen only as repulsive in the video-text feature space, while others are seen equally as attractive and repulsive. We illustrate this in Tab. 10.4 by

Method	Hard negatives	Verb phrase	Verb _H	K-400
Baseline			69.9	55.6
	✓		78.0 (+8.1)	55.8 (+0.2)
		✓	70.3 (+0.4)	57.6 (+2.0)
VFC (Ours)	✓	✓	76.3 (+6.4)	58.5 (+2.9)

Table 10.6: **Combining hard negative and verb phrase loss** achieves 9.2% and 5.2% relative improvements on MSR-VTT MC (acc.) and Kinetics (top-1) respectively compared to the baseline.

Method	N^{hard}	Verb _H	K-400
VFC (Ours)	1	76.3	58.5
VFC (Ours)	3	77.8	58.5
VFC (Ours)	5	78.3	58.5

Table 10.7: **Maximum number of hard negative captions.** We observe that increasing the maximum number of hard negative captions sampled per video increases the performance on Verb_H. We use $N^{\text{hard}} = 5$ in the remaining of the paper.

showing the confusion matrix for a subset of the Kinetics classes, along with the ratio R_ω (defined in Sec. 10.3.2) for each verb phrase. Intuitively, R_ω measures the ‘attraction’ (if low) and ‘repulsion’ (if high) of a verb phrase ω . The confusion matrix in Tab. 10.4 shows that the verb phrase ‘brushing hair’ becomes an attraction point in the absence of calibration. Indeed, the number of times the verb phrase ‘brushing hair’ is repulsive versus attractive is low ($R_{\text{brushing hair}} \approx 12$) compared to the other concepts such as ‘curling hair’ ($R_{\text{curling hair}} \approx 78$): we have $R_{\text{brushing hair}} \ll R_{\text{curling hair}}$. Hence, predictions for ‘brushing hair’ become dominant. This actually improves the performance for that class but deteriorates the performance on all the other classes related to ‘hair’. We see in Tab. 10.4 that our calibration mechanism alleviates this effect by making the ratio R_ω independent of ω as in regular contrastive learning. Calibration allows us to improve performance over the baselines on both tasks with a single model.

Generating positive *versus* negative captions. In Tab. 10.5 (left), we investigate the impact of generating *positive* captions instead of negatives with PaLM. In this case, positives correspond to sentences where the verb in the original caption is changed to a synonym verb, but the remaining context is unchanged: PaLM therefore acts as a data augmentation generator for text (similar to [Y. Zhao et al. 2022; Santurkar et al. 2022]). Details about positive caption generations are in

Method	Contrastive loss	Verb _H	K-400
Baseline	NCE	69.9	55.6
Baseline	HardNeg-NCE	72.0	56.4
VFC (Ours)	NCE	78.3	58.5
VFC (Ours)	HardNeg-NCE	80.5	58.8

Table 10.8: **Complementarity with other negative mining methods.** We observe that using the HardNeg-NCE loss, instead of standard NCE, gives the highest performance. We use HardNeg-NCE from now on. We note that for VFC we use $N^{\text{hard}} = 5$.

the appendix. We observe that using positive captions has a negative impact on the performance in our benchmarks, possibly because the model becomes more *invariant* to different verbs.

Verb phrase loss. In Tab. 10.5 (right), we explore two alternatives for verb phrase extraction used in the verb phrase loss: (i) using human-annotated action labels for clips from the Moments in Time (MiT) dataset (these are available as SMiT data inherits from MiT [Monfort et al. 2019]) and (ii) using a rule-based method (NLTK [Bird et al. 2009]) to isolate verbs. We observe in Tab. 10.5 that using PaLM to extract verb phrases from the caption outperforms both, probably because it extracts more fine-grained action information. Qualitative analysis of the verb phrases is shown in the appendix.

Combining calibrated hard negatives and verb phrase loss. We show in Tab. 10.6 the complementarity between our two contributions: the calibrated hard negative training and the verb phrase loss. The former greatly improves performance on tasks requiring complex language understanding such as MC Verb_H. On the other hand, the verb phrase loss improves transfer to video classification by focusing particularly on the action label in the sentence. We see in Tab. 10.6 that combining both approaches during training results in a *single model* with excellent performance on both MSR-VTT MC and Kinetics zero-shot transfer. Compared to the baseline, VFC pretraining achieves 9.2% and 5.2% relative improvements on MSR-VTT MC and Kinetics respectively.

Number of hard negative captions. In Tab. 10.7, we vary the maximum number of hard negative captions N^{hard} sampled per video in the batch. We find that setting this to 5 increases the performance on Verb_H while maintaining the performance on Kinetics. We use this setting going forward. We note that we do

Model	# params.	3k val.	Verb _H [Park et al. 2022]
ZERO-SHOT			
VideoCLIP [Hu Xu et al. 2021]	–	73.9	–
CLIP [Radford et al. 2021]	151M	91.1	64.1
InternVideo [Yi Wang et al. 2022]	≈ 460M	93.4	–
VFC (Ours)	164M	95.1	80.5
FINE-TUNED			
ClipBERT [Lei et al. 2021]	–	88.2	–
MMT [Gabeur et al. 2020]	–	92.4	71.3
VideoCLIP [Hu Xu et al. 2021]	–	92.1	–
CLIP-straight [Portillo-Quintero et al. 2021]	151M	94.1	65.1
MMT [Gabeur et al. 2020] (CLIP features)	–	95.0	71.4
C4CL-mP [Park et al. 2022]	151M	96.2	73.7
VFC (Ours)	164M	96.2	85.2

Table 10.9: **Multi-choice MSR-VTT**. We report accuracy on the 3k val and on the verb-focused Verb_H [Park et al. 2022] splits. While VFC improves the performance on both splits in a zero-shot setting, the gap with previous works is especially important on Verb_H [Park et al. 2022]. When available, we add model parameter counts.

not try larger values as our maximum number of hard negatives per video after calibration is 5.

Complementarity with other negative mining methods. We investigate whether our VFC framework is complementary to existing approaches for hard negatives with the contrastive learning framework. Specifically, we reimplement the hard negative noise contrastive multimodal alignment loss from [Radenovic et al. 2023; Robinson et al. 2021], which is denoted as HardNeg-NCE. With this objective, difficult negative pairs (with higher similarity) are emphasised, and easier pairs are ignored. We use $\alpha = 1$ and $\beta = 0.1$ in the equations from [Radenovic et al. 2023]. We note that we only adapt L_i^{t2v} and L_i^{CHN} with HardNeg-NCE. Adapting $L_i^{\text{verb-phrased}}$ does not bring further improvements, so we omit this for simplicity. We observe in Tab. 10.8 that VFC is complementary to existing frameworks: using HardNeg-NCE instead of the standard NCE loss achieves the highest performance. We observe a large boost on Verb_H [Park et al. 2022], a benchmark that specifically involves hard negatives. We adopt HardNeg-NCE in the remaining of this paper.

10.4.3 Comparisons to the State of the Art

We compare VFC to the state of the art on a diverse set of tasks requiring verb understanding. Note that we use the *same model* across different tasks, which is

Model					ATP _{hard} [Buch et al. 2022]		
	all	D	T	C	all	T	C
ZERO-SHOT							
CLIP [Radford et al. 2021]	43.9	57.0	38.1	43.6	23.0	21.8	23.8
VFC (Ours)	51.5	64.1	45.4	51.6	31.4	30.0	32.2
FINE-TUNED							
HGA [‡] [P. Jiang and Y. Han 2020]	49.7	59.3	50.7	46.3	44.1	45.3	43.3
ATP [Buch et al. 2022]	49.2	58.9	46.7	48.3	20.8	22.6	19.6
Temp[ATP] [Buch et al. 2022]	51.5	65.0	49.3	48.6	37.6	36.5	38.4
TAATP [†] [Xiao et al. 2022]	54.3	66.8	50.2	53.1	-	-	-
VGT [Xiao et al. 2022]	55.0	64.1	55.1	52.3	-	-	-
VFC (Ours)	58.6	72.8	53.3	57.6	39.3	38.3	39.9

Table 10.10: **NEXT-QA video question answering.** We report accuracy. We consider either ‘all’ questions or only causal (‘C’), temporal (‘T’) or descriptive (‘D’) questions. We also use ATP_{hard} split [Buch et al. 2022]. VFC improves performance for both zero-shot and fine-tuning. [†]Temp[ATP]+ATP. [‡] Uses additional motion features.

non-trivial as the tasks cover different domains and evaluation protocols.

MSR-VTT MC results. We see in Tab. 10.9 that our verb-focused pretraining transfers well to the MSR-VTT multi-choice task, especially on the hard verb split (curated to assess exactly the task we are trying to solve). We even outperform concurrent InternVideo [Yi Wang et al. 2022] while using a significantly smaller setting both in terms of architecture (InternVideo uses $2.8\times$ more parameters and $12.4\times$ more flops) and pretraining dataset size (they use $24\times$ more data). We also note that our method does not degrade performance on other standard object-based tasks, such as text-to-video retrieval on MSR-VTT (results compared to the state of the art are shown in the appendix).

NEXT-QA results. We show in Tab. 10.10 that our verb-focused pretraining gives a significant boost in both the standard and ATP_{hard} setting introduced by [Buch et al. 2022]. To the best of our knowledge, we are the first work to report zero-shot results for NEXT-QA and our zero-shot numbers improve upon some previously published fine-tuning numbers. Finally, although HGA [P. Jiang and Y. Han 2020] performs worse than ours on the standard setting, it achieves a high accuracy of 44.1 on ATP_{hard}. Their high performance on ATP_{hard} can be explained by the use of additional motion features, aiding in answering hard dynamics questions, as noted by [Buch et al. 2022]. The addition of extra motion features on the video side can be complementary to our verb-focused pretraining

Model	# param.	top-1	top-5	average
VAL-SET				
CLIP [Radford et al. 2021]	151M	48.9	75.8	62.4
ActionCLIP [M. Wang et al. 2021]	\approx 164M	56.4	-	-
VFC (Ours)	164M	59.4	85.3	72.4
TEST-SET				
Flamingo-3B [Alayrac et al. 2022]	3B	45.2	66.8	56.0
Flamingo-80B [Alayrac et al. 2022]	80B	49.1	71.5	60.3
Flamingo-9B [Alayrac et al. 2022]	9B	49.7	71.5	60.6
CLIP [Radford et al. 2021]	151M	47.9	75.1	61.5
VFC (Ours)	164M	58.8	84.5	71.7

Table 10.11: **Zero-shot transfer to Kinetics-400.** We report top-1 accuracy, top-5 accuracy, and their average on the validation and test set, as well as the parameter counts of the different models.

Method	all	Kinetics-verb
Baseline	55.6	52.1
VFC (Ours)	58.8 (+3.2)	57.1 (+5.0)

Table 10.12: **Zero-shot Kinetics-verb.** We report accuracy performance on our newly proposed Kinetics-verb split (from test split).

approach.

Zero-shot Kinetics-400 results. In Tab. 10.11 we see that our verb-focused features transfer very well to Kinetics video classification benchmark in a zero-shot setting, achieving state-of-the-art results. We achieve better results than Flamingo models [Alayrac et al. 2022] while using a significantly smaller model: relative improvement of 20% over Flamingo-80B model while using $489 \times$ less parameters.

Kinetics-verb. To further analyse the VFC framework’s effect on action classification, we introduce the Kinetics-verb split. We isolate classes from the Kinetics-400 dataset that share a common noun with another class, but have a different verb (and therefore action). For example, distinguishing between ‘braiding hair’, ‘brushing hair’ and ‘curling hair’ requires the model to focus on verb understanding as predictions cannot be inferred from the simple presence of hair in the frame. We use this rule to create a subset of 97 classes from the Kinetics-400 test set (see in the appendix) called ‘Kinetics-verb’. We show in Tab. 10.12 that our VFC improves substantially over the baseline (+5%) on this split.

Assessing verb understanding on SVO-probes. In Tab. 10.13, we see that

Model	AP	AP _{verb}
CLIP [Radford et al. 2021]	48.3	52.3
No-MRM-MMT [Hendricks and Nematzadeh 2021]†	51.5	53.1
Baseline (Ours)	60.2	61.9
VFC (Ours)	61.8	64.6

Table 10.13: **Verb understanding on SVO-probes** [Hendricks and Nematzadeh 2021]. We report Average Precision (AP) on the entire dataset and on the verb-specialized setting. †Scores provided by authors to calculate AP.

our VFC framework improves the performance on SVO-probes compared to the baseline (particularly in the verb setting), and outperforms prior work [Hendricks and Nematzadeh 2021] with 21.7% relative improvement in the verb setting.

10.5 Conclusion

Video-language models based on CLIP have been shown to have limited verb understanding, relying extensively on nouns. We attempt to alleviate this problem with two technical contributions on the contrastive learning framework: first, we leverage LLMs to automatically generate hard negative captions focused on verbs; second, we introduce a verb phrase alignment loss. We validate our verb-focused pretraining by showing improved performance on a suite of benchmarks, chosen in particular to assess verb understanding. Our framework is general and could be employed for other video-language tasks, and further readily scales with the rapid progress in language modelling.

Acknowledgements. We would like to thank Ahmet Iscen, Anurag Arnab, Paul Hongsuck Seo, Antoine Yang, Shyamal Buch, Alex Salcianu for their precious help and discussions. We also thank Sagar Vaze for his invaluable support.

Chapter 11

Discussion

In this chapter, we first provide a summary of the achievements and the impact of the presented works in this thesis (Section 11.1). We then briefly discuss ethical considerations (Section 11.2) and highlight directions for future works (Section 11.3).

11.1 Achievements and Impact

Keyword Spotting in Sign Language. In Chapter 2 (‘Audio-visual KWS’), we introduce a novel zero-shot keyword spotting method, suitable for in the wild videos, that can take as input either video, audio or both. The proposed method generalizes to other languages, specifically French and German. In Chapter 3 (‘Visual KWS’), we address the challenge of limited cross-modal interaction between the visual and phonetic streams by proposing a new spotting architecture based on Transformers. This model surpasses the previous state-of-the-art visual keyword spotting method presented in Chapter 2 on the challenging LRW [Chung and Zisserman 2016a], LRS2 [Chung et al. 2017], LRS3 [Afouras et al. 2018b] datasets by a significant margin. In Chapter 4 (‘BSL-1K’), we pioneer the use of a visual keyword spotting model along with weakly-aligned subtitles, which provide query words, to automatically label hundreds of thousands of signs in sign language interpreted TV shows via mouthing cues. The resulting collected training data not only enables training strong sign recognition model for co-articulated signs in BSL, but also serves as valuable pretraining data for other sign languages. Our scalable annotation strategy, leveraging signer mouthings, has been adopted in

other datasets such as How2Sign [Duarte et al. 2022], comprising American Sign Language.

Approaches for Sign Spotting. In Chapter 5 (‘Watch Read Lookup’), we propose to identify and localize signs in continuous signing by leveraging visual sign language dictionaries. Our approach not only aims to bridge the domain gap between continuous and isolated signing, but also uniquely considers the possibility of signs having multiple variants. This strategy greatly increases the amount of automatic annotations compared to relying solely on mouthings, both in terms of vocabulary coverage and number of instances. Employing dictionaries has been successfully applied to other datasets [Duarte et al. 2022] and inspired subsequent research in sign spotting [T. Jiang et al. 2021]. In Chapter 6 (‘Read and Attend’), we further extend our automatic annotation efforts by leveraging the attention mechanism of a Transformer trained on a video-to-text sequence prediction task. This approach enables the identification of hundreds of thousands of more signs. By combining these newly generated annotations with those previously collected from mouthings (Chapter 4) and dictionaries (Chapter 5), we develop a robust sign language recognition model. This model has served as a foundation for subsequent research on signing-subtitle retrieval [Y. Cheng et al. 2023] and alignment [Bull et al. 2021a]. In Chapter 7 (‘Automatic dense annotation’), we demonstrate the efficacy of pseudo-labelling from a sign recognition model as a way of sign spotting. We also introduce a novel approach of leveraging in-domain exemplars, further enhancing the density of annotations in sign language data.

Through our proposed sign spotting methodologies (outlined in Chapter 4 through 7), we curate the BOBSL BSL sign language corpus [Albanie et al. 2021b], including over 5M confident automatic annotations for a vocabulary of 25K signs. This large-scale dataset has supported research on diverse tasks, including sign recognition [Shen and anc Yi Yang 2022; Wong et al. 2023] and translation [Guo et al. 2024; Sincan et al. 2023; Sincan et al. 2024].

Sequence recognition in Sign Language. In Chapter 8 (‘Weakly-supervised fingerspelling’), we focus on BSL fingerspelling recognition, which is more challenging than American Sign Language (e.g., two-handed instead of one-handed). Our method both detects and recognises sequences of signed letters, using only weak annotations. Additionally, we contribute a test set of 5K video clips, anno-

tated by human experts for evaluating BSL fingerspelling recognition methods to support sign language research. In Chapter 9 (‘Large-vocabulary CSLR’), we design a multi-task transformer model, capable of performing both large vocabulary continuous sign language recognition (CSLR), and sign language retrieval. This marks an important step towards translation in open vocabulary settings. We also construct the largest CSLR test set, with continuous sign-level human annotations spanning six hours of videos, which will be made publicly available.

Enhancing verb representations. In Chapter 10 (‘Verbs in Action’), we propose the first method to tackle the verb understanding challenge in video-language models, while maintaining their effectiveness in noun-related tasks. Our framework utilizes LLMs to generate verb-focused hard negatives for cross-modal contrastive learning. Furthermore, it incorporates a fine-grained loss on isolated verb phrases. Our model achieves state of the art zero-shot verb understanding performance across a range of tasks and benchmarks, including MSR-VTT, Kinetics-400, NextQA and SVO-Probes. Our approach inspires many subsequent studies exploring the use of LLMs to augment text descriptions [X. Huang et al. 2024] and to generate hard negatives for action, event, compositional and fine-grained understanding [Zhenhailong Wang et al. 2023; Hakim et al. 2023; Sahin et al. 2023; G. Zhang et al. 2023; S et al. 2024], even in the context of diffusion models [Motamed et al. 2023].

11.2 Ethical considerations

In this section, we discuss some of the opportunities and limitations of the work in this thesis.

11.2.1 Applications

While research on sign language understanding holds promise for positive impact, it is crucial to ensure the resulting applications are genuinely useful and practical for deaf communities. To achieve this goal, it’s essential to involve deaf researchers and their perspectives from the outset of projects. This approach helps prevent misalignment between hearing researchers and members of the deaf community, thereby mitigating the risk of producing outcomes with limited practical

value [Bragg et al. 2019; Erard 2017].

In this context, we draw attention to two applications of particular significance to deaf communities: the efficient search and indexing of sign language videos, and the integration of sign-reading capabilities into virtual assistants such as Siri and Alexa [Bragg et al. 2019]. In fact, relying solely on text-based interfaces for communication with virtual assistants presents considerable practical limitations [Glasser et al. 2020]. Other applications include educational tools for sign language learners with features like auto-correct prompts (‘did you mean this sign?’) and real-time automatic interpreting in video calls, or critical scenarios such as hospitals, police stations, and airports, where human interpreters may not be readily available.

11.2.2 Limitations

While this thesis facilitates the creation of a large-scale dataset, there are important limitations to be highlighted. Firstly, the data covers interpreted signing, introducing certain biases compared to conversational signing. Interpreting can lead to a simplification in signing style and vocabulary, and even a reduction in speed for comprehension [Bragg et al. 2019]. More work is required to quantify and bridge this domain gap, allowing models to be transferred to native, conversational signing in real-world scenarios. Additionally, although the data spans a considerable number of hours, it involves only 39 signers, with a biased distribution of interpreters in terms of demographics and hearing status, potentially impacting signing style [Stone and Russell 2013]. These data biases can result in decreased model performance among underrepresented groups. Moreover, during the manual verification of test sets, a few signs were flagged as inappropriate (e.g. containing either harmful or racist content). While these signs have been removed from evaluation sets, it’s likely that similar instances exist within the remaining data. Efforts are needed to systematically eliminate such harmful content to prevent issues in models trained on this data. Finally, although interpreters have consented to the use of their footage for research purposes, additional measures should be explored for signer anonymisation to prevent tracking of particular individuals [Bigand et al. 2020].

11.3 Future Work

Here, we discuss several promising directions for future work related to this thesis.

11.3.1 Sign Language Translation

While recent works have shown promising outcomes for sign language translation in constrained settings [N. C. Camgoz et al. 2018; N. C. Camgoz et al. 2020b; Ko et al. 2019a], open-vocabulary sign language translation in the wild remains largely unsolved. In this thesis, the aim is to pave the way to this more realistic setting by focusing on automatic dense annotation of lexical signs (Chapter 7 ‘Automatic dense annotation’) and large vocabulary continuous sign language recognition (Chapter 9 ‘Large-vocabulary CSLR’). Indeed, research has indicated the advantages of sign-level supervision in enhancing translation performance [N. C. Camgoz et al. 2018; N. C. Camgoz et al. 2020b].

However, achieving sign language translation entails numerous additional steps. Firstly, understanding non/partially-lexical signs in sign language discourse is crucial. Partially lexical signs vary significantly based on context and are employed, for instance, to convey position, motion, size and shape of objects [Braffort and Filhol 2014]. Robust translation performance necessitates models capable of learning generalisable representations, that are also sensitive to context, given there is not a fixed sign language lexical dictionary as in spoken languages. Secondly, despite the known importance of non-manual components in sign languages [N. C. Camgoz et al. 2020a], our work does not explicitly address the simultaneous modeling of multiple articulators. In fact, this highly distinguishes sign languages from written languages, where a single stream of symbols is processed sequentially. To effectively leverage data from all articulators, translation architectures may necessitate hierarchical structures to generalize to such heterogeneous sources of data. Finally, to facilitate generalization in real-world scenarios, models must be trained with data from native signers and in more complex situations – such as conversations involving several signers or signers viewed from challenging viewpoints.

11.3.2 Model-assisted annotation for video

In this thesis, we explore various methods to automatically obtain stronger supervision for sign language datasets, such as leveraging other modalities (Chapter 3 ‘Visual KWS’) or visual exemplars (Chapter 5 ‘Watch Read Lookup’). However, it is interesting to also consider how such automatic labeling methods can be extended to other video tasks. Recent works utilize models pretrained on small annotated datasets [Kirillov et al. 2023], while others employ large language models [A. Yang et al. 2021; H. Liu et al. 2023], visual language models, or combinations [A. Zeng et al. 2023]. Advancing research in this direction for videos is important to enable scaling models without the need for manual annotation.

Further research is needed to determine the most efficient, automated methods for collecting annotations for videos. Annotating each individual frame in a video can incur substantial costs, particularly when leveraging large multimodal models. Furthermore, if the objective is to develop a comprehensive, unified video model that generalizes across numerous video tasks, encompassing both spatio-temporal localisation and understanding capabilities, it remains unclear what level of supervision in terms of quantity, granularity and style is optimal. Additionally, it’s important to consider the biases transferred through automatic labeling and iterative methods, necessitating thorough assessments of error propagation and annotation diversity. Finally, more research on how generalization varies as a function of dataset mixture can provide valuable insights to guide the automatic data curation process [Sorscher et al. 2022].

11.3.3 Temporal modeling

In Chapter 10 (‘Verb in Action’), we tackle the problem of improving verb understanding in video-language models from the language side, by leveraging the strong generalization capabilities of LLMs. Closely related to this theme is the broader goal in computer vision of temporal understanding in videos [Joao Carreira and Zisserman 2017; Goyal et al. 2017; Diba et al. 2019; Sigurdsson et al. 2017; Schindler and van Gool 2008; C.-Y. Wu et al. 2019]. Indeed, understanding temporal information is a fundamental ability required for intelligent systems, enabling the comprehension of relationships between events (such as causality and dynamics), and more generally physics and the structure of the world. For exam-

ple, by watching a cooking video, one might learn that a raw egg can be turned into an omelet but the omelet cannot be turned into a raw egg. In the context of videos, temporal understanding plays a pivotal role in various applications, such as video summarisation, compression and generation.

Further efforts are essential to design systems in a way that optimally captures temporal information. Firstly, there are fundamental challenges surrounding training data. Current training datasets have a strong visual bias towards single-frame concepts such as objects and static actions [Sevilla-Lara et al. 2021; D.-A. Huang et al. 2018], so models are less incentivised to learn temporal information from them. Moreover, achieving strong performance on current video and video-language benchmarks does not necessarily require temporal understanding, as evidenced by some studies that have achieved impressive results using a single frame [Buch et al. 2022; Lei et al. 2022]. Even when many frames are used, the evaluation might not require complex temporal reasoning to solve (for example, “Needle in a Haystack” [Reid et al. 2024]). Future endeavors should focus on either collecting or augmenting video data to ensure it contains sufficient richness in dynamics, both short and long-term, to prevent shortcuts in learning.

As progress is made in addressing data challenges, there are also many important modeling problems to solve from frame sampling, to tokenization and positional encoding strategies, as well as the design of model architectures and learning objectives. Further work is also required to develop efficient methods at training and inference for processing long videos, given their abundance of information compared to images and the potential redundancy across different frames. Additionally, we must account for the impact of variable-length videos on our modeling decisions. For instance, approaches akin to large language models may naturally accommodate varying duration, whereas special considerations may be needed for JEPA-style models [Bardes et al. 2024].

11.4 Conclusion

In this thesis, we develop multi-modal, weakly-supervised approaches to scale up sign language datasets in the large-vocabulary setting. We also demonstrate how the resulting large-scale, more strongly supervised data can empower sequence-

level tasks such as fingerspelling and continuous sign language recognition. These strides bring us closer to training translation models capable of generalizing in real-world contexts, where their impact is most profound. When coupled with forthcoming advancements in visual language models, these works will pave the way for the development of machines adept at bridging the gap between spoken and signed languages.

References

- Nikolas Adaloglou, Theodoris 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”. In: *arXiv preprint arXiv:2007.12530*.
- Triantafyllos Afouras, Joon Son Chung, Andrew Senior, Oriol Vinyals, and Andrew Zisserman (2019). “Deep Audio-Visual Speech Recognition”. In: *IEEE PAMI*.
- Triantafyllos Afouras, Joon Son Chung, and Andrew Zisserman (2018a). “Deep Lip Reading: a comparison of models and an online application”. In: *INTERSPEECH*.
- Triantafyllos Afouras, Joon Son Chung, and Andrew Zisserman (2018b). “LRS3-TED: a large-scale dataset for visual speech recognition”. In: *arXiv preprint arXiv:1809.00496*.
- Triantafyllos Afouras, Joon Son Chung, and Andrew Zisserman (2018c). “The Conversation: Deep Audio-Visual Speech Enhancement”. In: *INTERSPEECH*.
- Triantafyllos Afouras, Joon Son Chung, and Andrew Zisserman (2020). “ASR is all you need: Cross-modal distillation for lip reading”. In: *International Conference on Acoustics, Speech, and Signal Processing*.
- Rakesh Agrawal, Tomasz Imieliński, and Arun Swami (1993). “Mining association rules between sets of items in large databases”. In: *ACM SIGMOD International Conference on Management of Data*.
- Ulrich Agris, Jörg Zieren, Ulrich Canzler, Britta Bauer, and Karl-Friedrich Kraiss (2008). “Recent developments in visual sign language recognition”. In: *Universal Access in the Information Society*.
- Unaiza Ahsan, Rishi Madhok, and Irfan Essa (2019). “Video jigsaw: Unsupervised learning of spatiotemporal context for video action recognition.” In: *WACV*.

- Jean-Baptiste Alayrac, Piotr Bojanowski, Nishant Agrawal, Ivan Laptev, Josef Sivic, and Simon Lacoste-Julien (2016). “Unsupervised learning from Narrated Instruction Videos”. In: *CVPR*.
- Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob Menick, Sebastian Borgeaud, Andrew Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikolaj Binkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, and Karen Simonyan (2022). “Flamingo: a Visual Language Model for Few-Shot Learning”. In: *Neurips*.
- Samuel Albanie, Gül Varol, Liliane Momeni, Triantafyllos Afouras, Andrew Brown, Chuhan Zhang, Ernesto Coto, Necati Cihan Camgöz, Ben Saunders, Abhishek Dutta, Neil Fox, Richard Bowden, Bencie Woll, and Andrew Zisserman (2021a). “SeeHear: Signer diarisation and a new dataset”. In: *ICASSP*.
- Samuel Albanie, Gül Varol, Liliane Momeni, Triantafyllos Afouras, Hannah Bull, Himel Chowdhury, Neil Fox, Rob Cooper, Andrew McParland, Bencie Woll, and Andrew Zisserman (2021b). “BOBSL: BBC-Oxford British Sign Language Dataset”. In: *arXiv preprint arXiv:2111.03635*.
- 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: *Proc. ECCV*.
- Russell Aldersson and Lisa McEntee-Atalianis (Jan. 2007). “A lexical comparison of Icelandic sign language and Danish sign language”. In: *Birkbeck Studies in Applied Linguistics*.
- Epameinondas Antonakos, Anastasios Roussos, and Stefanos Zafeiriou (2015). “A survey on mouth modeling and analysis for Sign Language recognition”. In: *IEEE International Conference and Workshops on Automatic Face and Gesture Recognition*.
- Relja Arandjelovic and Andrew Zisserman (2017). “Objects that Sound”. In: *ECCV*.
- Sercan Arik, Markus Kliegl, Rewon Child, Joel Hestness, Andrew Gibiansky, Chris Fougner, Ryan Prenger, and Adam Coates (2017). “Convolutional Recurrent Neural Networks for Small-Footprint Keyword Spotting”. In: *INTERSPEECH*.
- Yannis M. Assael, Brendan Shillingford, Shimon Whiteson, and Nando de Freitas (2016). “LipNet: Sentence-level Lipreading”. In: *arXiv:1611.01599*.
- Vassilis Athitsos, Carol Neidle, Stan Sclaroff, Joan Nash, Alexandra Stefan, Ashwin Thangali, Haijing Wang, and Quan Yuan (2010). “Large lexicon project:

- American sign language video corpus and sign language indexing/retrieval algorithms”. In: *LREC*.
- Vassilis Athitsos, Carol Neidle, Stan Sclaroff, Joan Nash, Alexandra Stefan, Quan Yuan, and Ashwin Thangali (2008). “The American Sign Language Lexicon Video Dataset”. In: *CVPRW*.
- Kartik Audhkhasi, Andrew Rosenberg, Abhinav Sethy, Bhuvana Ramabhadran, and Brian Kingsbury (2017). “End to-end ASR-free keyword search from speech”. In: *IEEE Journal of Selected Topics in Signal Processing*.
- Piyush Bagad, Makarand Tapaswi, and Cees G. M. Snoek (2023). “Test of Time: Instilling Video-Language Models with a Sense of Time”. In: *arXiv preprint arXiv:2301.02074*.
- Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio (2015). “Neural machine translation by jointly learning to align and translate”. In: *ICLR*.
- Max Bain, Arsha Nagrani, Daniel Schofield, Sophie Berdugo, Joana Bessa, Jake Owens, Kimberley J Hockings, Tetsuro Matsuzawa, Misato Hayashi, Dora Biro, Susana Carvalho, and Andrew Zisserman (2021a). “Automated Audiovisual Behavior Recognition in Wild Primates”. In: *Science advances*.
- Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman (2021b). “Frozen in Time: A Joint Video and Image Encoder for End-to-End Retrieval”. In: *Proc. ICCV*.
- Richard Bank, Onno Crasborn, and Roeland Hout (2011). “Variation in mouth actions with manual signs in Sign Language of the Netherlands (NGT)”. In: *Sign Language & Linguistics*.
- Adrien Bardes, Quentin Garrido, Jean Ponce, Xinlei Chen, Michael Rabbat, Yann LeCun, Mido Assran, and Nicolas Ballas (2024). “Revisiting Feature Prediction for Learning Visual Representations from Video”. In: *arXiv preprint arXiv:2404.08471*.
- Britta Bauer and Hermann Hienz (2000). “Relevant features for video-based continuous sign language recognition”. In: *Proceedings Fourth IEEE International Conference on Automatic Face and Gesture Recognition (Cat. No. PR00580)*. IEEE.
- Nadine Behrmann, Mohsen Fayyaz, Juergen Gall, and Mehdi Noroozi (2021). “Long Short View Feature Decomposition via Contrastive Video Representation Learning”. In: *ICCV*.
- Valentin Belissen, Annelies Braffort, and Michèle Gouiffès (2020a). “Dicta-Sign-LSF-v2: Remake of a Continuous French Sign Language Dialogue Corpus and a First Baseline for Automatic Sign Language Processing”. In: *LREC*.

- Valentin Belissen, Annelies Braffort, and Michèle Gouiffès (2020b). “Experimenting the Automatic Recognition of Non-Conventionalized Units in Sign Language”. In: *Algorithms*.
- Sagie Benaim, Ariel Ephrat, Oran Lang, Inbar Mosseri, William T. Freeman, Michael Rubinstein, Michal Irani, and Tali Dekel (2020). “SpeedNet: Learning the Speediness in Videos”. In: *CVPR*.
- Axel Berg, Mark O’Connor, and Miguel Tairum Cruz (2021). “Keyword Transformer: A Self-Attention Model for Keyword Spotting”. In: *arXiv preprint arXiv:2104.00769*.
- Gedas Bertasius, Heng Wang, and Lorenzo Torresani (2021). “Is Space-Time Attention All You Need for Video Understanding?” In: *Proc. ICML*.
- Félix Bigand, Elise Prigent, and Annelies Braffort (2020). “Retrieving Human Traits From Gesture In Sign Language: The Example Of Gestural Identity”. In: *ICMC*.
- Yunus Can Bilge, Nazli Ikizler, and Ramazan Cinbis (2019). “Zero-Shot Sign Language Recognition: Can Textual Data Uncover Sign Languages?” In: *BMVC*.
- Steven Bird, Ewan Klein, and Edward Loper (2009). *Natural language processing with Python: analyzing text with the natural language toolkit*. " O’Reilly Media, Inc."
- Fedor Borisyuk, Albert Gordo, and Viswanath Sivakumar (2018). “Rosetta: Large scale system for text detection and recognition in images”. In: *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*.
- P Boyes Braem and RL Sutton-Spence (2001). *The Hands Are The Head of The Mouth. The Mouth as Articulator in Sign Languages*. English. Hamburg: Signum Press.
- Annelies Braffort and Michael Filhol (2014). *Constraints and Language*. Cambridge Scholars Publishing.
- Danielle Bragg, Oscar Koller, Mary Bellard, Larwan Berke, Patrick Boudreault, Annelies Braffort, Naomi Caselli, Matt Huenerfauth, Hernisa Kacorri, Tessa Verhoef, Christian Vogler, and Meredith Ringel Morris (2019). “Sign Language Recognition, Generation, and Translation: An Interdisciplinary Perspective”. In: *ACM SIGACCESS*.
- Diane Brentari (2009). “Effects of language modality on word segmentation: An experimental study of phonological factors in a sign language”. In: *Papers in laboratory phonology, vol.8*.
- Shyamal Buch, Cristobal Eyzaguirre, Adrien Gaidon, Jiajun Wu, Li Fei-Fei, and Juan Carlos Niebles (2022). “Revisiting the “Video” in Video-Language Understanding”. In: *CVPR*.
- Patrick Buehler, Mark Everingham, and Andrew Zisserman (2009). “Learning Sign Language by Watching TV (using Weakly Aligned Subtitles)”. In: *Proc. CVPR*.

- Hannah Bull, Triantafyllos Afouras, Gül Varol, Samuel Albanie, Liliane Momeni, and Andrew Zisserman (2021a). “Aligning Subtitles in Sign Language Videos”. In: *Proc. ICCV*. IEEE.
- Hannah Bull, Triantafyllos Afouras, Gül Varol, Samuel Albanie, Liliane Momeni, and Andrew Zisserman (2021b). “Aligning subtitles in sign language videos”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*.
- Hannah Bull, Michèle Gouiffès, and Annelies Braffort (2020). “Automatic Segmentation of Sign Language into Subtitle-Units”. In: *ECCVW*.
- Necati Camgoz, Ben Saunders, Guillaume Rochette, Marco Giovanelli, Giacomo Inches, Robin Nachtrab-Ribback, and Richard Bowden (2021). “Content4All Open Research Sign Language Translation Datasets”. In: *arXiv preprint arXiv:2105.02351*.
- Necati Cihan Camgoz, Simon Hadfield, Oscar Koller, and Richard Bowden (2016). “Using Convolutional 3D Neural Networks for User-Independent Continuous Gesture Recognition”. In: *IEEE International Conference of Pattern Recognition, ChaLearn Workshop*.
- Necati Cihan Camgoz, Simon Hadfield, Oscar Koller, and Richard Bowden (Oct. 2017). “SubUNets: End-to-end Hand Shape and Continuous Sign Language Recognition”. In: *Proc. ICCV*.
- Necati Cihan Camgoz, Simon Hadfield, Oscar Koller, Hermann Ney, and Richard Bowden (2018). “Neural Sign Language Translation”. In: *CVPR*.
- Necati Cihan Camgoz, Oscar Koller, Simon Hadfield, and Richard Bowden (2020a). “Multi-channel Transformers for Multi-articulatory Sign Language Translation”. In: *ECCVW*.
- Necati Cihan Camgoz, Oscar Koller, Simon Hadfield, and Richard Bowden (2020b). “Sign Language Transformers: Joint End-to-end Sign Language Recognition and Translation”. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Dogan Can and M. Saraçlar (2011). “Lattice Indexing for Spoken Term Detection”. In: *IEEE Transactions on Audio, Speech, and Language Processing*.
- Kaidi Cao, Jingwei Ji, Zhangjie Cao, Chien-Yi Chang, and Juan Carlos Niebles (2020). “Few-Shot Video Classification via Temporal Alignment”. In: *CVPR*.
- Meng Cao, Tianyu Yang, Junwu Weng, Can Zhang, Jue Wang, and Yuexian Zou (2022). “LocVTP: Video-Text Pre-training for Temporal Localization”. In: *ECCV*.

- Qiong Cao, Li Shen, Weidi Xie, Omkar M. Parkhi, and Andrew Zisserman (2018).
 “VGGFace2: A dataset for recognising faces across pose and age”. In: *Proc. Int. Conf. Autom. Face and Gesture Recog.*
- Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh (2018).
 “OpenPose: realtime multi-person 2D pose estimation using Part Affinity Fields”.
 In: *arXiv preprint arXiv:1812.08008*.
- Joao Carreira and Andrew Zisserman (2017). “Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset”. In: *CVPR*.
- João Carreira and Andrew Zisserman (2017). “Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset”. In: *CVPR*.
- Xiujuan Chai, Hanjie Wang, and Xilin Chen (2014). “The devisign large vocabulary of chinese sign language database and baseline evaluations”. In: *Technical report VIPL-TR-14-SLR-001. Key Lab of Intelligent Information Processing of Chinese Academy of Sciences (CAS), Institute of Computing Technology, CAS*.
- William Chan, Navdeep Jaitly, Quoc Le, and Oriol Vinyals (2016). “Listen, attend and spell: A neural network for large vocabulary conversational speech recognition”. In: *ICASSP*.
- Woong-Gi Chang, Tackgeun You, Seonguk Seo, Suha Kwak, and Bohyung Han (2019).
 “Domain-specific batch normalization for unsupervised domain adaptation”. In: *CVPR*.
- C Charayaphan and AE Marble (1992). “Image processing system for interpreting motion in American Sign Language”. In: *Journal of Biomedical Engineering*.
- Guoguo Chen, Carolina Parada, and Georg Heigold (2014). “Small-footprint keyword spotting using deep neural networks”. In: *ICASSP*.
- Jingyuan Chen, Xinpeng Chen, Lin Ma, Zequn Jie, and Tat-Seng Chua (2018).
 “Temporally grounding natural sentence in video”. In: *EMNLP*.
- Jun Chen, Han Guo, Kai Yi, Boyang Li, and Mohamed Elhoseiny (2022). “VisualGPT: Data-Efficient Adaptation of Pretrained Language Models for Image Captioning”.
 In: *CVPR*.
- Shizhe Chen, Yida Zhao, Qin Jin, and Qi Wu (2020). “Fine-grained Video-Text Retrieval with Hierarchical Graph Reasoning”. In: *CVPR*.
- Xi Chen, Shouyi Yin, Dandan Song, Peng Ouyang, Leibo Liu, and Shaojun Wei (2019).
 “Small-footprint Keyword Spotting with Graph Convolutional Network”. In: *IEEE Automatic Speech Recognition and Understanding Workshop*.
- Ka Leong Cheng, Zhaoyang Yang, Qifeng Chen, and Yu-Wing Tai (2020). “Fully Convolutional Networks for Continuous Sign Language Recognition”. In: *ECCV*.

- Yiting Cheng, Fangyun Wei, Bao Jianmin, Dong Chen, and Wen Qiang Zhang (2023). “CiCo: Domain-Aware Sign Language Retrieval via Cross-Lingual Contrastive Learning”. In: *CVPR*.
- Minsu Cho, Suha Kwak, Cordelia Schmid, and Jean Ponce (2015). “Unsupervised object discovery and localization in the wild: Part-based matching with bottom-up region proposals”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*.
- Joon Son Chung, Andrew Senior, Oriol Vinyals, and Andrew Zisserman (2017). “Lip Reading Sentences in the Wild”. In: *Proc. CVPR*.
- Joon Son Chung and Andrew Zisserman (2016a). “Lip Reading in the Wild”. In: *Proc. ACCV*.
- Joon Son Chung and Andrew Zisserman (2016b). “Out of time: automated lip sync in the wild”. In: *Workshop on Multi-view Lip-reading, ACCV*.
- Joon Son Chung and Andrew Zisserman (2016c). “Signs in time: Encoding human motion as a temporal image”. In: *Workshop on Brave New Ideas for Motion Representations, ECCV*.
- Helen Cooper and Richard Bowden (2009). “Learning signs from subtitles: A weakly supervised approach to sign language recognition”. In: *CVPR*.
- Helen Cooper, Nicolas Pugeault, and Richard Bowden (2011). “Reading the signs: A video based sign dictionary”. In: *ICCVW*.
- Alice Coucke, Mohammed Chlieh, Thibault Gisselbrecht, David Leroy, Mathieu Poumeyrol, and Thibaut Lavril (2019). “Efficient keyword spotting using dilated convolutions and gating”. In: *ICASSP*.
- Onno A Crasborn, Els Van Der Kooij, Dafydd Waters, Bencie Woll, and Johanna Mesch (2008). “Frequency distribution and spreading behavior of different types of mouth actions in three sign languages”. In: *Sign Language & Linguistics*.
- Runpeng Cui, Hu Liu, and Changshui Zhang (2017). “Recurrent convolutional neural networks for continuous sign language recognition by staged optimization”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*.
- Runpeng Cui, Hu Liu, and Changshui Zhang (2019). “A Deep Neural Framework for Continuous Sign Language Recognition by Iterative Training”. In: *IEEE Transactions on Multimedia*.
- Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc V Le, and Ruslan Salakhutdinov (2019). “Transformer-XL: Attentive language models beyond a fixed-length context”. In: *ACL*.

- Pradipto Das, Chenliang Xu, Richard F. Doell, and Jason J. Corso (2013). “A Thousand Frames in Just a Few Words: Lingual Description of Videos through Latent Topics and Sparse Object Stitching.” In: *Proc. CVPR*.
- Ishan Dave, Rohit Gupta, Mamshad Nayeem Rizve, and Mubarak Shah (2022). “Tclr: Temporal contrastive learning for video representation”. In: *Computer Vision and Image Understanding*.
- Jeffrey Davis (1990). “Linguistic Transference and Interference: Interpreting Between”. In: *Sign language research: Theoretical issues*.
- Mathieu De Coster, Mieke Van Herreweghe, and Joni Dambre (2020). “Sign language recognition with transformer networks”. In: *12th international conference on language resources and evaluation*. European Language Resources Association (ELRA).
- Chaorui Deng, Qi Wu, Qingyao Wu, Fuyuan Hu, Fan Lyu, and Mingkui Tan (2018). “Visual grounding via accumulated attention”. In: *CVPR*.
- Thomas Deselaers, Bogdan Alexe, and Vittorio Ferrari (2010). “Localizing objects while learning their appearance”. In: *Proc. ECCV*.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova (2019). “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding”. In: *ACL*.
- Ali Diba, Mohsen Fayyaz, Vivek Sharma, Manohar Paluri, Jurgen Gall, Rainer Stiefelhagen, and Luc Van Gool (2019). “Large Scale Holistic Video Understanding”. In: *ECCV*.
- Thomas G Dietterich, Richard H Lathrop, and Tomás Lozano-Pérez (1997). “Solving the multiple instance problem with axis-parallel rectangles”. In: *Artificial intelligence*.
- Runwei Ding, Cheng Pang, and Hong Liu (2018). “Audio-Visual Keyword Spotting Based on Multidimensional Convolutional Neural Network”. In: *IEEE International Conference on Image Processing*.
- Linhao Dong, Shuang Xu, and Bo Xu (2018). “Speech-Transformer: A No-Recurrence Sequence-to-Sequence Model for Speech Recognition”. In: *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*.
- Michael Dorkenwald, Fanyi Xiao, Biagio Brattoli, Joseph Tighe, and Davide Modolo (2022). “SCVRL: Shuffled contrastive video representation learning”. In: *CVPRW*.
- Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer,

- Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby (2021). “An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale”. In: *ICLR*.
- Hazel Doughty, Ivan Laptev, Walterio Mayol-Cuevas, and Dima Damen (2019). “Action Modifiers: Learning From Adverbs in Instructional Videos”. In: *CVPR*.
- Hazel Doughty and Cees G. M. Snoek (2022). “How Do You Do It? Fine-Grained Action Understanding with Pseudo-Adverbs”. In: *CVPR*.
- Philippe Dreuw, Jens Forster, Thomas Deselaers, and Hermann Ney (2008). “Efficient Approximations to Model-based Joint Tracking and Recognition of Continuous Sign Language”. In: *IEEE International Conference on Automatic Face and Gesture Recognition*.
- Amanda Duarte, Samuel Albanie, Xavier Giró-i-Nieto, and Gül Varol (2022). “Sign Language Video Retrieval with Free-Form Textual Queries”. In: *CVPR*.
- 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)*.
- Abhishek Dutta and Andrew Zisserman (Oct. 2019). “The VIA Annotation Software for Images, Audio and Video”. In: *Proc. ACMM*. MM 19. to appear in Proceedings of the 27th ACM International Conference on Multimedia (MM 19). ACM. New York, USA: ACM.
- Eng-Jon Ong, O. Koller, N. Pugeault, and R. Bowden (2014). “Sign Spotting Using Hierarchical Sequential Patterns with Temporal Intervals”. In: *CVPR*.
- Michael Erard (2017). *Why Sign-Language Gloves Don’t Help Deaf People*. The Atlantic, <https://www.theatlantic.com/technology/archive/2017/11/why-sign-language-gloves-dont-help-deaf-people/545441/>.
- Aakanksha Chowdhery et al (2022). “PaLM: Scaling Language Modeling with Pathways”. In: *arXiv preprint arXiv:2204.02311*.
- Zhenfang Chen et al. (2023). “See, Think, Confirm: Interactive Prompting Between Vision and Language Models for Knowledge-based Visual Reasoning”. In: *arXiv preprint arXiv:2301.05226*.
- Alex Falcon, Giuseppe Serra, and Oswald Lanz (2022). “A Feature-Space Multimodal Data Augmentation Technique for Text-Video Retrieval”. In: *ACM*.
- Angela Fan, Shruti Bhosale, Holger Schwenk, Zhiyi Ma, Ahmed El-Kishky, Siddharth Goyal, Mandeep Baines, Onur Celebi, Guillaume Wenzek, Vishrav Chaudhary, Naman Goyal, Tom Birch, Vitaliy Liptchinsky,

- Sergey Edunov, Edouard Grave, Michael Auli, and Armand Joulin (2021). “Beyond english-centric multilingual machine translation”. In.
- Gaolin Fang, Xiujuan Gao, Wen Gao, and Yiqiang Chen (2004). “A novel approach to automatically extracting basic units from chinese sign language”. In: *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004*. IEEE.
- A. Farhadi and D. Forsyth (2006). “Aligning ASL for Statistical Translation Using a Discriminative Word Model”. In: *CVPR*.
- Ali Farhadi and David Forsyth (2006). “Aligning ASL for statistical translation using a discriminative word model”. In: *CVPR*.
- Ali Farhadi, David Forsyth, and Ryan White (2007). “Transfer learning in sign language”. In: *IEEE conference on computer vision and pattern recognition*.
- Ingo Feinerer and Kurt Hornik (2020). *wordnet: WordNet Interface*. R package version 0.1-15. URL: <https://CRAN.R-project.org/package=wordnet>.
- Yang Feng, Lin Ma, Wei Liu, Tong Zhang, and Jiebo Luo (2018). “Video Re-localization”. In: *ECCV*.
- Santiago Fernandez, Alex Graves, and Jurgen Schmidhuber (2007). “An application of recurrent neural networks to discriminative keyword spotting”. In: *International Conference on Artificial Neural Networks*.
- Basura Fernando, Hakan Bilen, Efstratios Gavves, and Stephen Gould (2017). “Self-Supervised Video Representation Learning With Odd-One-Out Networks”. In: *Proc. ICCV*.
- Michael Filhol (2020). “Elicitation and Corpus of Spontaneous Sign Language Discourse Representation Diagrams”. In: *LREC*.
- Holger Fillbrandt, Suat Akyol, and Karl-Friedrich Kraiss (2003). “Extraction of 3D hand shape and posture from image sequences for sign language recognition”. In: *IEEE International SOI Conference*.
- Cletus G. Fisher (1968). “Confusions Among Visually Perceived Consonants”. In: *Journal of Speech and Hearing Research*.
- Jens Forster, Christian Oberdörfer, Oscar Koller, and Hermann Ney (2013). “Modality Combination Techniques for Continuous Sign Language Recognition”. In: *Pattern Recognition and Image Analysis*.
- Frank Fowley and Anthony Ventresque (2021). “Sign Language Fingerspelling Recognition using Synthetic Data.” In: *AICS*.
- Valentin Gabeur, Chen Sun, Karteeek Alahari, and Cordelia Schmid (2020). “Multi-modal transformer for video retrieval”. In: *ECCV*.

- Jiyang Gao, Chen Sun, Zhenheng Yang, and Ram Nevatia (2017). “Tall: Temporal activity localization via language query”. In: *ICCV*.
- Sarthak Garg, Stephan Peitz, Udhyakumar Nallasamy, and Matthias Paulik (2019). “Jointly Learning to Align and Translate with Transformer Models”. In: *EMNLP*.
- Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A. Wichmann (2020). “Shortcut learning in deep neural networks”. In: *Nature Machine Intelligence*.
- Deepti Ghadiyaram, Matt Feiszli, Du Tran, Xueting Yan, Heng Wang, and Dhruv Kumar Mahajan (2019). “Large-Scale Weakly-Supervised Pre-Training for Video Action Recognition”. In: *CVPR*.
- Soham Ghosh, Anuva Agarwal, Zarana Parekh, and Alexander Hauptmann (2019). “ExCL: Extractive Clip Localization Using Natural Language Descriptions”. In: *NAACL-HLT*.
- Abraham Glasser, Vaishnavi Mande, and Matt Huenerfauth (2020). “Accessibility for Deaf and Hard of Hearing Users: Sign Language Conversational User Interfaces”. In: *Proceedings of the 2nd Conference on Conversational User Interfaces*.
- Paul Goh and Eun-Jung Holden (2006). “Dynamic fingerspelling recognition using geometric and motion features”. In: *2006 International Conference on Image Processing*. IEEE.
- Ramazan Gokberk Cinbis, Jakob Verbeek, and Cordelia Schmid (2014). “Multi-fold mil training for weakly supervised object localization”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*.
- John N Gowdy, Amarnag Subramanya, Chris Bartels, and Jeff Bilmes (2004). “DBN based multi-stream models for audio-visual speech recognition”. In: *2004 IEEE International conference on acoustics, speech, and signal processing*. IEEE.
- Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fründ, Peter N. Yianilos, Moritz Mueller-Freitag, Florian Hoppe, Christian Thureau, Ingo Bax, and Roland Memisevic (2017). “The “Something Something” Video Database for Learning and Evaluating Visual Common Sense”. In: *ICCV*.
- Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber (2006). “Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks”. In: *Proc. ICML*. ACM.
- Anmol Gulati, James Qin, Chung-Cheng Chiu, Niki Parmar, Yu Zhang, Jiahui Yu, Wei Han, Shibo Wang, Zhengdong Zhang, Yonghui Wu, and Ruoming Pang (2020).

- “Conformer: Convolution-augmented Transformer for Speech Recognition”. In: *INTERSPEECH*.
- Zhengsheng Guo, Zhiwei He, Wenxiang Jiao, Xing Wang, Rui Wang, Kehai Chen, Zhaopeng Tu, Yong Xu, and Min Zhang (2024). “Unsupervised Sign Language Translation and Generation”. In: *Proc. ICLR*.
- Michael Gutmann and Aapo Hyvärinen (2010). “Noise-contrastive estimation: A new estimation principle for unnormalized statistical models”. In: *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*.
- Zaber Ibn Abdul Hakim, Najibul Haque Sarker, Rahul Pratap Singh, Bishmoy Paul, Ali Dabouei, and Min Xu (2023). “Leveraging Generative Language Models for Weakly Supervised Sentence Component Analysis in Video-Language Joint Learning”. In: *arXiv preprint arXiv:2312.06699*.
- Tengda Han, Weidi Xie, and Andrew Zisserman (2019). “Video Representation Learning by Dense Predictive Coding”. In: *Workshop on Large Scale Holistic Video Understanding, ICCV*.
- Thomas Hanke (2004). “HamNoSys - representing sign language data in language resources and language processing contexts”. In: *LREC Workshop proceedings: Representation and processing of sign languages*.
- Laura Hanu, James Thewlis, Yuki M. Asano, and Christian Rupprecht (2022). “VTC: Improving Video-Text Retrieval with User Comments”. In: *Proc. ECCV*.
- David Harwath, Adria Recasens, Dídac Surís, Galen Chuang, Antonio Torralba, and James Glass (2018). “Jointly discovering visual objects and spoken words from raw sensory input”. In: *ECCV*.
- Ben Harwood, Vijay Kumar B.G., Gustavo Carneiro, Ian Reid, and Tom Drummond (2017). “Smart Mining for Deep Metric Learning”. In: *ICCV*.
- Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick (2020). “Momentum contrast for unsupervised visual representation learning”. In: *CVPR*.
- Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun (2016). “Deep Residual Learning for Image Recognition”. In: *Proc. CVPR*.
- Yanzhang He, Rohit Prabhavalkar, Kanishka Rao, Wei Li, Anton Bakhtin, and Ian McGraw (2017). “Streaming small-footprint keyword spotting using sequence-to-sequence models”. In: *IEEE Automatic Speech Recognition and Understanding Workshop*.
- Lisa Anne Hendricks and Aida Nematzadeh (2021). “Probing image-language transformers for verb understanding”. In: *ACL*.

- Lisa Anne Hendricks, O. Wang, E. Shechtman, Josef Sivic, Trevor Darrell, and Bryan C. Russell (2017). “Localizing Moments in Video with Natural Language”. In: *ICCV*.
- Sepp Hochreiter, Yoshua Bengio, Paolo Frasconi, Jürgen Schmidhuber, et al. (2001). *Gradient flow in recurrent nets: the difficulty of learning long-term dependencies*.
- Sepp Hochreiter and Jürgen Schmidhuber (1997). “Long short-term memory”. In: *Neural computation*.
- <https://www.signbsl.com/> (n.d.). *British Sign Language Dictionary*. URL: <https://www.signbsl.com/>.
- Anthony Hu, Lloyd Russell, Hudson Yeo, Zak Murez, George Fedoseev, Alex Kendall, Jamie Shotton, and Gianluca Corrado (2023). “GAIA-1: A Generative World Model for Autonomous Driving”. In: *arXiv preprint arXiv:2309.17080*.
- Hezhen Hu, Weichao Zhao, Wengang Zhou, Yuechen Wang, and Houqiang Li (2021a). “SignBERT: Pre-Training of Hand-Model-Aware Representation for Sign Language Recognition”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*.
- Hezhen Hu, Wengang Zhou, Junfu Pu, and Houqiang Li (2021b). “Global-local enhancement network for NMF-aware sign language recognition”. In: *ACM transactions on multimedia computing, communications, and applications (TOMM)*.
- Jie Hu, Li Shen, Samuel Albanie, Gang Sun, and Enhua Wu (2019). “Squeeze-and-Excitation Networks”. In: *IEEE PAMI*.
- De-An Huang, Vignesh Ramanathan, Dhruv Mahajan, Lorenzo Torresani, Manohar Paluri, Li Fei-Fei, and Juan Carlos Niebles (2018). “What Makes a Video a Video: Analyzing Temporal Information in Video Understanding Models and Datasets”. In: *CVPR*.
- Jie Huang, Wengang Zhou, Houqiang Li, and Weiping Li (2015). “Sign Language Recognition using 3D convolutional neural networks”. In: *International Conference on Multimedia and Expo (ICME)*.
- Jie Huang, Wengang Zhou, Houqiang Li, and Weiping Li (2018a). “Attention-based 3D-CNNs for large-vocabulary sign language recognition”. In: *IEEE Transactions on Circuits and Systems for Video Technology*.
- Jie Huang, Wengang Zhou, Qilin Zhang, Houqiang Li, and Weiping Li (2018b). “Video-based Sign Language Recognition without Temporal Segmentation”. In: *AAAI*.
- Xiaohu Huang, Hao Zhou, Kun Yao, and Kai Han (2024). “FROSTER: Frozen CLIP Is A Strong Teacher for Open-Vocabulary Action Recognition”. In: *Proc. ICLR*.

- Yuqi Huo, Mingyu Ding, Haoyu Lu, Zhiwu Lu, Tao Xiang, Ji-Rong Wen, Ziyuan Huang, Jianwen Jiang, Shiwei Zhang, Mingqian Tang, Songfang Huang, and Ping Luo (2021). “Selfsupervised video representation learning with constrained spatiotemporal jigsaw”. In: *Proc. IJCAI*.
- Kyuyeon Hwang, Minjae Lee, and Wonyong Sung (2015). “Online Keyword Spotting with a Character-Level Recurrent Neural Network”. In: *arXiv:1512.08903*.
- Fumitada Itakura (1990). “Minimum Prediction Residual Principle Applied to Speech Recognition”. In: *Readings in Speech Recognition*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.
- Abhishek Jha, Vinay P. Namboodiri, and C. V. Jawahar (2018). “Word spotting in silent lip videos”. In: *IEEE Winter Conference on Applications of Computer Vision*.
- Pin Jiang and Yahong Han (2020). “Reasoning with Heterogeneous Graph Alignment for Video Question Answering”. In: *AAAI*.
- Tao Jiang, Necati Cihan Camgoz, and Richard Bowden (2021). “Looking for the Signs: Identifying Isolated Sign Instances in Continuous Video Footage”. In: *IEEE International Conferene on Automatic Face and Gesture Recognition*.
- Peiqi Jiao, Yuecong Min, Yanan Li, Xiaotao Wang, Lei Lei, and Xilin Chen (2023). “CoSign: Exploring Co-occurrence Signals in Skeleton-based Continuous Sign Language Recognition”. In: *ICCV*.
- Tao Jin and Zhou Zhao (2021). “Contrastive Disentangled Meta-Learning for Signer-Independent Sign Language Translation”. In: *Proceedings of the 29th ACM International Conference on Multimedia*.
- Trevor Johnston (2012). “Lexical frequency in sign languages”. In: *Journal of deaf studies and deaf education*.
- Armand Joulin, Francis Bach, and Jean Ponce (2010). “Discriminative clustering for image co-segmentation”. In: *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE.
- Armand Joulin, Kevin Tang, and Li Fei-Fei (2014). “Efficient image and video co-localization with frank-wolfe algorithm”. In: *European Conference on Computer Vision*. Springer.
- Hamid Reza Vaezi Joze and Oscar Koller (2019). “MS-ASL: A Large-Scale Data Set and Benchmark for Understanding American Sign Language”. In: *BMVC*.
- Prajwal K R, Triantafyllos Afouras, Andrew Zisserman, et al. (2021). “Sub-word Level Lip Reading With Visual Attention”. In: *arXiv preprint arXiv:2110.07603*.

- Timor Kadir, Richard Bowden, Eng-Jon Ong, and Andrew Zisserman (2004). “Minimal Training, Large Lexicon, Unconstrained Sign Language Recognition”. In: *Proc. BMVC*.
- Yannis Kalantidis, Mert Bulent Sariyildiz, Noe Pion, Philippe Weinzaepfel, and Diane Larlus (2020). “Hard Negative Mixing for Contrastive Learning”. In: *Neurips*.
- Aishwarya Kamath, Mannat Singh, Yann LeCun, Ishan Misra, Gabriel Synnaeve, and Nicolas Carion (2021). “MDETR–Modulated Detection for End-to-End Multi-Modal Understanding”. In: *arXiv preprint arXiv:2104.12763*.
- Shigeki Karita, Nelson Yalta, Shinji Watanabe, M. Delcroix, A. Ogawa, and T. Nakatani (2019). “Improving Transformer-Based End-to-End Speech Recognition with Connectionist Temporal Classification and Language Model Integration”. In: *INTERSPEECH*.
- Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. (2017). “The kinetics human action video dataset”. In: *arXiv preprint arXiv:1705.06950*.
- Daniel Kelly, John Mc Donald, and Charles Markham (2010). “Weakly supervised training of a sign language recognition system using multiple instance learning density matrices”. In: *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*.
- Dahun Kim, Donghyeon Cho, and In So Kweon (2019). “Self-Supervised Video Representation Learning with Space-Time Cubic Puzzles”. In: *AAAI*.
- Gunhee Kim, Eric P Xing, Li Fei-Fei, and Takeo Kanade (2011). “Distributed cosegmentation via submodular optimization on anisotropic diffusion”. In: *2011 international conference on computer vision*. IEEE.
- Taehwan Kim, Jonathan Keane, Weiran Wang, Hao Tang, Jason Riggie, Gregory Shakhnarovich, Diane Brentari, and Karen Livescu (2017). “Lexicon-free fingerspelling recognition from video: Data, models, and signer adaptation”. In: *Computer Speech & Language*.
- Taejun Kim and Juhan Nam (2019). “Temporal Feedback Convolutional Recurrent Neural Networks for Keyword Spotting”. In: *arXiv:1911.01803*.
- Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick (2023). “Segment Anything”. In: *iccv*.

- Sang-Ki Ko, Chang Jo Kim, Hyedong Jung, and Choongsang Cho (2019a). “Neural Sign Language Translation Based on Human Keypoint Estimation”. In: *Applied Sciences*.
- Sang-Ki Ko, Chang Jo Kim, Hyedong Jung, and Choongsang Cho (2019b). “Neural Sign Language Translation based on Human Keypoint Estimation”. In: *Appl. Sci.*
- Philipp Koehn, Franz J. Och, and Daniel Marcu (2003). “Statistical Phrase-Based Translation”. In: *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*. URL: <https://aclanthology.org/N03-1017>.
- Oscar Koller (2020). “Quantitative survey of the state of the art in sign language recognition”. In: *arXiv preprint arXiv:2008.09918*.
- Oscar Koller, Jens Forster, and Hermann Ney (2015a). “Continuous sign language recognition: Towards large vocabulary statistical recognition systems handling multiple signers”. In: *Computer Vision and Image Understanding*.
- Oscar Koller, Jens Forster, and Hermann Ney (2015b). “Continuous sign language recognition: Towards large vocabulary statistical recognition systems handling multiple signers”. In: *Computer Vision and Image Understanding*.
- Oscar Koller, Hermann Ney, and Richard Bowden (2014a). “Read My Lips: Continuous Signer Independent Weakly Supervised Viseme Recognition”. In: *ECCV*.
- Oscar Koller, Hermann Ney, and Richard Bowden (2014b). “Weakly Supervised Automatic Transcription of Mouthings for Gloss-Based Sign Language Corpora”. In: *LREC Workshop on the Representation and Processing of Sign Languages: Beyond the Manual Channel*.
- Oscar Koller, Hermann Ney, and Richard Bowden (2015c). “Deep Learning of Mouth Shapes for Sign Language”. In: *Third Workshop on Assistive Computer Vision and Robotics, ICCV*.
- Oscar Koller, Hermann Ney, and Richard Bowden (2016). “Deep hand: How to train a cnn on 1 million hand images when your data is continuous and weakly labelled”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*.
- Oscar Koller, Sepehr Zargaran, and Hermann Ney (2017). “Re-Sign: Re-Aligned End-to-End Sequence Modelling with Deep Recurrent CNN-HMMs”. In: *CVPR*.
- Nitish Krishnamurthy and John Hansen (2009). “Babble Noise: Modeling, Analysis, and Applications”. In: *IEEE Audio, Speech, and Language Processing*.
- Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton (2012). “ImageNet Classification with Deep Convolutional Neural Networks”. In: *NeurIPS*.

- Hilde Kuehne, Hueihan Jhuang, Rainer Stiefelhagen, and Thomas Serre (2011). “HMDB: A large video database for human motion recognition”. In: *Proc. ICCV*.
- Dong-Hyun Lee et al. (2013). “Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks”. In: *Workshop on challenges in representation learning, ICML*.
- Jie Lei, Tamara L. Berg, and Mohit Bansal (2022). “Revealing Single Frame Bias for Video-and-Language Learning”. In: *arXiv preprint arXiv:2206.03428*.
- Jie Lei, Linjie Li, Luowei Zhou, Zhe Gan, Tamara L. Berg, Mohit Bansal, and Jingjing Liu (2021). “Less is More: ClipBERT for Video-and-Language Learning via Sparse Sampling”. In: *CVPR*.
- Chris Lengerich and Awni Hannun (2016). “An End-to-End Architecture for Keyword Spotting and Voice Activity Detection”. In: *NIPS 2016 End-to-End Learning for Speech and Audio Processing Workshop*.
- Dongxu Li, Cristian Rodriguez Opazo, Xin Yu, and Hongdong Li (2019). “Word-level Deep Sign Language Recognition from Video: A New Large-scale Dataset and Methods Comparison”. In: *WACV*.
- Dongxu Li, Chenchen Xu, Liu Liu, Yiran Zhong, Rong Wang, Lars Petersson, and Hongdong Li (2021). “Transcribing Natural Languages for The Deaf via Neural Editing Programs”. In: *arXiv preprint arXiv:2112.09600*.
- Dongxu Li, Chenchen Xu, Xin Yu, Kaihao Zhang, Ben Swift, Hanna Suominen, and Hongdong Li (2020a). “Tspnet: Hierarchical feature learning via temporal semantic pyramid for sign language translation”. In: *arXiv preprint arXiv:2010.05468*.
- Dongxu Li, Xin Yu, Chenchen Xu, Lars Petersson, and Hongdong Li (2020b). “Transferring cross-domain knowledge for video sign language recognition”. In: *CVPR*.
- Hui Li, Peng Wang, and Chunhua Shen (2017). “Towards end-to-end text spotting with convolutional recurrent neural networks”. In: *Proceedings of the IEEE international conference on computer vision*.
- Junnan Li, Dongxu Li, Caiming Xiong, and Steven C. H. Hoi (2022). “BLIP: Bootstrapping Language-Image Pre-training for Unified Vision-Language Understanding and Generation”. In: *ICML*.
- Manling Li, Ruochen Xu, Shuohang Wang, Luowei Zhou, Xudong Lin, Chenguang Zhu, Michael Zeng, Heng Ji, and Shih-Fu Chang (2022). “CLIP-Event: Connecting Text and Images with Event Structures”. In: *CVPR*.

- Wei Li, Sicheng Wang, Ming Lei, Sabato Marco Siniscalchi, and Chin-Hui Lee (2019). “Improving Audio-visual Speech Recognition Performance with Cross-modal Student-teacher Training”. In: *Proc. ICASSP*. IEEE.
- Hanwen Liang, Niamul Quader, Zhixiang Chi, Lizhe Chen, Peng Dai, Juwei Lu, and Yang Wang (2021). “Self-supervised Spatiotemporal Representation Learning by Exploiting Video Continuity”. In: *AAAI*.
- Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick (2014). “Microsoft coco: Common objects in context”. In: *ECCV*.
- Xudong Lin, Fabio Petroni, Gedas Bertasius, Marcus Rohrbach, Shih-Fu Chang, and Lorenzo Torresani (2022). “Learning To Recognize Procedural Activities with Distant Supervision”. In: *CVPR*.
- Ziyi Lin, Shijie Geng, Renrui Zhang, Peng Gao, Gerard de Melo, Xiaogang Wang, Jifeng Dai, Yu Qiao, and Hongsheng Li (2022). “Frozen CLIP Models are Efficient Video Learners”. In: *ECCV*.
- Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee (2023). “Visual Instruction Tuning”. In: *NeurIPS*.
- Meng Liu, Xiang Wang, Liqiang Nie, Xiangnan He, Baoquan Chen, and Tat-Seng Chua (2018). “Attentive moment retrieval in videos”. In: *ACM SIGIR*.
- Song Liu, Haoqi Fan, Shengsheng Qian, Yiru Chen, Wenkui Ding, and Zhongyuan Wang (2021). “Hit: Hierarchical transformer with momentum contrast for video-text retrieval”. In: *ICCV*.
- Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg (2016). “Ssd: Single shot multibox detector”. In: *ECCV*.
- Yang Liu, Samuel Albanie, Arsha Nagrani, and Andrew Zisserman (2019). “Use What You Have: Video Retrieval Using Representations From Collaborative Experts”. In: *Proc. BMVC*.
- Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu (2022). “Video Swin Transformer”. In: *CVPR*.
- Stephan Liwicki and Mark Everingham (2009). “Automatic recognition of fingerspelled words in british sign language”. In: *2009 IEEE computer society conference on computer vision and pattern recognition workshops*. IEEE.
- Huaishao Luo, Lei Ji, Ming Zhong, Yang Chen, Wen Lei, Nan Duan, and Tianrui Li (2022). “CLIP4Clip: An empirical study of CLIP for end to end video clip retrieval and captioning”. In: *Neurocomputing*.

- Ziyang Luo, Yadong Xi, Rongsheng Zhang, and Jing Ma (2022). “A Frustratingly Simple Approach for End-to-End Image Captioning”. In: *arXiv preprint arXiv:2201.12723*.
- Takaki Makino, Hank Liao, Yannis Assael, Brendan Shillingford, Basilio Garcia, Otavio Braga, and Olivier Siohan (2019). “Recurrent neural network transducer for audio-visual speech recognition”. In: *IEEE Workshop on Automatic Speech Recognition and Understanding*.
- Joanna Materzynska, Guillaume Berger, Ingo Bax, and Roland Memisevic (2019). “The Jester Dataset: A Large-Scale Video Dataset of Human Gestures”. In: *ICCVW*.
- Antoine Miech, Jean-Baptiste Alayrac, Lucas Smaira, Ivan Laptev, Josef Sivic, and Andrew Zisserman (2020). “End-to-end learning of visual representations from uncurated instructional videos”. In: *CVPR*.
- Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic (2019). “HowTo100M: Learning a Text-Video Embedding by Watching Hundred Million Narrated Video Clips”. In: *ICCV*.
- Ishan Misra, C. Lawrence Zitnick, and Martial Hebert (2016). “Shuffle and Learn: Unsupervised Learning Using Temporal Order Verification”. In: *ECCV*.
- Abdelrahman Mohamed, Dmytro Okhonko, and Luke Zettlemoyer (2019). “Transformers with convolutional context for ASR”. In: *arXiv preprint arXiv:1904.11660*.
- Liliane Momeni, Triantafyllos Afouras, Themis Stafylakis, Samuel Albanie, and Andrew Zisserman (2020a). “Seeing wake words: Audio-visual keyword spotting”. In: *BMVC*.
- Liliane Momeni, Hannah Bull, K R Prajwal, Samuel Albanie, Gül Varol, and Andrew Zisserman (2022). “Automatic dense annotation of large-vocabulary sign language videos”. In: *Proc. ECCV*.
- Liliane Momeni, Gül Varol, Samuel Albanie, Triantafyllos Afouras, and Andrew Zisserman (2020b). “Watch, Read and Lookup: Learning to Spot Signs from Multiple Supervisors”. In: *Proc. ACCV*.
- Mathew Monfort, Alex Andonian, Bolei Zhou, Kandan Ramakrishnan, Sarah Adel Bargal, Tom Yan, Lisa Brown, Quanfu Fan, Dan Gutfruehd, Carl Vondrick, et al. (2019). “Moments in Time Dataset: one million videos for event understanding”. In: *TPAMI*.
- Mathew Monfort, SouYoung Jin, Alexander Liu, David Harwath, Rogerio Feris, James Glass, and Aude Oliva (2021). “Spoken Moments: Learning Joint Audio-Visual Representations From Video Descriptions”. In: *CVPR*.

- Amit Moryossef, Kayo Yin, Graham Neubig, and Yoav Goldberg (2021). “Data Augmentation for Sign Language Gloss Translation”. In: *MTSUMMIT*.
- Saman Motamed, Danda Pani Paudel, and Luc Van Gool (2023). “Lego: Learning to Disentangle and Invert Concepts Beyond Object Appearance in Text-to-Image Diffusion Models”. In: *arXiv preprint arXiv:2311.13833*.
- Saeid Motiian, Quinn Jones, Seyed Mehdi Iranmanesh, and Gianfranco Doretto (2017). “Few-Shot Adversarial Domain Adaptation”. In: *NeurIPS*.
- Samuel Myer and Vikrant Singh Tomar (2018). “Efficient keyword spotting using time delay neural networks”. In: *INTERSPEECH*.
- Arsha Nagrani, Chen Sun, David Ross, Rahul Sukthankar, Cordelia Schmid, and Andrew Zisserman (2020). “Speech2action: Cross-modal supervision for action recognition”. In: *CVPR*.
- Minh Hoai Nguyen, Lorenzo Torresani, Fernando De La Torre, and Carsten Rother (2009). “Weakly supervised discriminative localization and classification: a joint learning process”. In: *2009 IEEE 12th International Conference on Computer Vision*. IEEE.
- Tan Dat Nguyen and Surendra Ranganath (2008). “Tracking facial features under occlusions and recognizing facial expressions in sign language”. In: *International Conference on Automatic Face and Gesture Recognition*.
- Mehdi Noroozi and Paolo Favaro (2016). “Unsupervised learning of visual representations by solving jigsaw puzzles”. In: *Proc. ECCV*. Springer.
- Eng-Jon Ong, Helen Cooper, Nicolas Pugeault, and Richard Bowden (2012). “Sign Language Recognition using Sequential Pattern Trees”. In: *CVPR*.
- Eng-Jon Ong, Oscar Koller, Nicolas Pugeault, and Richard Bowden (2014). “Sign spotting using hierarchical sequential patterns with temporal intervals”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*.
- Aaron van den Oord, Yazhe Li, and Oriol Vinyals (2018). “Representation learning with contrastive predictive coding”. In: *arXiv preprint arXiv:1807.03748*.
- David Ouyang, Bryan He, Amirata Ghorbani, Neal Yuan, Joseph Ebinger, Curt P. Langlotz, Paul A. Heidenreich, Robert A. Harrington, David H. Liang, Euan A. Ashley, and James Y. Zou. (2020). “Video-based AI for beat-to-beat assessment of cardiac function”. In: *Nature*.
- Carol Padden and Darline Clark Gunsauls (2003). “How the Alphabet Came to Be Used in a Sign Language”. In: *Sign Language Studies*.
- Dimitri Palaz, Gabriel Synnaeve, and Ronan Collobert (2016). “Jointly learning to locate and classify words using convolutional networks”. In: *INTERSPEECH*.

- George Papandreou, Athanassios Katsamanis, Vassilis Pitsikalis, and Petros Maragos (2009). “Adaptive multimodal fusion by uncertainty compensation with application to audiovisual speech recognition”. In: *Audio, Speech, and Language Processing, IEEE Transactions on*.
- Jae Sung Park, Sheng Shen, Ali Farhadi, Trevor Darrell, Yejin Choi, and Anna Rohrbach (2022). “Exposing the Limits of Video-Text Models through Contrast Sets”. In: *ACL*.
- Jeffrey Pennington, Richard Socher, and Christopher Manning (2014). “GloVe: Global Vectors for Word Representation”. In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.
- Stavros Petridis, Themis Stafylakis, Pingchuan Ma, Georgios Tzimiropoulos, and Maja Pantic (2018). “Audio-Visual Speech Recognition with a Hybrid CTC/Attention Architecture”. In: *IEEE Spoken Language Technology Workshop (SLT)*.
- Tomas Pfister, James Charles, and Andrew Zisserman (2013). “Large-scale Learning of Sign Language by Watching TV (Using Co-occurrences)”. In: *BMVC*.
- Tomas Pfister, James Charles, and Andrew Zisserman (2014). “Domain-adaptive Discriminative One-shot Learning of Gestures”. In: *Proc. ECCV*.
- Lyndsey C. Pickup, Zheng Pan, Donglai Wei, YiChang Shih, Changshui Zhang, Andrew Zisserman, Bernhard Scholkopf, and William T. Freeman (2014). “Seeing the Arrow of Time”. In: *CVPR*.
- Jesús Andrés Portillo-Quintero, José Carlos Ortiz-Bayliss, and Hugo Terashima-Marín (2021). “A Straightforward Framework For Video Retrieval Using CLIP”. In: *arXiv preprint arXiv:2102.12443*.
- Gerasimos Potamianos, Chalapathy Neti, Guillaume Gravier, Ashutosh Garg, and Andrew W Senior (2003). “Recent advances in the automatic recognition of audiovisual speech”. In: *Proceedings of the IEEE*.
- K R Prajwal, Triantafyllos Afouras, and Andrew Zisserman (2022a). “Sub-word Level Lip Reading With Visual Attention”. In: *Proc. CVPR*.
- K R Prajwal, Hannah Bull, Liliane Momeni, Samuel Albanie, Gül Varol, and Andrew Zisserman (2022b). “Weakly-supervised Fingerspelling Recognition in British Sign Language Videos”. In: *Proc. BMVC*.
- KR Prajwal, Liliane Momeni, Triantafyllos Afouras, and Andrew Zisserman (2021). “Visual Keyword Spotting with Attention”. In: *BMVC*.
- Will Price and Dima Damen (2019). “Retro-Actions: Learning ‘Close’ by Time-Reversing ‘Open’ Videos”. In: *ICCVW*.

- Nicolas Pugeault and Richard Bowden (2011). “Spelling it out: Real-time ASL fingerspelling recognition”. In: *2011 IEEE International conference on computer vision workshops (ICCV workshops)*. IEEE.
- Filip Radenovic, Abhimanyu Dubey, Abhishek Kadian, Todor Mihaylov, Simon Vandenhende, Yash Patel, Yi Wen, Vignesh Ramanathan, and Dhruv Mahajan (2023). “Filtering, Distillation, and Hard Negatives for Vision-Language Pre-Training”. In: *arXiv preprint arXiv:2301.02280*.
- 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: *ICML*.
- Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever (2019). “Language models are unsupervised multitask learners”. In: *OpenAI blog*.
- 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”. In: *The Journal of Machine Learning Research*.
- Adria Recasens, Pauline Luc, Jean-Baptiste Alayrac, Luyu Wang, Florian Strub, Corentin Tallec, Mateusz Malinowski, Viorica Patraauean, Florent Altché, Michal Valko, Jean-Bastien Grill, Aaron Oord, and Andrew Zisserman (2021). “Broaden Your Views for Self-Supervised Video Learning”. In: *arXiv preprint arXiv:2021.00129*.
- Machel Reid, Nikolay Savinov, Denis Teplyashin, Dmitry Lepikhin, and more (2024). “Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context”. In: *arXiv preprint arXiv:2403.05530*.
- Katrin Renz, Nicolaj Stache, Samuel Albanie, and Gül Varol (2021a). “Sign Segmentation with Temporal Convolutional Networks”. In: *International Conference on Acoustics, Speech, and Signal Processing*.
- Katrin Renz, Nicolaj Stache, Neil Fox, Gül Varol, and Samuel Albanie (2021b). “Sign Segmentation with Change-point-Modulated Pseudo-Labeling”. In: *Workshop on ChaLearn Looking at People Sign Language Recognition in the Wild, CVPR*. IEEE.
- Rami Al-Rfou, Dokook Choe, Noah Constant, Mandy Guo, and Llion Jones (2019). “Character-level language modeling with deeper self-attention”. In: *Proceedings of the AAAI Conference on Artificial Intelligence*.

- Susanna Ricco and Carlo Tomasi (2009). “Fingerspelling recognition through classification of letter-to-letter transitions”. In: *Asian conference on computer vision*. Springer.
- Joshua Robinson, Ching-Yao Chuang, Suvrit Sra, and Stefanie Jegelka (2021). “Contrastive Learning with Hard Negative Samples”. In: *ICLR*.
- Jason Rodolitz, Evan Gambill, Brittany Willis, Christian Vogler, and Raja Kushalnagar (2019). “Accessibility of voice-activated agents for people who are deaf or hard of hearing”. In: *The Journal On Technology and Persons with Disabilities*.
- Anna Rohrbach, Atousa Torabi, Marcus Rohrbach, Niket Tandon, Chris Pal, Hugo Larochelle, Aaron Courville, and Bernt Schiele (2017). “Movie Description”. In: *IJCV*.
- Richard C Rose and Douglas B Paul (1990). “A hidden Markov model based keyword recognition system”. In: *International Conference on Acoustics, Speech, and Signal Processing*. IEEE.
- Andrew Rosenberg, Kartik Audhkhasi, Abhinav Sethy, Bhuvana Ramabhadran, and Michael Picheny (2017). “End-to-end speech recognition and keyword search on low-resource languages”. In: *ICASSP*.
- Carsten Rother, Tom Minka, Andrew Blake, and Vladimir Kolmogorov (2006). “Cosegmentation of image pairs by histogram matching-incorporating a global constraint into mrfs”. In: *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’06)*. IEEE.
- Michael Rubinstein, Armand Joulin, Johannes Kopf, and Ce Liu (2013). “Unsupervised joint object discovery and segmentation in internet images”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*.
- Darshan Singh S, Zeeshan Khan, and Makarand Tapaswi (2024). “FiGCLIP: Fine-Grained CLIP Adaptation via Densely Annotated Videos”. In: *arXiv preprint arXiv:2401.07669*.
- Arka Sadhu, Tanmay Gupta, Mark Yatskar, Ram Nevatia, and Aniruddha Kembhavi (2021). “Visual Semantic Role Labeling for Video Understanding”. In: *CVPR*.
- Ugur Sahin, Hang Li, Qadeer Khan, Daniel Cremers, and Volker Tresp (2023). “Enhancing Multimodal Compositional Reasoning of Visual Language Models with Generative Negative Mining”. In: *Proc. WACV*.
- Tara N. Sainath and Carolina Parada (2015). “Convolutional neural networks for small-footprint keyword spotting”. In: *INTERSPEECH*.

- Hiroaki Sakoe and Seibi Chiba (1978). “Dynamic programming algorithm optimization for spoken word recognition”. In: *IEEE TRANSACTIONS ON ACOUSTICS, SPEECH, AND SIGNAL PROCESSING*.
- Shibani Santurkar, Yann Dubois, Rohan Taori, Percy Liang, and Tatsunori Hashimoto (2022). “Is a Caption Worth a Thousand Images? A Controlled Study for Representation Learning”. In: *arXiv preprint arXiv:2207.07635*.
- Adam Schembri, Jordan Fenlon, Ramas Rentelis, and Kearsy Cormier (2017). *British Sign Language Corpus Project: A corpus of digital video data and annotations of British Sign Language 2008-2017 (Third Edition)*. URL: <http://www.bslcorpusproject.org>.
- Adam Schembri, Jordan Fenlon, Ramas Rentelis, Sally Reynolds, and Kearsy Cormier (2013). “Building the British sign language corpus”. In: *Language Documentation & Conservation*.
- Adam Schembri and Trevor Johnston (2007). “Sociolinguistic variation in the use of fingerspelling in Australian Sign Language: A pilot study”. In: *Sign Language Studies*.
- Konrad Schindler and Luc van Gool (2008). “Action snippets: How many frames does human action recognition require?” In: *CVPR*.
- Mike Schuster and K.K. Paliwal (Nov. 1997). “Bidirectional Recurrent Neural Networks”. In: *Trans. Sig. Proc.*
- Arda Senocak, Tae-Hyun Oh, Junsik Kim, Ming-Hsuan Yang, and In So Kweon (2018). “Learning to Localize Sound Source in Visual Scenes”. In: *CVPR*.
- Paul Hongsuck Seo, Arsha Nagrani, Anurag Arnab, and Cordelia Schmid (2022). “End-to-end Generative Pretraining for Multimodal Video Captioning”. In: *CVPR*.
- Laura Sevilla-Lara, Shengxin Zha, Zhicheng Yan, Vedanuj Goswami, Matt Feiszli, and Lorenzo Torresani (2021). “Only Time Can Tell: Discovering Temporal Data for Temporal Modeling”. In: *WACV*.
- Changhao Shan, Junbo Zhang, Yujun Wang, and Lei Xie (2018). “Attention-based end-to-end models for small-footprint keyword spotting”. In: *arXiv preprint arXiv:1803.10916*.
- Xiaolong Shen and Zhedong Zheng and Yi Yang (2022). “StepNet: Spatial-temporal Part-aware Network for Isolated Sign Language Recognition”. In.
- Bowen Shi, Diane Brentari, Greg Shakhnarovich, and Karen Livescu (2021). “Fingerspelling Detection in American Sign Language”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

- Bowen Shi, Diane Brentari, Greg Shakhnarovich, and Karen Livescu (2022a). “Open-Domain Sign Language Translation Learned from Online Video”. In: *EMNLP*.
- Bowen Shi, Diane Brentari, Greg Shakhnarovich, and Karen Livescu (2022b). “Searching for fingerspelled content in American Sign Language”. In: *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.
- Bowen Shi, Aurora Martinez Del Rio, Jonathan Keane, Jonathan Michaux, Diane Brentari, Greg Shakhnarovich, and Karen Livescu (2018). “American sign language fingerspelling recognition in the wild”. In: *2018 IEEE Spoken Language Technology Workshop (SLT)*. IEEE.
- Bowen Shi and Karen Livescu (2017). “Multitask training with unlabeled data for end-to-end sign language fingerspelling recognition”. In: *2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE.
- 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: *Proceedings of the IEEE/CVF International Conference on Computer Vision*.
- Zhiyuan Shi, Timothy M Hospedales, and Tao Xiang (2013). “Bayesian joint topic modelling for weakly supervised object localisation”. In: *Proceedings of the IEEE International Conference on Computer Vision*.
- Brendan Shillingford, Yannis Assael, Matthew W. Hoffman, Thomas Paine, Cían Hughes, Utsav Prabhu, Hank Liao, Hasim Sak, Kanishka Rao, Lorraine Bennett, Marie Mulville, Ben Coppin, Ben Laurie, Andrew Senior, and Nando de Freitas (2018). “Large-Scale Visual Speech Recognition”. In: *arXiv preprint arXiv:1807.05162*.
- Zheng Shou, Dongang Wang, and Shih-Fu Chang (2016). “Temporal action localization in untrimmed videos via multi-stage CNNs”. In: *CVPR*.
- David SignumMcKee and Graeme Kennedy (2000). “Lexical comparison of signs from American, Australian, British and New Zealand sign languages.” In: *The signs of language revisited: An anthology to honor Ursula Bellugi and Edward Klima*.
- Gunnar Sigurdsson, Olga Russakovsky, and Abhinav Gupta (2017). “What Actions are Needed for Understanding Human Actions in Videos?” In: *ICCV*.
- Ozge Mercanoglu Sincan, Necati Cihan Camgoz, and Richard Bowden (2023). “Is context all you need? Scaling Neural Sign Language Translation to Large Domains of Discourse”. In: *ICCV, ACVR workshop*.

- Ozge Mercanoglu Sincan, Necati Cihan Camgoz, and Richard Bowden (2024). “Using an LLM to Turn Sign Spottings into Spoken Language Sentences”. In: *arXiv preprint arXiv:2403.10434*.
- Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah (2012). “UCF101: A Dataset of 101 Human Actions Classes From Videos in The Wild”. In: *CoRR*.
- Ben Sorscher, Robert Geirhos, Shashank Shekhar, Surya Ganguli, and Ari S. Morcos (2022). “Beyond neural scaling laws: beating power law scaling via data pruning”. In: *NeurIPS*.
- Speech Group at Carnegie Mellon University (2014). *CMU Pronouncing Dictionary*. <http://www.speech.cs.cmu.edu/cgi-bin/cmudict>.
- Themos Stafylakis and Georgios Tzimiropoulos (2017). “Combining Residual Networks with LSTMs for Lipreading”. In: *Interspeech*.
- Themos Stafylakis and Georgios Tzimiropoulos (2018). “Zero-shot keyword spotting for visual speech recognition in-the-wild”. In: *ECCV*.
- Thad Starner (1995). “Visual Recognition of American Sign Language Using Hidden Markov Models”. MA thesis. Massachusetts Institute of Technology.
- Christopher Stone and Debra Russell (2013). “Interpreting in International Sign: Decisions of Deaf and non-deaf interpreters”. In.
- Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid (2019). “Videobert: A joint model for video and language representation learning”. In: *ICCV*.
- Ming Sun, Anirudh Raju, George Tucker, Sankaran Panchapagesan, Gengshen Fu, Arindam Mandal, Spyridon Matsoukas, Nikko Strom, and Shiv Vitaladevuni (2016). “Max-pooling loss training of long short-term memory networks for small-footprint keyword spotting”. In: *2016 IEEE Spoken Language Technology Workshop (SLT)*.
- Ming Sun, David Snyder, Yixin Gao, Varun Nagaraja, Mike Rodehorst, Sankaran Panchapagesan, Nikko Strom, Spyros Matsoukas, and Shiv Vitaladevuni (2017). “Compressed Time Delay Neural Network for Small-Footprint Keyword Spotting”. In: *INTERSPEECH*.
- Yuchong Sun, Hongwei Xue, Ruihua Song, Bei Liu, Huan Yang, and Jianlong Fu (2022). “Long-Form Video-Language Pre-Training with Multimodal Temporal Contrastive Learning”. In: *Neurips*.
- Valerie Sutton (1990). *Lessons in sign writing*. SignWriting.
- Rachel Sutton-Spence (2007). “Mouthings and Simultaneity in British Sign Language”. In: *Simultaneity in Signed Languages: Form and Function*. John Benjamins.

- Rachel Sutton-Spence and Bencie Woll (1999). *The Linguistics of British Sign Language: An Introduction*. Cambridge University Press.
- Igor Szoke, Petr Schwarz, Pavel Matejka, Lukás Burget, Martin Karafiát, Michal Fapso, and Jan Cernocky (2005). “Comparison of keyword spotting approaches for informal continuous speech”. In: *Ninth European conference on speech communication and technology*.
- Shinichi Tamura and Shingo Kawasaki (1988). “Recognition of sign language motion images”. In: *Pattern recognition*.
- Kevin Tang, Armand Joulin, Li-Jia Li, and Li Fei-Fei (2014). “Co-localization in real-world images”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*.
- Zhan Tong, Yibing Song, Jue Wang, and Limin Wang (2022). “VideoMAE: Masked Autoencoders are Data-Efficient Learners for Self-Supervised Video Pre-Training”. In: *NeurIPS*.
- Maria Tsimpoukelli, Jacob Menick, Serkan Cabi, SM Eslami, Oriol Vinyals, and Felix Hill (2021). “Multimodal few-shot learning with frozen language models”. In: *Neurips*.
- George Tucker, Minhua Wu, Ming Sun, Sankaran Panchapagesan, Gengshen Fu, and Shiv Vitaladevuni (2016). “Model Compression Applied to Small-Footprint Keyword Spotting”. In: *INTERSPEECH*.
- Clayton Valli and Gallaudet University (2005). *The Gallaudet Dictionary of American Sign Language*. Gallaudet University Press.
- Aäron van den Oord, Yazhe Li, and Oriol Vinyals (2018). “Representation Learning with Contrastive Predictive Coding”. In: *arXiv preprint arXiv:1807.03748*.
- Gül Varol, Liliane Momeni, Samuel Albanie, Triantafyllos Afouras, and Andrew Zisserman (2021). “Read and Attend: Temporal Localisation in Sign Language Videos”. In: *Proc. CVPR*.
- 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: *NeurIPS*.
- Tom Véniat, Olivier Schwander, and Ludovic Denoyer (2019). “Stochastic adaptive neural architecture search for keyword spotting”. In: *ICASSP*.
- Ville Viitaniemi, Tommi Jantunen, Leena Savolainen, Matti Karppa, and Jorma Laaksonen (2014). “S-pot – a benchmark in spotting signs within continuous signing”. In: *LREC*.

- Van Huy Vo, Elena Sizikova, Cordelia Schmid, Patrick Pérez, and Jean Ponce (2021). “Large-scale unsupervised object discovery”. In: *Advances in Neural Information Processing Systems*.
- Christian Vogler and Dimitris Metaxas (1997). “Adapting hidden Markov models for ASL recognition by using three-dimensional computer vision methods”. In: *1997 IEEE International Conference on Systems, Man, and Cybernetics. Computational Cybernetics and Simulation*. IEEE.
- Christian Vogler and Dimitris Metaxas (1998). “ASL recognition based on a coupling between HMMs and 3D motion analysis”. In: *Sixth International Conference on Computer Vision (IEEE Cat. No. 98CH36271)*. IEEE.
- Christian Vogler and Dimitris Metaxas (2001). “A framework for recognizing the simultaneous aspects of American Sign Language”. In: *Computer Vision and Image Understanding*.
- Christian Vogler and Dimitris Metaxas (2003). “Handshapes and movements: Multiple-channel american sign language recognition”. In: *International Gesture Workshop*. Springer.
- Ulrich von Agris, Moritz Knorr, and Karl-Friedrich Kraiss (2008). “The significance of facial features for automatic sign language recognition”. In: *2008 8th IEEE International Conference on Automatic Face Gesture Recognition*.
- Chong Wang, Weiqiang Ren, Kaiqi Huang, and Tieniu Tan (2014). “Weakly supervised object localization with latent category learning”. In: *European Conference on Computer Vision*. Springer.
- Jiangliu Wang, Jianbo Jiao, and Yunhui Liu (2020). “Self-Supervised Video Representation Learning by Pace Prediction”. In: *Proc. ECCV*.
- Jue Wang, Gedas Bertasius, Du Tran, and Lorenzo Torresani (2022). “Long-Short Temporal Contrastive Learning of Video Transformers”. In: *CVPR*.
- Mengmeng Wang, Jiazheng Xing, and Yong Liu (2021). “ActionCLIP: A New Paradigm for Video Action Recognition”. In: *arXiv preprint arXiv:2109.08472*.
- Yi Wang, Kunchang Li, Yizhuo Li, Yinan He, Bingkun Huang, Zhiyu Zhao, Hongjie Zhang, Jilan Xu, Yi Liu, Zun Wang, Sen Xing, Guo Chen, Junting Pan, Jiashuo Yu, Yali Wang, Limin Wang, and Yu Qiao (2022). “InternVideo: General Video Foundation Models via Generative and Discriminative Learning”. In: *arXiv preprint arXiv:2212.03191*.
- Yuxuan Wang, Pascal Getreuer, Thad Hughes, Richard Lyon, and Rif Saurous (2017). “Trainable frontend for robust and far-field keyword spotting”. In: *ICASSP*.

- Zhe Wang, Petar Velickovic, Daniel Hennes, Nenad Tomasev, Laurel Prince, Michael Kaisers, Yoram Bachrach, Romuald Élie, Wenliang Kevin Li, Federico Piccinini, William Spearman, Ian Graham, Jerome T. Connor, Yi Yang, Adrià Recasens, Mina Khan, Nathalie Beauguerlange, Pablo Sprechmann, Pol Moreno, Nicolas Manfred Otto Heess, Michael Bowling, Demis Hassabis, and Karl Tuyls (2023). “TacticAI: an AI assistant for football tactics”. In: *Nature Communications*.
- Zhenhailong Wang, Ansel Blume, Sha Li, Genglin Liu, Jaemin Cho, Zineng Tang, Mohit Bansal, and Heng Ji (2023). “Paxion: Patching Action Knowledge in Video-Language Foundation Models”. In: *NeurIPS*.
- Zhenhailong Wang, Manling Li, Ruochen Xu, Luowei Zhou, Jie Lei, Xudong Lin, Shuohang Wang, Ziyi Yang, Chenguang Zhu, Derek Hoiem, et al. (2022). “Language Models with Image Descriptors are Strong Few-Shot Video-Language Learners”. In: *arXiv preprint arXiv:2205.10747*.
- Donglai Wei, Joseph J Lim, Andrew Zisserman, and William T Freeman (2018). “Learning and Using the Arrow of Time”. In: *CVPR*.
- Fangyun Wei and Yutong Chen (2023). “Improving Continuous Sign Language Recognition with Cross-Lingual Signs”. In: *ICCV*.
- Ronnie B. Wilbur (2000). “Phonological and prosodic layering of nonmanuals in American Sign Language.” In: *The Signs of Language Revisited: Festschrift for Ursula Bellugi and Edward Klima*.
- Ronnie B. Wilbur and Avinash C. Kak (2006). “Purdue RVL-SLLL American Sign Language Database”. In: *School of Electrical and Computer Engineering Technical Report, TR-06-12, Purdue University, W. Lafayette, IN 47906*.
- Ronald J Williams and David Zipser (1989). “A learning algorithm for continually running fully recurrent neural networks”. In: *Neural computation*.
- Jay Wilpon, Chin-Hui Lee, and Lawrence Rabiner (June 1989). “Application of hidden Markov models for recognition of a limited set of words in unconstrained speech”. In: *International Conference on Acoustics, Speech, and Signal Processing*.
- Gabriella Wojtanowski, Colleen Gilmore, Barbra Seravalli, Kristen Fargas, Christian Vogler, and Raja S. Kushalnagar (2020). “Alexa, can you see me?” making individual personal assistants for the home accessible to deaf consumers”. In: *Journal on Technology and Persons with Disabilities*.
- Bencie Woll (2001). “The sign that dares to speak its name: Echo phonology in British Sign Language (BSL)”. In: *The hands are the head of the mouth: The mouth as*

- articulator in sign languages*. Ed. by Penny Boyes-Braem and Rachel Sutton-Spence. Hamburg: Signum Press.
- Ryan Wong, Necati Cihan Camgoz, and Richard Bowden (2023). “Learnt Contrastive Concept Embeddings for Sign Recognition”. In: *Proc. ICCV*.
- Michael Wray, G. Csurka, Diane Larlus, and Dima Damen (2019). “Fine-Grained Action Retrieval Through Multiple Parts-of-Speech Embeddings”. In: *ICCV*.
- Michael Wray and Dima Damen (2019). “Learning Visual Actions Using Multiple Verb-Only Labels”. In: *BMVC*.
- Bichen Wu, Chenfeng Xu, Xiaoliang Dai, Alvin Wan, Peizhao Zhang, Masayoshi Tomizuka, Kurt Keutzer, and Peter Vajda (2020). “Visual Transformers: Token-based Image Representation and Processing for Computer Vision”. In: *CoRR*.
- Chao-Yuan Wu, Christoph Feichtenhofer, Haoqi Fan, Kaiming He, Philipp Krähenbühl, and Ross Girshick (2019). “Long-Term Feature Banks for Detailed Video Understanding”. In: *CVPR*.
- Chao-Yuan Wu, R. Manmatha, Alexander J. Smola, and Philipp Krähenbühl (2017). “Sampling Matters in Deep Embedding Learning”. In: *ICCV*.
- Pingping Wu, Hong Liu, Xiaofei Li, Ting Fan, and Xuewu Zhang (2016). “A Novel Lip Descriptor for Audio-Visual Keyword Spotting Based on Adaptive Decision Fusion”. In: *IEEE Transactions on Multimedia*.
- Junbin Xiao, Xindi Shang, Angela Yao, and Tat-Seng Chua (2021). “NExT-QA: Next Phase of Question-Answering to Explaining Temporal Actions”. In: *CVPR*.
- Junbin Xiao, Pan Zhou, Tat-Seng Chua, and Shuicheng Yan (2022). “Video Graph Transformer for Video Question Answering”. In: *ECCV*.
- Dejing Xu, Jun Xiao, Zhou Zhao, Jian Shao, Di Xie, and Yueting Zhuang (2019). “Self-supervised spatiotemporal learning via video clip order prediction”. In: *Proc. CVPR*.
- Hu Xu, Gargi Ghosh, Po-Yao Huang, Dmytro Okhonko, Armen Aghajanyan, Florian Metze, Luke Zettlemoyer, and Christoph Feichtenhofer (2021). “VideoCLIP: Contrastive Pre-training for Zero-shot Video-Text Understanding”. In: *EMNLP*.
- Huijuan Xu, Kun He, Bryan A. Plummer, Leonid Sigal, Stan Sclaroff, and Kate Saenko (2019). “Multilevel Language and Vision Integration for Text-to-Clip Retrieval.” In: *AAAI*.
- Jun Xu, Tao Mei, Ting Yao, and Yong Rui (2016). “MSR-VTT: A Large Video Description Dataset for Bridging Video and Language.” In: *Proc. CVPR*.

- Ran Xu, Caiming Xiong, Wei Chen, and Jason J. Corso (2015). “Jointly Modeling Deep Video and Compositional Text to Bridge Vision and Language in a Unified Framework”. In: *AAAI*.
- Shen Yan, Xuehan Xiong, Anurag Arnab, Zhichao Lu, Mi Zhang, Chen Sun, and Cordelia Schmid (2022). “Multiview transformers for video recognition”. In: *CVPR*.
- Antoine Yang, Antoine Miech, Josef Sivic, Ivan Laptev, and Cordelia Schmid (2021). “Just ask: Learning to answer questions from millions of narrated videos”. In: *ICCV*.
- Antoine Yang, Antoine Miech, Josef Sivic, Ivan Laptev, and Cordelia Schmid (2022). “Zero-Shot Video Question Answering via Frozen Bidirectional Language Models”. In: *Neurips*.
- Hee-Deok Yang, Stan Sclaroff, and Seong-Whan Lee (2008). “Sign language spotting with a threshold model based on conditional random fields”. In: *IEEE transactions on pattern analysis and machine intelligence*.
- Hongtao Yang, Xuming He, and Fatih Porikli (2018). “One-Shot Action Localization by Learning Sequence Matching Network”. In: *CVPR*.
- Jianwei Yang, Yonatan Bisk, and Jianfeng Gao (2021). “TACo: Token-aware Cascade Contrastive Learning for Video-Text Alignment”. In: *ICCV*.
- Ming-Hsuan Yang, Narendra Ahuja, and Mark Tabb (2002). “Extraction of 2d motion trajectories and its application to hand gesture recognition”. In: *IEEE Transactions on pattern analysis and machine intelligence*.
- Shuo Yang, Ping Luo, Chen-Change Loy, and Xiaoou Tang (2016). “Wider face: A face detection benchmark”. In: *CVPR*.
- Zhengyuan Yang, Zhe Gan, Jianfeng Wang, Xiaowei Hu, Yumao Lu, Zicheng Liu, and Lijuan Wang (2022). “An Empirical Study of GPT-3 for Few-Shot Knowledge-Based VQA”. In: *AAAI*.
- Polina Yanovich, Carol Neidle, and Dimitris Metaxas (2016). “Detection of major ASL sign types in continuous signing for ASL recognition”. In: *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC’16)*.
- Yuan Yao, Chang Liu, Dezhao Luo, Yu Zhou, and Qixiang Ye (2020). “Video Playback Rate Perception for Self-Supervised Spatio-Temporal Representation Learning”. In: *CVPR*.
- Yue Yao, Tianyu Wang, Heming Du, Liang Zheng, and Tom Gedeon (2019). “Spotting Visual Keywords from Temporal Sliding Windows”. In: *Mandarin Audio-Visual Speech Recognition Challenge*.

- David Yarowsky (1995). “Unsupervised word sense disambiguation rivaling supervised methods”. In: *33rd annual meeting of the association for computational linguistics*.
- Yuancheng Ye, Yingli Tian, Matt Huenerfauth, and Jingya Liu (2018). “Recognizing American Sign Language Gestures from Within Continuous Videos”. In: *CVPRW*.
- Aoxiong Yin, Tianyun Zhong, Li Tang, Weike Jin, Tao Jin, and Zhou Zhao (2023). “Gloss attention for gloss-free sign language translation”. In: *CVPR*.
- Kayo Yin and Jesse Read (2020). “Better Sign Language Translation with STMC-Transformer”. In: *COLING*.
- Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu (2022). “Coca: Contrastive captioners are image-text foundation models”. In: *arXiv*.
- Jianwei Yu, Shi-Xiong Zhang, Jian Wu, Shahram Ghorbani, Bo Wu, Shiyin Kang, Shansong Liu, Xunying Liu, Helen Meng, and Dong Yu (May 2020). “Audio-Visual Recognition of Overlapped Speech for the LRS2 Dataset”. In: *ICASSP*.
- Licheng Yu, Zhe Lin, Xiaohui Shen, Jimei Yang, Xin Lu, Mohit Bansal, and Tamara L Berg (2018). “Mattnet: Modular attention network for referring expression comprehension”. In: *CVPR*.
- Youngjae Yu, Jongseok Kim, and Gunhee Kim (2018). “A joint sequence fusion model for video question answering and retrieval”. In: *ECCV*.
- Yitian Yuan, Tao Mei, and Wenwu Zhu (2019). “To find where you talk: Temporal sentence localization in video with attention based location regression”. In: *AAAI*.
- Mert Yuksekgonul, Federico Bianchi, Pratyusha Kalluri, Dan Jurafsky, and James Zou (2023). “When and why Vision-Language Models behave like Bags-of-Words, and what to do about it?” In: *ICLR*.
- Rowan Zellers, Jiasen Lu, Ximing Lu, Youngjae Yu, Yanpeng Zhao, Mohammadreza Salehi, Aditya Kusupati, Jack Hessel, Ali Farhadi, and Yejin Choi (2022). “MERLOT Reserve: Multimodal Neural Script Knowledge through Vision and Language and Sound”. In: *CVPR*.
- Andy Zeng, Maria Attarian, brian ichter, Krzysztof Marcin Choromanski, Adrian Wong, Stefan Welker, Federico Tombari, Aveek Purohit, Michael S Ryoo, Vikas Sindhwani, Johnny Lee, Vincent Vanhoucke, and Pete Florence (2023). “Socratic Models: Composing Zero-Shot Multimodal Reasoning with Language”. In: *ICLR*.
- Runhao Zeng, H. Xu, W. Huang, Peihao Chen, Mingkui Tan, and Chuang Gan (2020). “Dense Regression Network for Video Grounding”. In: *CVPR*.

- Bowen Zhang, Hexiang Hu, and Fei Sha (2018). “Cross-Modal and Hierarchical Modeling of Video and Text”. In: *ECCV*.
- Gengyuan Zhang, Jinhe Bi, Jindong Gu, Yanyu Chen, and Volker Tresp (2023). “SPOT! Revisiting Video-Language Models for Event Understanding”. In: *arXiv preprint arXiv:2311.12919*.
- Haitong Zhang, Junbo Zhang, and Yujun Wang (2018). “Sequence-to-sequence Models for Small-Footprint Keyword Spotting”. In: *arXiv:1811.00348*.
- Huaiwen Zhang, Zihang Guo, Yang Yang, Xin Liu, and De Hu (2023). “C2ST: Cross-Modal Contextualized Sequence Transduction for Continuous Sign Language Recognition”. In: *ICCV*.
- Jihai Zhang, Wengang Zhou, and Houqiang Li (2014). “A threshold-based hmm-dtw approach for continuous sign language recognition”. In: *Proceedings of International Conference on Internet Multimedia Computing and Service*.
- Junyi Zhang, Ziliang Chen, Junying Huang, Liang Lin, and Dongyu Zhang (2019). “Few-Shot Structured Domain Adaptation for Virtual-to-Real Scene Parsing”. In: *ICCVW*.
- Shilin Zhang and Bo Zhang (2010). “Using revised string edit distance to sign language video retrieval”. In: *2010 Second International Conference on Computational Intelligence and Natural Computing*. IEEE.
- Xingxuan Zhang, Feng Cheng, and Shilin Wang (2019). “Spatio-Temporal Fusion based Convolutional Sequence Learning for Lip Reading”. In: *Proc. ICCV*.
- Yaodong Zhang and James Glass (Jan. 2010). “Unsupervised spoken keyword spotting via segmental DTW on Gaussian Posteriorgrams”. In: *Proceedings of the 2009 IEEE Workshop on Automatic Speech Recognition and Understanding*.
- Yundong Zhang, Naveen Suda, Liangzhen Lai, and Vikas Chandra (2017). “Hello Edge: Keyword Spotting on Microcontrollers”. In: *CoRR*.
- Hang Zhao, Antonio Torralba, Lorenzo Torresani, and Zhicheng Yan (2019). “Hacs: Human action clips and segments dataset for recognition and temporal localization”. In: *ICCV*.
- Yue Zhao, Ishan Misra, Philipp Krähenbühl, and Rohit Girdhar (2022). “Learning Video Representations from Large Language Models”. In: *arXiv preprint arXiv:2212.04501*.
- 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: *ICCV*.

- Bolei Zhou, Alex Andonian, Aude Oliva, and Antonio Torralba (2018). “Temporal Relational Reasoning in Videos”. In: *ECCV*.
- Hao Zhou, Wen-gang Zhou, Weizhen Qi, Junfu Pu, and Houqiang Li (2020a). “Improving Sign Language Translation with Monolingual Data by Sign Back-Translation”. In: *CVPR*.
- Hao Zhou, Wengang Zhou, Yun Zhou, and Houqiang Li (2020b). “Spatial-temporal multi-cue network for continuous sign language recognition”. In: *Proceedings of the AAAI Conference on Artificial Intelligence*.
- Zhenxing Zhou, Vincent WL Tam, and Edmund Y Lam (2021). “SignBERT: A BERT-Based Deep Learning Framework for Continuous Sign Language Recognition”. In: *IEEE Access*.
- Ziheng Zhou, Guoying Zhao, Xiaopeng Hong, and Matti Pietikäinen (2014). “A review of recent advances in visual speech decoding”. In: *Image and vision computing*.
- Yimeng Zhuang, Xuankai Chang, Yanmin Qian, and Kai Yu (2016). “Unrestricted Vocabulary Keyword Spotting Using LSTM-CTC”. In: *INTERSPEECH*.
- Ronglai Zuo and Brian Mak (2022). “C2SLR: Consistency-enhanced continuous sign language recognition”. In: *CVPR*.

Appendix A

Statement of Authorship

A statement of authorship is provided for each multi-authored paper included in this thesis. The statements describe the candidate's and co-authors' independent research contributions in the thesis publications. For each publication, there exists a complete statement that is filled out and signed by the candidate and supervisor.

Statement of Authorship for the paper: 'Seeing wake words: Audio-Visual Keyword Spotting' in Chapter 2.

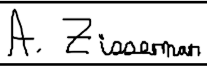
Paper title	Seeing wake words: Audio-Visual Keyword Spotting
Authors	Liliane Momeni , Triantafyllos Afouras, Themos Stafylakis, Samuel Albanie, Andrew Zisserman.
Publication status	Published
Publication details	British Machine Vision Conference, 2020.

Student confirmation

Student name	Liliane Momeni
Contribution to the paper	<ul style="list-style-type: none">• Joint conception of the idea• Research of prior work• Design and implementation of models• Running of all experiments• Writing and presentation of the paper
Signature and Date	April 17th 2024

Supervisor confirmation

By signing the Statement of Authorship, the supervisor is certifying that the candidate made a substantial contribution to the publication, and that the description above is accurate.

Supervisor name	Prof. Andrew Zisserman
Supervisor comments	
Signature and Date	 17/4/2024

Statement of Authorship for the paper: 'Visual Keyword Spotting with Attention' in Chapter 3.

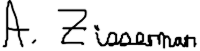
Paper title	Visual Keyword Spotting with Attention
Authors	K R Prajwal*, Liliane Momeni* , Triantafyllos Afouras, Andrew Zisserman.
Publication status	Published
Publication details	British Machine Vision Conference, 2021.

Student confirmation

Student name	Liliane Momeni
Contribution to the paper	<ul style="list-style-type: none">• Joint conception of the idea• Research of prior work• Design of models and evaluations• Running part of the experiments and sign language data processing• Writing and presentation of the paper
Signature and Date	April 17th 2024

Supervisor confirmation

By signing the Statement of Authorship, the supervisor is certifying that the candidate made a substantial contribution to the publication, and that the description above is accurate.

Supervisor name	Prof. Andrew Zisserman
Supervisor comments	
Signature and Date	 17/4/2024

Statement of Authorship for the paper: ‘BSL-1K: Scaling up co-articulated sign language recognition using mouthing cues’ in Chapter 4.

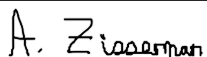
Paper title	BSL-1K: Scaling up co-articulated sign language recognition using mouthing cues
Authors	Samuel Albanie*, Gül Varol*, Liliane Momeni , Triantafyllos Afouras, Joon Son Chung, Andrew Zisserman.
Publication status	Published
Publication details	European Conference on Computer Vision, 2020.

Student confirmation

Student name	Liliane Momeni
Contribution to the paper	<ul style="list-style-type: none">• Joint conception of the idea• Research of prior work• Design and implementation of keyword spotting model• Writing and presentation of the paper
Signature and Date	April 17th 2024

Supervisor confirmation

By signing the Statement of Authorship, the supervisor is certifying that the candidate made a substantial contribution to the publication, and that the description above is accurate.

Supervisor name	Prof. Andrew Zisserman
Supervisor comments	
Signature and Date	 17/4/2024

Statement of Authorship for the paper: ‘Watch, read and lookup: learning to spot signs from multiple supervisors’ in Chapter 5.

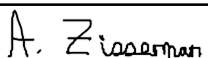
Paper title	Watch, read and lookup: learning to spot signs from multiple supervisors
Authors	Liliane Momeni* , Gül Varol*, Samuel Albanie*, Triantafyllos Afouras, Andrew Zisserman.
Publication status	Published
Publication details	Asian Conference on Computer Vision (Best Application Paper), 2020.

Student confirmation

Student name	Liliane Momeni
Contribution to the paper	<ul style="list-style-type: none">• Joint conception of the idea• Research of prior work• Design and implementation of models• Running of experiments• Writing and presentation of the paper
Signature and Date	April 17th 2024

Supervisor confirmation

By signing the Statement of Authorship, the supervisor is certifying that the candidate made a substantial contribution to the publication, and that the description above is accurate.

Supervisor name	Prof. Andrew Zisserman
Supervisor comments	
Signature and Date	 17/4/2024

Statement of Authorship for the paper: ‘Read and Attend: Temporal Localisation in Sign Language Videos’ in Chapter 6.

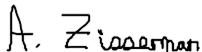
Paper title	Read and Attend: Temporal Localisation in Sign Language Videos
Authors	Gül Varol*, Liliane Momeni* , Samuel Albanie*, Triantafyllos Afouras*, Andrew Zisserman.
Publication status	Published
Publication details	Conference on Computer Vision and Pattern Recognition, 2021.

Student confirmation

Student name	Liliane Momeni
Contribution to the paper	<ul style="list-style-type: none">• Joint conception of the idea• Research of prior work• Design and implementation of models• Running of experiments• Writing and presentation of the paper
Signature and Date	April 17th 2024

Supervisor confirmation

By signing the Statement of Authorship, the supervisor is certifying that the candidate made a substantial contribution to the publication, and that the description above is accurate.

Supervisor name	Prof. Andrew Zisserman
Supervisor comments	
Signature and Date	 17/4/2024

Statement of Authorship for the paper: ‘Automatic dense annotation of large-vocabulary sign language videos’ in Chapter 7.

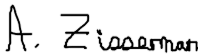
Paper title	Automatic dense annotation of large-vocabulary sign language videos
Authors	Liliane Momeni* , Hannah Bull*, K R Prajwal*, Samuel Albanie, Gül Varol, Andrew Zisserman.
Publication status	Published
Publication details	European Conference on Computer Vision, 2022.

Student confirmation

Student name	Liliane Momeni
Contribution to the paper	<ul style="list-style-type: none">• Joint conception of the idea• Research of prior work• Design and implementation of models• Running of all experiments• Writing and presentation of the paper
Signature and Date	April 17th 2024

Supervisor confirmation

By signing the Statement of Authorship, the supervisor is certifying that the candidate made a substantial contribution to the publication, and that the description above is accurate.

Supervisor name	Prof. Andrew Zisserman
Supervisor comments	
Signature and Date	 17/4/2024

Statement of Authorship for the paper: ‘Weakly-supervised Fingerspelling Recognition in British Sign Language’ in Chapter 8.

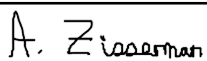
Paper title	Weakly-supervised Fingerspelling Recognition in British Sign Language
Authors	K R Prajwal*, Hannah Bull*, Liliane Momeni* , Samuel Albanie, Gül Varol, Andrew Zisserman.
Publication status	Published
Publication details	British Machine Vision Conference, 2022.

Student confirmation

Student name	Liliane Momeni
Contribution to the paper	<ul style="list-style-type: none">• Joint conception of the idea• Research of prior work• Design of models• Data pre-processing and manual test set collection• Writing and presentation of the paper
Signature and Date	April 17th 2024

Supervisor confirmation

By signing the Statement of Authorship, the supervisor is certifying that the candidate made a substantial contribution to the publication, and that the description above is accurate.

Supervisor name	Prof. Andrew Zisserman
Supervisor comments	
Signature and Date	 17/4/2024

Statement of Authorship for the paper: ‘A Tale of Two Languages: Large-Vocabulary Continuous Sign Language Recognition from Spoken language supervision’ in Chapter 9.

Paper title	A Tale of Two Languages: Large-Vocabulary Continuous Sign Language Recognition from Spoken language supervision
Authors	Charles Raude*, K R Prajwal*, Liliane Momeni* , Hannah Bull, Samuel Albanie, Andrew Zisserman, Gül Varol.
Publication status	Unpublished - work in progress
Publication details	

Student confirmation

Student name	Liliane Momeni
Contribution to the paper	<ul style="list-style-type: none">• Joint conception of the idea• Research of prior work• Design of models and evaluations• Data pre-processing and manual test set collection• Writing and presentation of the paper
Signature and Date	April 17th 2024

Supervisor confirmation

By signing the Statement of Authorship, the supervisor is certifying that the candidate made a substantial contribution to the publication, and that the description above is accurate.

Supervisor name	Prof. Andrew Zisserman
Supervisor comments	
Signature and Date	A. Zisserman 17/4/2024

Statement of Authorship for the paper: ‘Verbs in Action: Improving verb understanding in video-language models’ in Chapter 10.

Paper title	Verbs in Action: Improving verb understanding in video-language models
Authors	Liliane Momeni , Mathilde Caron, Arsha Nagrani, Andrew Zisserman, Cordelia Schmid.
Publication status	Published
Publication details	International Conference on Computer Vision, 2023.

Student confirmation

Student name	Liliane Momeni
Contribution to the paper	<ul style="list-style-type: none">• Joint conception of the idea• Research of prior work• Design and implementation of models• Running of all experiments• Writing and presentation of the paper
Signature and Date	April 17th 2024

Supervisor confirmation

By signing the Statement of Authorship, the supervisor is certifying that the candidate made a substantial contribution to the publication, and that the description above is accurate.

Supervisor name	Prof. Andrew Zisserman
Supervisor comments	
Signature and Date	