Scaling Up Sign Spotting Through Sign Language Dictionaries

Gül Varol[1,2] · Liliane Momeni[1] · Samuel Albanie[1,3] · Triantafyllos Afouras[1] · Andrew Zisserman[1]

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 footage which is sparsely labelled using mouthing cues; (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 page.

Keywords Sign language recognition · Sign spotting · Few-shot learning

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 (see Fig. 1). Sign languages represent the natural means of communication for deaf communities (Sutton-Spence and Woll 1999) and sign spotting has a broad range of practical applications. Exam-

✉ Gül Varol
  gul@robots.ox.ac.uk
  https://www.robots.ox.ac.uk/~vgg/research/bsldict/

  Liliane Momeni
  liliane@robots.ox.ac.uk

  Samuel Albanie
  albanie@robots.ox.ac.uk

  Triantafyllos Afouras
  afourast@robots.ox.ac.uk

  Andrew Zisserman
  az@robots.ox.ac.uk

[1] Visual Geometry Group, University of Oxford, Oxford, UK

[2] LIGM, École des Ponts, Univ Gustave Eiffel, CNRS, Marne-la-Vallée, France

[3] Department of Engineering, University of Cambridge, Cambridge, UK

ples 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 available to deaf communities; and building sign language datasets by automatically labelling examples of signs.

Recently, deep neural networks, equipped with large-scale, labelled datasets produced considerable progress in audio (Coucke et al. 2019; Véniat et al. 2019) and visual (Momeni et al. 2020; 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. 2018)]], but such datasets are not available for sign language.

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 typically: (1)

consist of *isolated signs* which differ in appearance from the *co-articulated*[1] sequences of continuous signs (for which we ultimately wish to perform spotting); and (2) differ in speed (are performed more slowly) relative to co-articulated signing. Furthermore, (3) dictionaries only possess a few examples of each sign (so learning must be *low shot*); and as one more challenge, (4) 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 Sect. 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 obtained from mouthing cues (Albanie et al. 2020); (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 a large-scale, subtitled dataset 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.

Our loss formulation is an extension of InfoNCE (Oord et al. 2018; Wu et al. 2018) [and in particular the multiple instance variant MIL-NCE (Miech et al. 2020)]. The novelty of our method lies in the *batch formulation* that leverages the mouthing annotations, subtitles, and visual dictionaries to define positive and negative bags. Moreover, this work specifically focuses on computing similarities across two different domains to learn matching between isolated and co-articulated signing.

We make the following contributions, originally introduced in Momeni et al. (2020): (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 vali-

date 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. In addition, we extend these contributions with (7) the demonstration that our framework can be effectively deployed to obtain large numbers of sign examples, enabling state-of-the-art performance to be reached on the BSL-1K sign recognition benchmark (Albanie et al. 2020), and on the recently released BOBSL dataset (Albanie et al. 2021).

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 (Camgoz et al. 2017; Huang et al. 2018; Ye and Tian 2018; Zhou et al. 2020), and Transformers (Camgoz et al. 2020) proving to be effective mechanisms for this task. Recently, convolutional neural networks have emerged as the dominant approach for appearance modelling (Camgoz et al. 2017), and in particular, action recognition models using spatio-temporal convolutions (Carreira and Zisserman 2017) have proven very well-suited for video-based sign recognition (Joze and Koller 2019; Li et al. 2019; Albanie et al. 2020). We adopt the I3D architecture (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 et al. 2014). Different

---

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

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 (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 variations in the dictionary for a given word (and validate the improvements brought by doing so in Sect. 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 work with a much larger vocabulary (9K words and phrases).

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; 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 instead most effective when jointly training on two domains after pre-training on the bigger domain. The concurrent work of Li et al. (2020) 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 from mouthings (Albanie et al. 2020) and dictionaries (this integrates cues beyond the learning formulation in Li et al. (2020)); (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 multiple instance learning formulation that explicitly accounts for signing variations that are present in the dictionary (Fig. 2).

Sign language datasets. A number of sign language datasets have been proposed for studying Finnish (Viitaniemi et al. 2014), German (Koller et al. 2015; Agris et al. 2008), American (Athitsos et al. 2008; Joze and Koller 2019; Li et al. 2019; Wilbur and Kak 2006) and Chinese (Chai et al.

2014; Huang et al. 2018) sign recognition. For British Sign Language (BSL), (Schembri et al. 2013) gathered the BSL Corpus which represents continuous signing, labelled with fine-grained linguistic annotations. 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. Further details on this method are given in Sect. 3.1. 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 English 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. While BSLDICT does not represent a linguistic corpus, as the correspondences to English words and phrases are not carefully annotated with *glosses*[2], it is significantly larger than its linguistic counterparts (e.g., 4K videos corresponding to 2K words in BSL SignBank (Fenlon et al. 2014), as opposed to 14K videos of 9K words in BSLDICT), therefore BSLDICT is particularly suitable to be used in conjunction with subtitles.

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 2016). 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; Yang et al. 2018; 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.

---

[2] Glosses are atomic lexical units used to annotate sign languages.
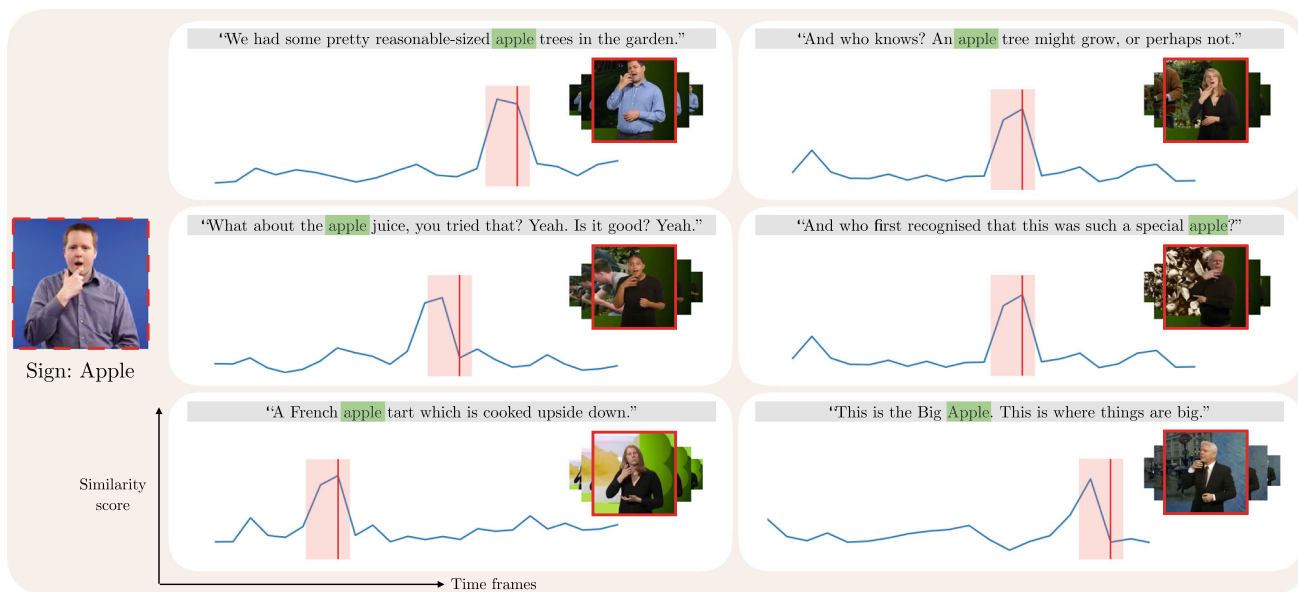
Fig. 1 We consider the task of *sign spotting* in co-articulated, continuous signing. Given a query dictionary video of an isolated sign (e.g., "apple"), we aim to identify *whether* and *where* it appears in videos of continuous signing. The wide domain gap between dictionary examples of *isolated* signs and target sequences of *continuous* signing makes the task extremely challenging (Color figure online)
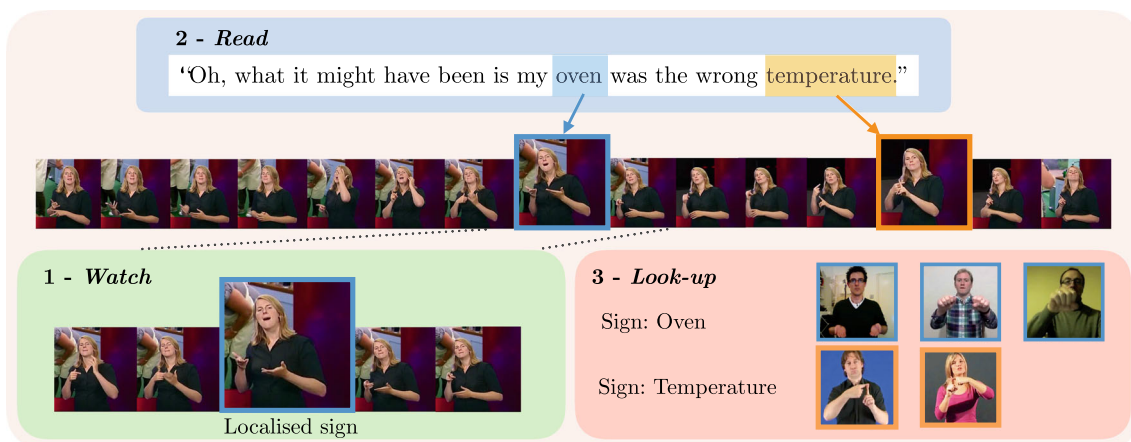
Fig. 2 The proposed Watch, Read and Lookup framework trains sign spotting embeddings with three cues: (1) *watching* videos and learning from sparse annotation in the form of localised signs obtained from mouthings (Albanie et al. 2020) (lower-left); (2) *reading* subtitles to find candidate signs that 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) (Color figure online)

## 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 $\mathcal{X}_{\mathfrak{L}}$ denote the space of RGB video segments containing a frontal-facing individual communicating in sign language $\mathfrak{L}$ and denote by $\mathcal{X}_{\mathfrak{L}}^{\text{single}}$ its restriction to the set of segments containing a single sign. Further, let $\mathcal{T}$ denote the space of subtitle sentences and $\mathcal{V}_{\mathfrak{L}} = \{1, \ldots, V\}$ denote the *vocabulary*—an index set corresponding to an enumeration of written words that are equivalent to signs that can be performed in $\mathfrak{L}$.[3]

Our objective, illustrated in Fig. 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 sign-

---

[3] Sign 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 (Sutton-Spence and Woll 1999).

Keyword Spotting Annotation Pipeline

Result: BSL-1K
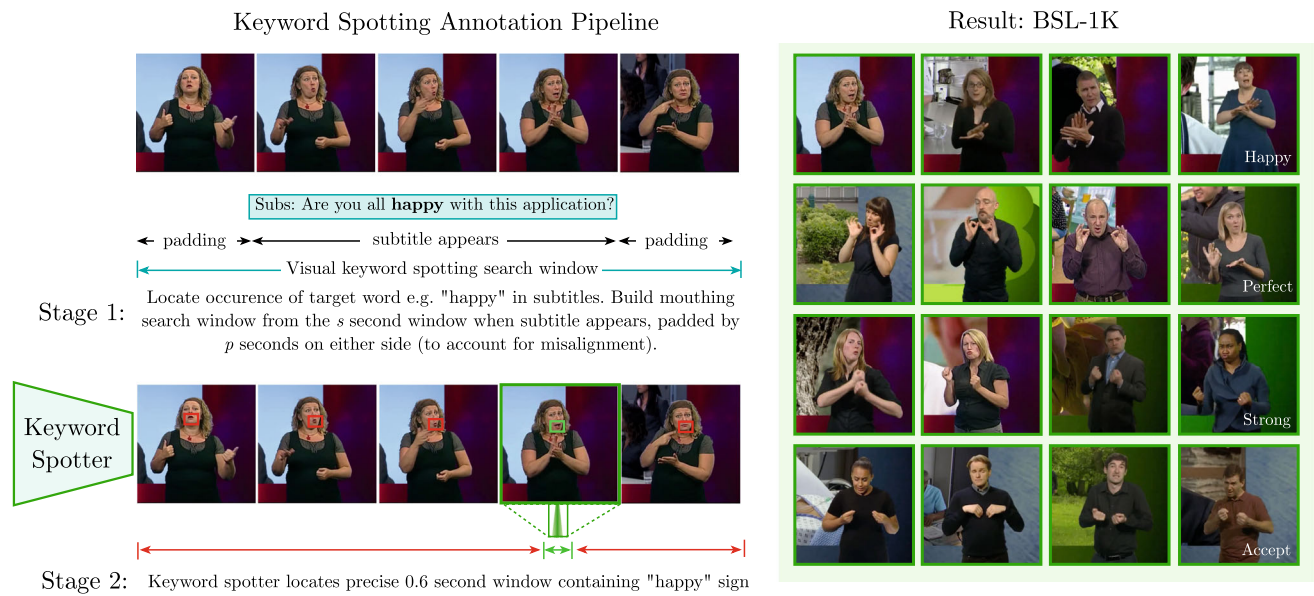
Fig. 3 Mouthing-based sign annotation from (Albanie et al. 2020): (Left, the annotation pipeline): Stage 1: for a given sign (e.g. "happy"), each instance of the word in the subtitles provides a candidate temporal segment where the sign may occur (the subtitle timestamps are padded by several seconds to account for the asynchrony between audio-aligned subtitles and signing interpretation); Stage 2: a mouthing visual keyword spotter uses the lip movements of the signer to perform precise localisation of the sign within this window. (Right): Examples of localised signs through mouthings from the BSL-1K dataset- produced by applying keyword spotting for a vocabulary of 1K words (Color figure online)

ing, $\mathcal{S} = \{(x_i, s_i) : i \in \{1, \ldots, I\}, x_i \in \mathcal{X}_{\mathfrak{L}}, s_i \in \mathcal{T}\}$; (ii) a sparse collection of temporal sub-segments of these videos that have been annotated with their corresponding word, $\mathcal{M} = \{(x_k, v_k) : k \in \{1, \ldots, K\}, v_k \in \mathcal{V}_{\mathfrak{L}}, x_k \in \mathcal{X}_{\mathfrak{L}}^{single}, \exists (x_i, s_i) \in \mathcal{S} \, s.t. \, x_k \subseteq x_i\}$; (iii) a curated *dictionary* of signing instances $\mathcal{D} = \{(x_j, v_j) : j \in \{1, \ldots, J\}, x_j \in \mathcal{X}_{\mathfrak{L}}^{single}, v_j \in \mathcal{V}_{\mathfrak{L}}\}$. To address the sign spotting task, we propose to learn a *data representation* $f : \mathcal{X}_{\mathfrak{L}} \rightarrow \mathbb{R}^d$ that maps video segments to vectors such that they 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 $x, x' \in \mathcal{X}_{\mathfrak{L}}$ and $v, v' \in \mathcal{V}_{\mathfrak{L}}$, we seek a data representation, f, that satisfies the constraint $\delta_{f(x)f(x')} = \delta_{vv'}$, where δ represents the Kronecker delta.

3.1 Sparse Annotations from Mouthing Cues

As the source of temporal video segments with corresponding word annotations, $\mathcal{M}$, we make use of automatic annotations that were collected as part of our prior work on visual keyword spotting with mouthing cues (Albanie et al. 2020), which we briefly recap here. Signers sometimes mouth a word while simultaneously signing it, as an additional signal (Bank et al. 2011; Sutton-Spence and Woll 1999; Sutton-Spence 2007), performing similar lip patterns as for the spoken word. Figure 3 presents an overview of how we use such mouthings to spot signs.

As a starting point for this approach, we assume access to TV footage that is accompanied by: (i) a frontal facing sign language interpreter, who provides a translation of the spoken content of the video, and (ii) a subtitle track, representing a direct transcription of the spoken content. The method of Albanie et al. (2020) first searches among the subtitles for any occurrences of "keywords" from a given vocabulary. Subtitles containing these keywords provide a set of candidate temporal windows in which the interpreter may have produced the sign corresponding to the keyword (see Fig. 3, Left, Stage 1). However, these temporal windows are difficult to make use of directly since: (1) the occurrence of a keyword in a subtitle does not ensure the presence of the corresponding sign in the signing sequence, (2) the subtitles themselves are not precisely aligned with the signing, and can differ in time by several seconds. To address these issues, (Albanie et al. 2020) demonstrated that the sign corresponding to a particular keyword can be localised within a candidate temporal window—given by the *padded* subtitle timings to account for the asynchrony between the audio-aligned subtitles and signing interpretation—by searching for its *spoken components* (Sutton-Spence and Woll 1999) amongst the mouth movements of the interpreter. While there are challenges associated with using spoken components as a cue [signers do not typically mouth continuously and may only produce mouthing patterns that correspond to a portion of the keyword (Sutton-Spence and Woll 1999)], it has the

significant advantage of transforming the general annotation problem from classification (i.e., "which sign is this?") into the much easier problem of localisation (i.e., "find a given token amongst a short sequence"). In Albanie et al. (2020), the visual keyword spotter uses the candidate temporal window with the target keyword to estimate the probability that the sign was mouthed at each time step. If the peak probability over time is above a threshold parameter, the predicted location of the sign is taken as the 0.6 s window starting before the position of the peak probability (see Fig. 3, Left, Stage 2). For building the BSL-1K dataset, Albanie et al. (2020) use a probability threshold of 0.5 and runs the visual keyword spotter with a vocabulary of 1350 keywords across 1000 h of signing. A further filtering step is performed 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 1064 words. The resulting BSL-1K dataset has 273K mouthing annotations, some of which are illustrated in Fig. 3 (right). We employ these annotations directly to form the set $\mathcal{M}$ in this work.

3.2 Integrating Cues Through Multiple Instance Learning

To learn f, we must address several challenges. First, as noted in Sect. 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 (e.g., resulting from regional variations 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 (a formal mathematical description of the positive and negative bags described below is deferred to

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 (green regions in 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 (red regions in 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. 4, right-bottom). Negatives (red regions in Fig. 4) 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. 4, 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 $\psi : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ whose value increases as its arguments become more similar (in this work, we use cosine similarity). For notational convenience
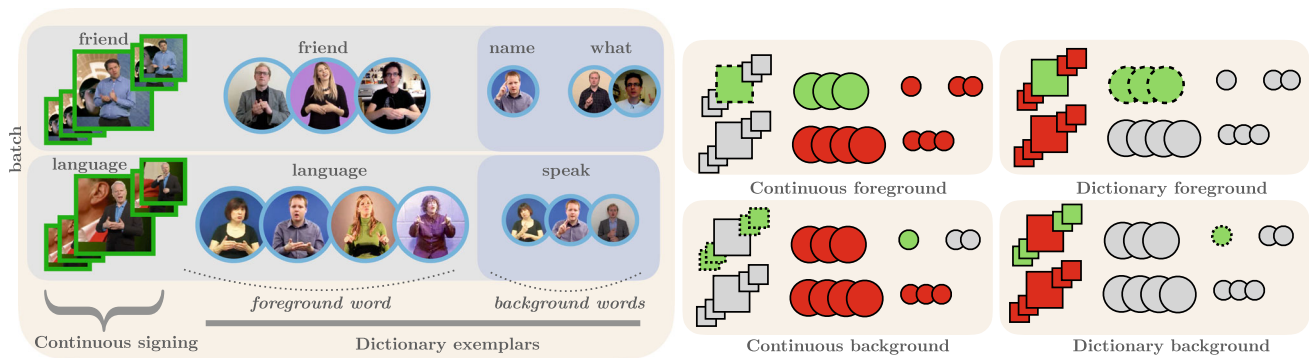
Fig. 4 Batch sampling and positive/negative pairs We illustrate the formation of a batch when jointly training on continuous signing video (squares) and dictionaries of isolated signing (circles). **Left** For each continuous video, we sample the dictionaries corresponding to the labelled word (foreground), 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 Appendix C.2 for details) (Color figure online)

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; Wu et al. 2018) (a non-parametric softmax loss formulation of Noise Contrastive Estimation Gutmann and Hyvärinen 2010) recently proposed by Miech et al. (2020) as MIL-NCE loss:

$$\mathcal{L} = -\mathbb{E}_i \left[\log \frac{\sum\limits_{(j,k)\in\mathcal{P}(i)} e^{\psi_{jk}/\tau}}{\sum\limits_{(j,k)\in\mathcal{P}(i)} e^{\psi_{jk}/\tau} + \sum\limits_{(l,m)\in\mathcal{N}(i)} e^{\psi_{lm}/\tau}} \right], \quad (1)$$

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 Sect. 4).

3.3 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 (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.

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 dic-

tionary 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 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 Sect. 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 $\mathcal{V}_{\mathfrak{L}}$ (see Fig. 4 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 Sect. 4.4). The embeddings are then trained by minimising the loss defined in Eq. (1)

in conjunction 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. 4).

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 epochs 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 show the importance of using such a dense stride for a precise localisation in our ablations (Sect. 4.4). However, for simplicity, all qualitative visualisations are performed with continuous video embeddings obtained with a sliding window of stride 8.

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.

4 Experiments

In this section, we first present the datasets used in this work (including the contributed BSLDICT dataset) in Sect. 4.1, followed by the evaluation protocol in Sect. 4.2. We then illustrate the benefits of the *Watch, Read and Lookup* learning framework for sign spotting against several baselines (Sect. 4.3) with a comprehensive ablation study that validates our design choices (Sect. 4.4). Next, we investigate three applications of our method in Sect. 4.5, 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. We then provide experiments on sign language recognition, significantly improving the state of the art by applying our labelling technique to obtain more training examples automatically (Sects. 4.6 and 4.7). Finally, we discuss limitations of our sign spotting technique using dictionaries (Sect. 4.8).

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 h of continuous signing which offers an ideal setting in which to evaluate the effectiveness of the *Watch, Read and Lookup* 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 1 and described in more detail below. We further include experiments on a new dataset, BOBSL (Albanie et al. 2021), which we describe in Sect. 4.7 together with results. The BOBSL dataset has similar properties to BSL-1K.
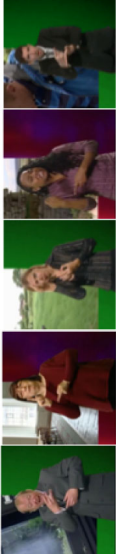
BSL-1K comprises over 1000 hours of video of continuous sign-language-interpreted television broadcasts, with accompanying subtitles of the audio content (Albanie et al. 2020). In Albanie et al. (2020), this data is processed for the task of individual sign recognition: a visual keyword spotter is applied to signer mouthings giving a total of 273K sparsely localised sign annotations from a vocabulary of 1064 signs (169K in the training partition as shown in Table 1). Please refer to Sect. 3.1 and (Albanie et al. 2020) for more details on the automatic annotation pipeline. We refer to Sect. 4.6 for a description of the BSL-1K sign recognition benchmark ($\text{Test}_{2K}^{\text{Rec}}$ and $\text{Test}_{37K}^{\text{Rec}}$ in Table 1).

In this work, we process this data for the task of retrieval, extracting long videos with associated subtitles. In particular, we pad ± 2 s around the subtitle timestamps and we add the corresponding video to our training set if there is a sparse annotation from mouthing falling within this time window—we assume this constraint indicates that the signing is reasonably well-aligned with its subtitles. We further consider only the videos whose subtitle duration is longer than 2 s. For testing, we use the automatic test set (corresponding to mouthing locations with confidences above 0.9). Thus we obtain 78K training ($\text{Train}^{\text{ReT}}$) and 2K test ($\text{Test}^{\text{ReT}}$) videos as shown in Table 1, 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 (British 1999), giving us a total of 14,210 video clips for a vocabulary of 9283 signs. Each sign is typically performed several times by different signers, often in different ways. The dictionary videos are linked from 28 known website sources and each source has at least 1 signer. We used face embeddings computed with SENet-50 (Hu et al. 2019) (trained on VGGFace2 Cao et al. 2018) to cluster signer identities and manually verified that there are a total of 124 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 (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

Table 1 Datasets: We provide (i) the number of individual signing videos, (ii) the vocabulary size of the *annotated* signs, and (iii) the number of signers for several subsets of BSL-1K and BSLDICT

Dataset	Split	#Videos	Vocabulary	#Signers	Examples
BSL-1K Albanie et al. (2020)	Train$^{\text{Rec}}$	169K	1064	36	
	Test$^{\text{Rec}}_{2K}$	2103	334	4	
	Test$^{\text{Rec}}_{37K}$	36,854	950	4	
	Train$^{\text{ReT}}$	78,211	1064	36	
	Test$^{\text{ReT}}$	1834	264	4	
BSLDICT	Full	14,210	9283	124	
	1K-subset	2963	1064	70	

BSL-1K is large in the number of annotated signs whereas BSLDICT is large in the vocabulary size. Note that BSL-1K is constructed differently depending on whether it is used for the task of recognition or retrieval: for retrieval, longer signing sequences are used around individual localised signs as described in Sect. 4.1

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. A collection of metadata associated for the BSLDICT dataset is made publicly available, as well as our pre-computed video embeddings from this work.

For the experiments in which BSLDICT is filtered to the 1064 vocabulary of BSL-1K, we have 3K videos as shown in Table 1. Within this subset, each sign has between 1 and 10 examples (average of 3).

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 as in retrieval. 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 times that a dictionary clip of s_i appears 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 mouthing 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 the BSL-1K test set is class-unbalanced, we report performances averaged over the test classes.

4.3 Comparison to Baselines

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. Finally, we 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 Table 2: (i) I3D^{BSL-1K} provided by Albanie et al. (2020) which is trained only on the BSL-1K dataset, and we also train (ii) I3D^{BSLDICT} and (iii) I3D$^{BSL-1K,BSLDICT}$. Training only on BSLDICT (I3D^{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 I3D^{BSL-1K} jointly on the two datasets (I3D$^{BSL-1K,BSLDICT}$), 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 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; Wu et al. 2018) 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 2 summarises 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 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 improves the results on the unseen setting.

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). The sign spotting benchmark requires a model to localise every instance belonging to a given set of sign classes (334 in total) within long sequences of untrimmed footage. The benchmark is challenging because each sign appears infrequently (corresponding to approximately one positive instance in every 90 minutes of continuous signing). 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).

Table 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.

| Embedding arch. | Supervision | Train (1064) Seen (264) | | Train (800) Unseen (264) | |
		mAP	R@5	mAP	R@5
I3D$^{\textsc{BslDict}}$	Classification	2.68	3.57	1.21	1.29
I3D$^{\textsc{BSL-1K}}$ Albanie et al. (2020)	Classification	13.09	17.25	6.74	8.94
I3D$^{\textsc{BSL-1K},\textsc{BslDict}}$	Classification	19.81	25.57	4.81	6.89
I3D$^{\textsc{BSL-1K},\textsc{BslDict}}$+MLP	Classification	37.13 ± 0.29	39.68 ± 0.57	10.33 ± 0.43	13.33 ± 1.11
I3D$^{\textsc{BSL-1K},\textsc{BslDict}}$+MLP	InfoNCE	43.59 ± 0.76	52.59 ± 0.75	11.40 ± 0.42	14.76 ± 0.40
I3D$^{\textsc{BSL-1K},\textsc{BslDict}}$+MLP	Watch-Lookup	44.72 ± 0.85	55.51 ± 2.17	11.02 ± 0.27	15.03 ± 0.45
I3D$^{\textsc{BSL-1K},\textsc{BslDict}}$+MLP	Watch-Read-Lookup	**47.93** ± 0.20	**60.76** ± 1.45	**14.86** ± 1.29	**19.85** ± 1.94

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. We report the relatively cheaper MLP-based models with three random seeds for each model and report the mean and the standard deviation
Best performances are highlighted in bold following standard practice

Table 3 Extending the dictionary vocabulary We show the benefits of sampling dictionary videos outside of the sparse annotations, using subtitles.

Supervision	Dictionary Vocab	mAP	R@5
Watch-Read-Lookup	800 training vocab	14.86 ± 1.29	19.85 ± 1.94
Watch-Read-Lookup	9K full vocab	**15.82** ± 0.48	**21.67** ± 0.72

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)
Best performances are highlighted in bold following standard practice
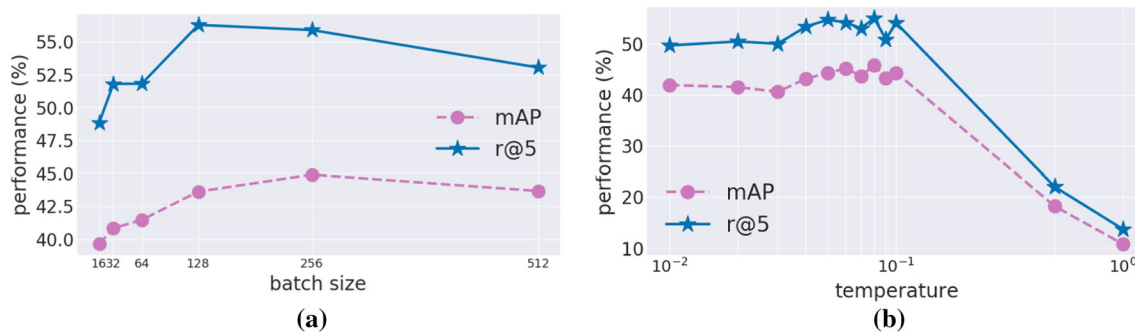
Fig. 5 The effect of **a** the **batch size** that determines the number of negatives across sign classes and **b** the **temperature** hyper-parameter for the MIL-NCE loss in Watch-Lookup against mAP and R@5 (trained on the full 1064 vocab.) (Color figure online)

4.4 Ablation Study

We provide ablations for the learning hyperparameters, such as the batch size and the temperature; the mouthing confidence threshold as the training data selection parameter; and the stride parameter of the sliding window at test time.

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 Fig. 5a an improvement in performance with a greater number 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 conclude from Fig. 5b that setting τ to values between [0.04–0.10] does not impact the performance significantly; therefore, we keep $\tau = 0.07$ following previous work (Wu et al. 2018; He et al. 2020) for all other experiments. However, values outside this range negatively impact the performance, especially for high values, i.e.,

Table 4 Mouthing confidence threshold the results suggest that lower confidence automatic annotations of BSL-1K provide better training, by increasing the amount of data (training on the full 1064 vocabulary with Watch-Lookup)

Mouthing confidence	Training size	mAP	R@5
0.9	10K	37.55	47.54
0.8	21K	39.49	48.84
0.7	33K	41.87	51.15
0.6	49K	42.44	52.42
0.5	78K	**43.65**	**53.03**

Table 5 Stride parameter of sliding window A small stride at test time, when extracting embeddings from the continuous signing video, allows us to temporally localise the signs more precisely

Stride	mAP	R@5
16	31.96	38.98
8	38.46	47.38
4	44.92	54.65
2	**45.39**	**55.63**
1	43.65	53.03

The window size is 16 frames and the typical co-articulated sign duration is 7–13 frames at 25 fps. (testing 1064-class model trained with Watch-Lookup)

{0.50, 1.00}; we observe a major decrease in performance when τ approaches 1.

Mouthing confidence threshold at training. As explained in Sect. 3.1, the sparse annotations from the BSL-1K dataset are obtained automatically by running a visual keyword spotting method based on mouthing cues. The dataset provides a confidence value associated with each label ranging between 0.5 and 1.0. Similar to Albanie et al. (2020), we experiment with different thresholds to determine the training set. Lower thresholds result in a noisier but larger training set. From Table 4, we conclude that 0.5 mouthing confidence threshold performs the best. This is in accordance with the conclusion from Albanie et al. (2020).

Effect of the sliding window stride. As explained in Sect. 3.3, at test time, we extract features from the continuous signing sequence using a sliding window approach with 1 frame as the stride parameter. In Table 5, we investigate the effect of the stride parameter. Our window size is 16 frames, i.e., the number of input frames for the I3D feature extractor. A standard approach when extracting features from longer videos is to use a sliding window with 50% overlap (i.e., stride of 8 frames). However, this means the temporal resolution of the search space is reduced by a factor of 8, and a stride of 8 may skip the most discriminative moment since a sign duration is typically between 7-13 frames (but can be shorter) (Pfister et al. 2013) in continuous signing video. In Table 5, we see that we can gain a significant localisation improvement by computing the similarities more densely, e.g., stride of 4 frames may be sufficiently dense. In our experiments, we use stride 1.

We refer to for additional ablations.

4.5 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. 6, 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. 7, we show qualitative examples of localising multiple signs in a given sentence in BSL-1K, where we only query the words that occur 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 (Li et al. 2019), an ASL isolated sign language dataset with a 2K vocabulary. We provide some examples in Fig. 8. We automatically identify several signs with similar manual features some of which correspond to different meanings in English (left), as well as same meanings, such as "ball", "stand", "umbrella" (right).

4.6 Sign Language Recognition

As demonstrated qualitatively in Sect. 4.5, we can reliably obtain automatic annotations using our sign spotting technique when the search space is reduced to candidate words in the subtitle. A natural way to exploit our method is to apply it on the BSL-1K *training* set in conjunction with the weakly-aligned subtitles to collect new localised sign instances. This
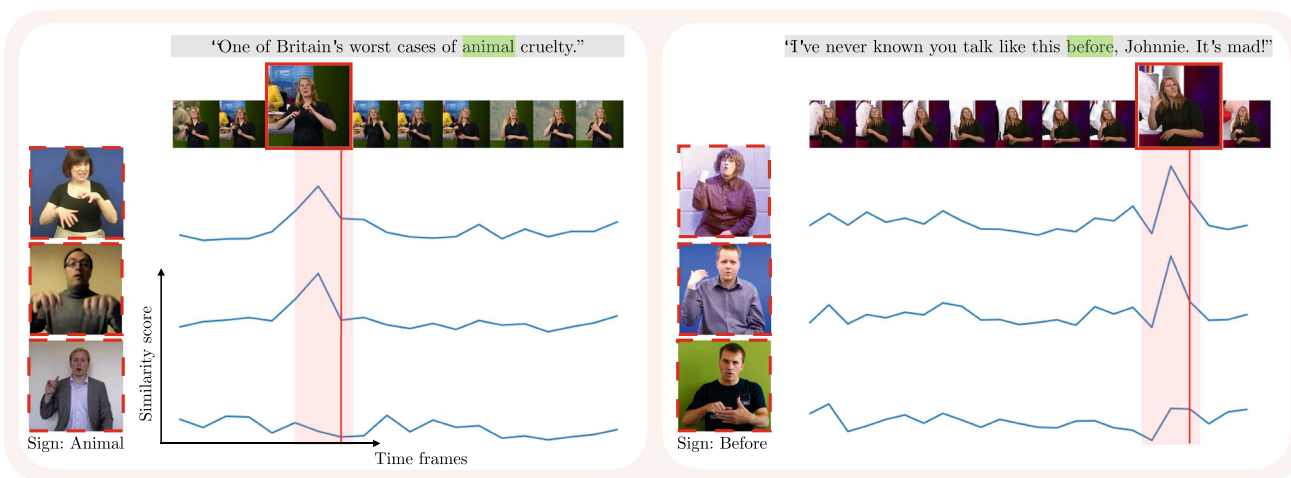
Fig. 6 Sign variant identification: We plot the similarity scores between BSL-1K test clips and BSLDICT variants of the sign "animal" (left) and "before" (right) over time. A high similarity occurs for the first two rows, where the BSLDICT examples match the variant used in BSL-1K. The labelled mouthing times from Albanie et al. (2020) are shown by red vertical lines and approximate windows for signing times are shaded.

Note that neither the mouthing annotations (ground truth) nor the dictionary spottings provide the *duration* of the sign, but only a point in time where the response is highest. The mouthing peak (red vertical line) tends to appear at the end of the sign (due to the use of LSTM in visual keyword spotter). The dictionary peak (blue curve) tends to appear in the middle of the sign (Color figure online)
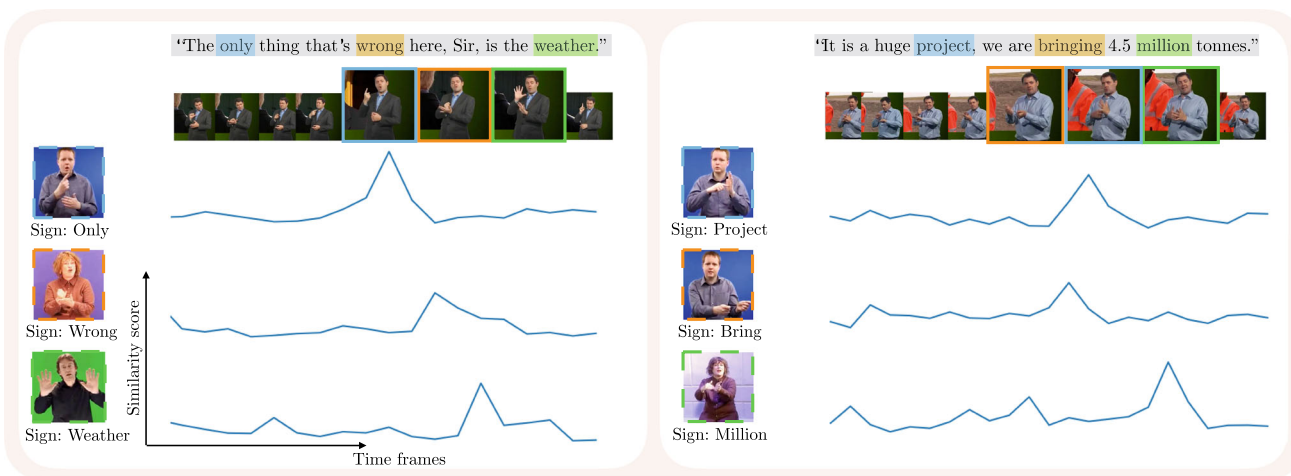
Fig. 7 Densification: We plot the similarity scores between BSL-1K test clips and BSLDICT examples over time, by querying only the words in the subtitle. We visually inspect the quality of the dictionary spottings with which we obtain cases of multiple words per subtitle spotted. The predicted locations of the signs correspond to the peak similarity scores. Note that unlike in Fig. 6, we cannot overlay the ground truth since the annotations using the mouthing cues are not dense enough to provide ground truth sign locations for 3 words per subtitle (Color figure online)

allows us to train a sign recognition model: in this case, to *retrain* the I3D architecture from Albanie et al. (2020) which was previously supervised only with signs localised through mouthings.

BSL-1K automatic annotation. Similar to our previous work using mouthing cues (Albanie et al. 2020), where words in the subtitle were queried within a neighborhood around the subtitle timestamps, we query each subtitle word if they fall within a predefined set of vocabulary. In particular, we query words and phrases from the 9K BSLDICT vocabulary if they

occur in the subtitles. To determine whether a query from the dictionary occurs in the subtitle, we apply several checks. We look for the original word or phrase as it appears in the dictionary, as well as its text-normalised form (e.g., "20" becomes "twenty"). For the subtitle, we look for its original, text-normalised, and lemmatised forms. Once we find a match between any form of the dictionary text and any form of the subtitle text, we query the dictionary video feature within the search window in the continuous video features. We use search windows of ±4 s padding around the subtitle times-

Fig. 8 "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 (Color figure online)
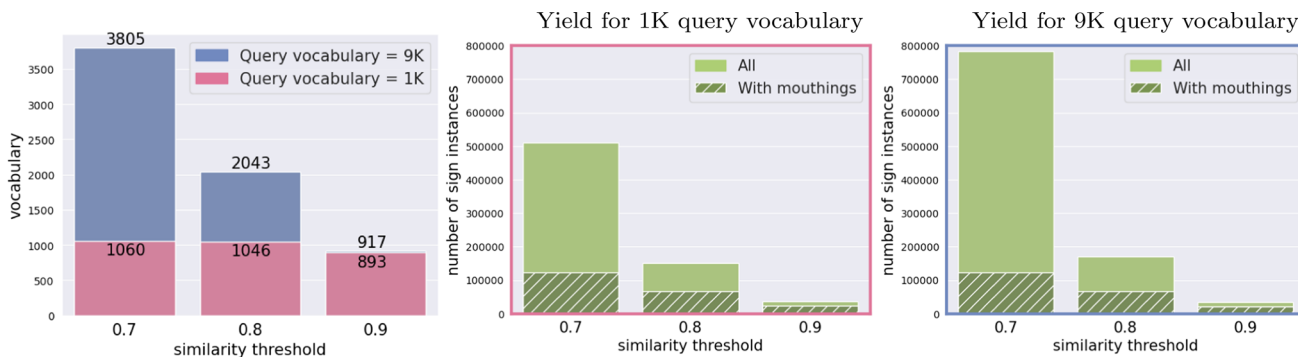
Fig. 9 Statistics on the yield from the automatic annotations: We plot the vocabulary size (left) and the number of localised sign instances (middle) and (right) over several similarity thresholds for the new automatic annotations in the training set that we obtain through dictionaries. While we obtain a large number of localised signs (783K at 0.7 threshold) for the full 9K vocabulary, in our recognition experiments we use a subset of 510K annotations that correspond to the 1K vocabulary. To approximately quantify the amount of annotations that represent duplicates from those found through mouthing cues, we count those localisations for which the same keyword exists for mouthing annotations within the same search window. We observe that the majority of the annotations are new (783K vs 122K) (Color figure online)

tamps. We compute the similarity between the continuous signing search window and each of the dictionary variants for a given word: we record the frame location of maximum similarity for all variants and choose the best match as the one with highest similarity score. The final sign localisations are obtained by filtering the peak similarity scores to those above 0.7 threshold—resulting in a vocabulary of 4K signs—and taking 32 frames centered around the peak location. Fig. 9 summarises several statistics computed over the training set. We note that sign spotting with dictionaries (D) is more effective than with mouthing (M) in terms of the yield (510K versus 169K localised signs). Since, D can include duplicates from M, we further report the number of instances for which a mouthing spotting for the same keyword query exists within the same search window. We find that the majority of our D spottings represent new, not previously localised instances (see Fig. 9 right).

BSL-1K sign recognition benchmark. 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 (a) running our dictionary-based sign spotting technique

on the BSL-1K test set and (b) verifying the predicted sign instances with human annotators using the VIA tool (Dutta and Zisserman 2019) as in Albanie et al. (2020). Our goal in keeping these two divisions is three-fold: (i) $\text{Test}_{2K}^{\text{Rec}}$ is the result of annotating "mouthing" spottings above 0.9 confidence, which means the models can largely rely on mouthing cues to recognise the signs. The new $\text{Test}_{37K}^{\text{Rec}}$ annotations have both "mouthing" (10K) and "dictionary" (27K) spottings. The dictionary annotations are the result of annotating dictionary spottings above 0.7 confidence from this work; therefore, models are required to recognise the signs even in the absence of mouthing, reducing the bias towards signs with easily spotted mouthing cues. (ii) $\text{Test}_{37K}^{\text{Rec}}$ spans a much larger fraction of the training vocabulary as seen in Table 1, with 950 out of 1064 sign classes (vs only 334 classes in the original benchmark $\text{Test}_{2K}^{\text{Rec}}$ of Albanie et al. (2020)). (iii) We wish to maintain direct comparison to our previous work (Albanie et al. 2020); therefore, we report on both sets in this work.

Comparison to prior work. In Table 6, we compare three I3D models trained on mouthing annotations (M), dictionary annotations (D) , and their combination (M+D). First, we

Table 6 An improved I3D sign recognition model We find signs via automatic dictionary spotting (D), significantly expanding the training and testing data obtained from mouthing cues by Albanie et al. (2020) (M).

| Training | #ann. | $\text{Test}_{2K}^{\text{Rec}}$ Albanie et al. (2020) 2K inst. / 334 cls. | | | | $\text{Test}_{37K}^{\text{Rec}}$ 37K inst. / 950 cls. | | | |
| | | 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**

We also significantly expand the test set by manually verifying these new automatic annotations from the test partition ($\text{Test}_{2K}^{\text{Rec}}$ vs $\text{Test}_{37K}^{\text{Rec}}$). By training on the extended M+D data, we obtain state-of-the-art results, outperforming the previous work of Albanie et al. (2020). §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)

observe that D-only model significantly outperforms M-only model on $\text{Test}_{37K}^{\text{Rec}}$ (60.9% vs 26.4%), while resulting in lower performance on $\text{Test}_{2K}^{\text{Rec}}$ (70.8% vs 76.6%). This may be due to the strong bias towards mouthing cues in the small test set $\text{Test}_{2K}^{\text{Rec}}$. Second, the benefits of combining annotations from both can be seen in the sign classifier trained using 678K automatic annotations. This obtains state-of-the-art performance on $\text{Test}_{2K}^{\text{Rec}}$, as well as the more challenging test set $\text{Test}_{37K}^{\text{Rec}}$. All three models in the table (M, D, M+D) are pretrained on Kinetics (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.

Our results can be interpreted as bootstrapping from an initial model, which has access to a large audio-visual training set with mouthing annotations. The M recognition model has distilled this information while incorporating manual patterns. The Watch-Read-Lookup framework has mainly relied on these mouthing locations to learn matching with dictionary samples. The D recognition model is the result of this series of annotation expansion. The final recognition model therefore exploits multiple sources of supervision. We refer to our recent work (Varol et al. 2021) for a complementary way of expanding the automatic annotations. There, we introduce an attention-based sign localisation where the localisation ability emerges from a sequence prediction task.

Sign recognition ablations. In Table 7 we provide further ablations for training the recognition models based on automatic dictionary spotting annotations. In particular, we investigate (i) the similarity threshold that determines the amount of training data, as well as the noise, and (ii) no padding versus ±4-s padding to the subtitle locations when defining the search window. We see in Table 7 that filtering the sign annotations with a high threshold such as 0.9, denoted with D.9, drastically reduces the training size (from 510K to 36K) which in return results in poor recognition performance. The accuracy with D.7 is slightly above that of D.8. Moreover, both the performance and the training size

Table 7 Recognition ablations We train on a portion of the automatic annotations obtained through dictionaries

| Training | #ann. | $\text{Test}_{37K}^{\text{Rec}}$ Per-instance | | Per-class | |
		Top-1	Top-5	Top-1	Top-5
D.9 (sim ≥ 0.9)	36K	28.4	44.8	15.8	28.8
D.8 (sim ≥ 0.8)	152K	60.6	78.0	**35.6**	52.6
D.7 (sim ≥ 0.7)	510K	**60.9**	**80.3**	34.7	**53.5**
D.7 (no padding)	236K	56.5	75.9	28.7	45.4

We filter the localisations which have more than a similarity threshold: 0.7, 0.8, 0.9. We find that lower threshold results in a larger training and increased performance. On the other hand, restricting the annotations to only those which fall within the subtitle timestamps without temporal padding for the search window reduces the accuracy

Table 8 Recognition results on the BOBSL dataset We report 2281-way classification performance on the 25,045 manually verified test signs of BOBSL

| Training | #ann. | BOBSL Per-instance | | Per-class | |
		top-1	top-5	top-1	top-5
M.8	154K	39.0	55.1	43.0	66.3
M.5	502K	39.6	55.2	44.3	66.3
D.9	9K	24.0	39.5	4.5	9.6
D.8	272K	63.2	79.0	25.7	41.2
D.75	727K	64.8	80.8	29.1	44.7
M.8+D.8	426K	**75.8**	92.4	50.5	**77.6**
M.5+D.75	1.2 M	**75.8**	**92.5**	**51.1**	77.1

We train on different training sets obtained with automatic annotations through mouthing (M) and dictionaries (D) at different thresholds. Combining two sets of annotations yields the best result, even without lowering the thresholds
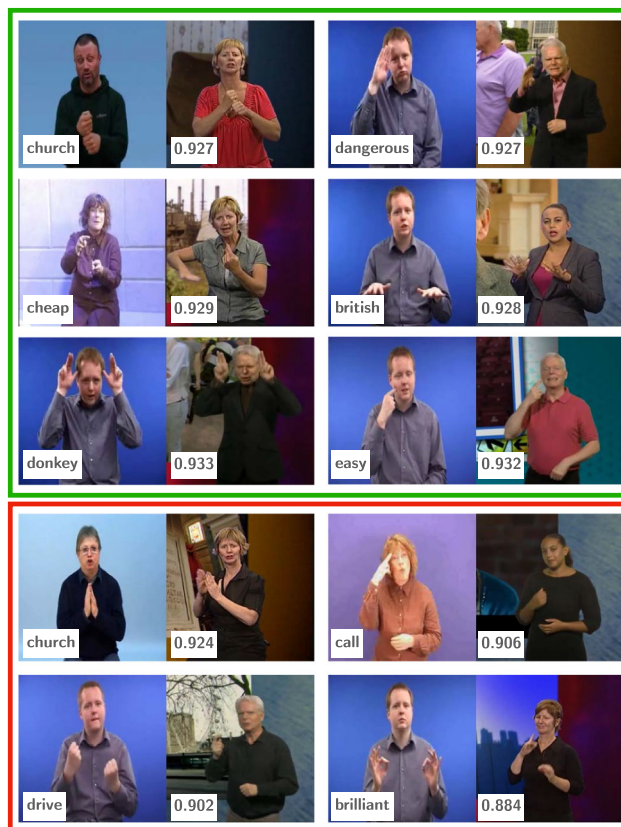
decreases if we restrict the sign annotations to those which fall within the subtitle timestamps, i.e., no temporal padding in the search window when applying sign spotting. We retain a similarity threshold of 0.7 and a ±4-s padding for our final model.
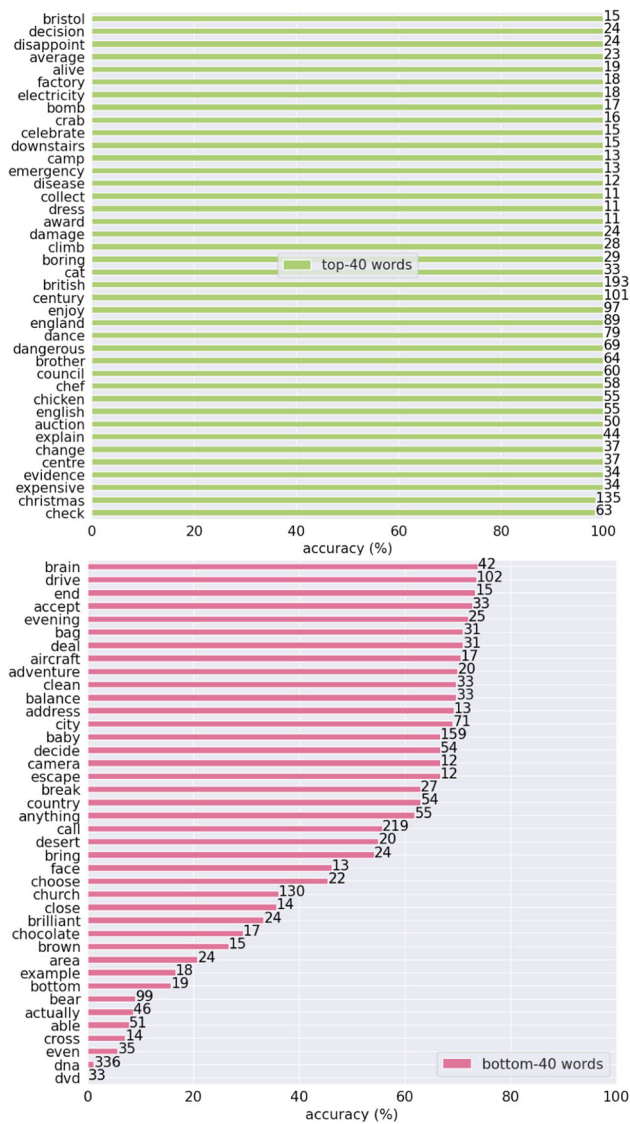
Fig. 10 Error analysis: We plot the per-word verification accuracies on BOBSL for top- and bottom-40 words sorted according to accuracies. Each bar also shows the total number of manual verifications performed for the corresponding word. While many words have close to 100% accuracy, certain words fail drastically such as 'dvd' and 'dna', mainly due to being fingerspelled signs. Other words such as 'even' and 'able' may have different meanings depending on context. Note that the query word here is obtained from subtitles (Color figure online)

Fig. 11 Qualitative analysis: We visualise sample spotting results from the manually verified set of BOBSL that obtain the highest similarity scores (overlaid on the spotted frame). Note that the query word is obtained from the subtitles. Top and bottom blocks represent success and failure cases, respectively. We notice weak hand shape and motion similarities in the failing examples. The figure only shows the middle frame for each video; therefore, we provide video visualisations at https://www.robots.ox.ac.uk/~vgg/research/bsldict/viz/viz_bobsl_dicts.html (Color figure online)

4.7 Results on the BOBSL Dataset

BOBSL is a dataset similar to BSL-1K; however, unlike BSL-1K, BOBSL is publicly available (Albanie et al. 2021). The dataset consists of 1400h of BSL-interpreted BBC broadcast footage accompanied by written English subtitles. We repeated our sign spotting techniques on this data using mouthing and dictionary cues in combination with subtitles. Keyword spotting with mouthing follows our previous work (Albanie et al. 2020) and obtains 502K sign localisations over 0.5 confidence ($M_{0.5}$). Sign spotting with dictionaries is similar to the procedure described in Sect. 4.6, resulting in 727K sign localisations over 0.75 similarity ($D_{0.75}$).

In Table 8, we present sign recognition results using these automatic annotations for classification training over a vocabulary of 2281 categories. The BOBSL test set contains 25,045 manually verified signs obtained through both types of spotting techniques. We experiment with various sets of annotations for training. We observe that mouthing (M) and dictionary (D) spottings are complementary. Similar to Table 7, we find that lowering the similarity threshold improves the performance for D-only training. However, when combined into a significantly larger training set (i.e., a total of 1.2 million clips with low thresholds), this improvement disappears (75.8% top-1 accuracy for both).

4.8 Limitations

In this section, we investigate failure modes of our sign spotting mechanism, in particular by using the data obtained through manual verifications. More specifically, we make statistics from 10K annotations on the BOBSL test set that were obtained via dictionary spotting through subtitles. From these, 76% were marked as correct. In Fig. 10, we present a breakdown for per-word accuracy to check whether certain signs fail more than others. We note two main failure modes: (i) fingerspelled words (e.g., 'dvd', 'dna') are difficult for the model, perhaps due to sparse frame sampling from long dictionary videos, (ii) common words such as 'even' and 'able' may have context-dependent meanings; querying these words due to occurrence in subtitles lead to false positives.

In Fig. 11, we further visualise samples from this manually verified set of spottings. We focus on cases where high similarities occur and group the examples into success (top) and failure (bottom) cases. Within failures, we observe weak hand shape and motion similarities. As previously mentioned, this might be due to querying a word for which a sign correspondence does not exist within the temporal search window. Future work may develop a mechanism to determine which words to query from the subtitle to ensure correspondence with a sign, so that the problem only becomes localisation.

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 (in our case, from mouthings), (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. We finally propose several potential applications of sign spotting and demonstrate its ability to scale up sign language datasets for training strong sign language recognition models.

Appendix

This appendix provides additional qualitative (Sect. A) and experimental results (Sect. B), as well as detailed explanations of the training of our Watch-Read-Lookup framework (Sect. C).

A Qualitative Results

Please watch our video in the project webpage[4] to see qualitative results of our model in action. We illustrate the sign spotting task, as well as the specific applications considered in the main paper: sign variant identification, densification of annotations, and "faux amis" identification between languages.

B Additional Experiments

In this section, we present complementary experimental results to the main paper. We report the effect of class-balancing (Sect. B.1), domain-specific layers (Sect. B.2), language-aware negative sampling (Sect. B.3), and the trunk network architecture (Sect. B.4).

B.1 Class-Balanced Sampling

As described in the main paper, we construct each batch by maximizing the number of negative pairs. To this end, we include one labelled sample per word when sampling continuous sequences, i.e., class-balancing the minibatches. Thus, all but one of the labelled samples in the batch can be used as negatives for a given dictionary bag corresponding to a labelled sample. Note that this approach limits the batch size to be less than or equal to the number of sign classes. Table A.1 experiments with the sampling strategy. We observe that the performance is not significantly different with/without class-balanced sampling for various batch sizes.

---

4 https://www.robots.ox.ac.uk/~vgg/research/bsldict/.

Table A.1 Class-balancing In the main paper, we class-balance our minibatches by including one sample per word from the labelled continuous sequences, thus maximizing the number of negatives within a batch

Class-balancing	Batch size	mAP	R@5
✗	512	41.65	54.73
✗	1024	42.07	54.25
✗	2048	43.14	54.28
✓	512	43.65	53.03
✓	1024	43.55	54.20

Here, we investigate removing such class-balancing constraint. In that case, we make sure we do not mark samples with the same labels as negatives, instead we discard them. We experiment with various batch sizes, also going beyond the total number of classes (2048). We observe that the performance is not significantly affected by these changes. (training on the full 1064 vocabulary with Watch-Lookup)

Table A.2 Domain-specific layers We experiment with separating the MLP layers to be specific to the continuous and isolated domains

Domain-specific layers	mAP	R@5
✓	43.58	53.54
✗	43.65	53.03

We do not observe any significant difference in performance and therefore adopt a shared MLP for simplicity in all experiments. (Training on the full 1064 vocabulary with Watch-Lookup)

B.2 Domain-Specific Layers

As noted in the main paper, the videos from the continuous signing and from the dictionaries differ significantly, e.g., continuous signing data is faster than the dictionary signing, and is co-articulated whereas the dictionary has isolated signs. Given such a domain gap, we explore whether it is beneficial to learn domain-specific MLP layers: one for the continuous, and one for the dictionary. Table A.2 presents a comparison between domain-specific layers versus shared parameters. We do not observe any gains from such separation. Therefore, we keep a single MLP for both domains for simplicity.

B.3 Language-Aware Negative Sampling

Working with a large vocabulary of words brings the additional challenge of handling synonyms. We consider two types of similarities. First, two different categories in the BSLDICT sign dictionary may belong to the same sign category if the corresponding English words are synonyms. Second, the meta-data we have collected with the BSLDICT dataset provides similarity labels between sign categories, which may be used to group certain signs. In this work, we have largely ignored this issue by associating each sign to a single word. This results in constructing negative pairs for

Table A.3 Language-aware negative sampling We explore the use of external knowledge such as English synonyms or the meta-data of the dictionary denoting similar sign categories

Negative sampling	mAP	R@5
Discarding English synonyms	43.27	54.24
Discarding Sign synonyms	45.03	54.19
Keeping all	43.65	53.03

We experiment with discarding such similar word pairs, excluding them from both positive and negative pairs. The last row instead marks any pair as negative if their corresponding words are not identical. We observe only marginal gains with the use of external knowledge about the languages. (Training on the full 1064 vocabulary with Watch-Lookup)

Table A.4 Trunk network architecture We compare I3D (Carreira and Zisserman 2017) with the S3D (Xie et al. 2018) architecture for the task of sign language recognition, in a comparable setup to Albanie et al. (2020)

Training data	Per-instance		Per-class	
	Top-1	Top-5	Top-1	Top-5
S3D	64.76	81.88	46.27	63.71
I3D Albanie et al. (2020)	**75.51**	**88.83**	**52.76**	**72.14**

We use the last 20 frames before the mouthing annotations with confidence above 0.5. We do not obtain gains with the S3D architecture; therefore, we use I3D in all the experiments to compute video features

two identical signs such as 'happy' and 'content'. Here, we explore whether it is beneficial to discard such pairs during training, instead of marking them as negatives. Table A.3 reports the results. We observe marginal gains with discarding synonyms. However, given the insignificant difference, we do not make such separation in other experiments for simplicity.

B.4 Trunk Network Architecture: S3D versus I3D

As shown in Table A.4, we compare two popular architectures for computing video representations. We have used I3D (Carreira and Zisserman 2017) in all our experiments. Here, we also train a 1064-way classification with the S3D architecture (Xie et al. 2018) on BSL-1K as in Albanie et al. (2020) for sign language recognition. We do not observe improvements with S3D (in practice we found that it overfit the training set to a greater degree); therefore, we use an I3D trunk. Note that the hyperparameters (e.g., learning rate) are tuned for I3D and kept the same for S3D.

C Training Details

In this section, we cover architectural details (Sect. C.1), a detailed formulation of our positive/negative bag sampling

strategy (Sect. C.2) and a brief description of the infrastructure used to perform the experiments in the main paper (Sect. C.3).

C.1 Architectural Details

As explained in the main paper, our sign embeddings correspond to the output of a two-stage architecture: (i) an I3D trunk, and (ii) a three-layer MLP. We first train the I3D on both labelled continuous video clips and the dictionary videos jointly. We then freeze the I3D trunk and use it as a feature extractor. We only train the MLP layers with our loss formulation in the Watch-Read-Lookup framework.

I3D trunk. We first train the I3D parameters only with the BSL-1K annotated clips that have mouthing confidences more than 0.5. For 1064-class training, we use the publicly available model from Albanie et al. (2020); for 800-class training, we perform our own training, also first pretraining with pose distillation.

We then *re-initialise the batch normalization layers* (as noted in We fine-tune the model jointly on BSL-1K annotated clips (the ones with mouthing confidence more than 0.8) and BSLDICT samples. The sampling frequency for the two data sources are balanced. In the I3D classification pretraining phase, we treat each dictionary video independently with its corresponding label. We observe that the 1064-way classification performance on the *training* dictionary videos remain at 48.09% per-instance top-1 accuracy without the batch normalization re-initialization, as opposed to 78.94%. We also experimented with domain-specific batch normal-

Fig. 12 MLP architecture: We detail the layers of our embedding architecture. We freeze the I3D trunk and use it as a feature extractor. We only train the MLP layers with our loss formulation in the proposed framework. The same layers (and parameters) are used both for the dictionary video inputs and the continuous signing video inputs (Color figure online)

ization layers (Chang et al. 2019), but the training accuracy for the dictionary videos was still low (62.73%).
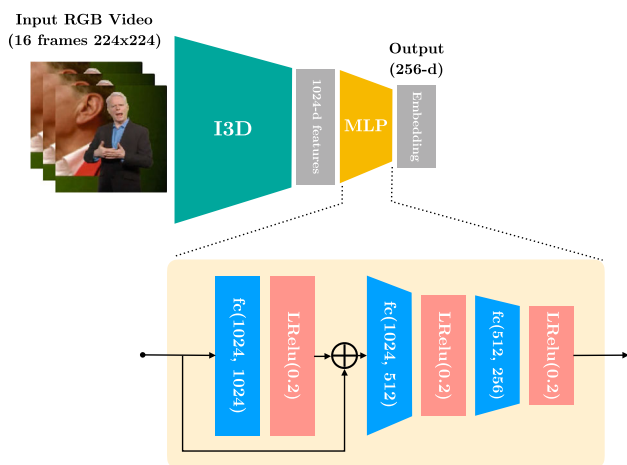
As detailed in we subsample the dictionary videos to roughly match their speed to the continuous signing videos. This subsampling includes *a random shift and a random fps*. We observe a decrease of 6.68% in the training dictionary classification accuracy if we instead sample 16 consecutive frames from the original temporal resolution, which is not sufficient to capture the full extent of a sign because one dictionary video is 56 frames on average.

MLP Fig. 12 illustrates the layers considered for our MLP architecture. It consists of 3 fully connected layers with LeakyRelu activations between them. The first linear layer also has a residual connection on the 1024-dimensional input features. We then reduce the dimensionality gradually to 512 and 256 for efficient training and testing.

C.2 Positive/Negative Bag Sampling Formulations

In the main paper, we described two approaches for sampling positive/negative MIL bags in Due to space constraints, the sampling mechanisms were described at a high-level. Here, we provide more precise definitions of each bag. In addition to the set notation below, we include in the code release, the loss implementation as a PyTorch (Paszke et al. 2019) function in `loss/loss.py`, together with a sample input (`loss/sample_inputs.pkl`) comprising embedding outputs from the MLP for continuous and dictionary videos.

As noted in the main paper, we do not have access to positive pairs because: (1) for the segments of videos in $\mathcal{S}$ that are annotated (i.e. $(x_k, v_k) \in \mathcal{M}$), 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 segment from a video from $\mathcal{S}$ (or a video from $\mathcal{M}$ when labels are available) and a video $\mathcal{D}$ to contain the same sign, and negative bags of pairs, $\mathcal{N}^{\text{bags}}$, in which we expect no pair of video segments from $\mathcal{S}$ (or $\mathcal{M}$) and $\mathcal{D}$ 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. Each bag is formulated as a set of paired indices—the first value indexes into the collections of continuous signing videos (either $\mathcal{S}$ or $\mathcal{M}$, depending on context) and the second value indexes into the set of dictionary videos contained in $\mathcal{D}$.

Watch and Lookup: using sparse annotations and dictionaries. In the first formulation, *Watch-Lookup*, we only make use of $\mathcal{D}$ and $\mathcal{M}$ (and not $\mathcal{S}$) to learn the data representation f. We define positive bags in two ways: (1) by anchoring on the labelled segment

$$\mathcal{P}_{\text{watch,lookup}}^{\text{bags(seg)}} = \{\{i\} \times B_i : (x_i^{\mathcal{M}}, v_i^{\mathcal{M}}) \in \mathcal{M}, \\ (x_j^{\mathcal{D}}, v_j^{\mathcal{D}}) \in \mathcal{D}, B_i = \{j : v_j^{\mathcal{D}} = v_i^{\mathcal{M}}\}\} \quad (2)$$

i.e. each bag consists of a labelled temporal segment and the set of sign variations of the corresponding word in the dictionary (illustrated in Fig. 13 (i), top row), or by (2) anchoring on the dictionary samples that correspond to the labelled segment, to define a second set $\mathcal{P}_{\text{watch,lookup}}^{\text{bags(dict)}}$, which takes a mathematically identical form to $\mathcal{P}_{\text{watch,lookup}}^{\text{bags(seg)}}$ (i.e. each bag consists of the set of sign variations of the word in the dictionary that corresponds to a given labelled temporal segment, illustrated in Fig. 13 (ii), top row). The key assumption in both cases is that each labelled segment matches *at least one* sign variation in the dictionary. Negative bags can be constructed by (1) anchoring on labelled segments and selecting dictionary examples corresponding to different words (Fig. 13 (i), red examples); (2) anchoring on the dictionary set for a given word and selecting labelled segments of a different word (Fig. 13 (ii), red example). These sets manifest as

$$\mathcal{N}_{\text{watch,lookup}}^{\text{bags(seg)}} = \{\{i\} \times B_i : (x_i^{\mathcal{M}}, v_i^{\mathcal{M}}) \in \mathcal{M}, \\ (x_j^{\mathcal{D}}, v_j^{\mathcal{D}}) \in \mathcal{D}, B_i = \{j : v_j^{\mathcal{D}} \neq v_i^{\mathcal{M}}\}\} \quad (3)$$

for the former and as

$$\mathcal{N}_{\text{watch,lookup}}^{\text{bags(dict)}} = \{A_i \times B_i : \\ A_i = \{l : x_l, x_i \subseteq x_k, (x_k, s_k) \in \mathcal{S}, x_l \cap x_i = \emptyset\}, \\ B_i = \{j : v_j^{\mathcal{D}} \neq v_i^{\mathcal{M}}\}, (x_i^{\mathcal{M}}, v_i^{\mathcal{M}}) \in \mathcal{M}, (x_j^{\mathcal{D}}, v_j^{\mathcal{D}}) \in \mathcal{D}\}\}. \quad (4)$$

for the latter. The complete set of positive and negative bags is formed via the unions of these collections:

$$\mathcal{P}_{\text{watch,lookup}}^{\text{bags}} \triangleq \mathcal{P}_{\text{watch,lookup}}^{\text{bags(seg)}} \cup \mathcal{P}_{\text{watch,lookup}}^{\text{bags(dict)}} \quad (5)$$

and

$$\mathcal{N}_{\text{watch,lookup}}^{\text{bags}} \triangleq \mathcal{N}_{\text{watch,lookup}}^{\text{bags(seg)}} \cup \mathcal{N}_{\text{watch,lookup}}^{\text{bags(dict)}}. \quad (6)$$

Watch, Read and Lookup. The *Watch-Lookup* bag formulation defined above has a significant limitation: the data representation, f, is not encouraged to represent signs beyond the initial vocabulary represented in $\mathcal{M}$. We therefore look at the subtitles present in $\mathcal{S}$ (which contain words beyond $\mathcal{M}$) in addition to $\mathcal{M}$ to construct bags. To do so, we introduce

an additional piece of terminology—when considering a subtitled video for which only one segment is labelled, we use the term "foreground" to refer to the subtitle word that corresponds to the label, and "background" for words which do not possess labelled segments in the video. Similarly to *Watch-Lookup*, we can construct positive bags, $\mathcal{P}_{\text{watch,lookup}}^{\text{bags}}$ (Fig. 14 (i) and (ii), top rows) which correspond to the use of foreground subtitle words. However, these can now by extended by (a) anchoring on a background segment in the continuous footage and find candidate matches in the dictionary among all possible matches for the subtitles words (Fig. 14 (iii), top row) and (b) anchoring on dictionary entries for background subtitle words (Fig. 14 (iv), top row). Formally, let Tokenize$(\cdot) : \mathcal{S} \to \mathcal{V}_{\mathfrak{L}}$ denote the function which extracts words from the subtitle that are present in the vocabulary: Tokenize$(s) \triangleq \{w \in s : w \in \mathcal{V}_{\mathfrak{L}}\}$. Then define background segment-anchored positive bags as:

$$\mathcal{P}_{\text{watch,read,lookup}}^{\text{bags(seg-back)}} = \{\{i\} \times B_i : \\ \exists (x_k, s_k) \in \mathcal{S} \text{ s.t } x_i \subseteq x_k, (x_j^{\mathcal{D}}, v_j^{\mathcal{D}}) \in \mathcal{D}, \\ B_i = \{j : v_j^{\mathcal{D}} \in \text{Tokenize}(s_k)\}, (x_i, v_i) \notin \mathcal{M}\}\} \quad (7)$$

i.e. each bag contains a background segment from the continuous signing which is paired with all dictionary segments whose labels match any token from the corresponding subtitle sentence (visualised as the top row of Fig. 14 (iii)). Next, we define dictionary-anchored positive background bags as follows:

$$\mathcal{P}_{\text{watch,read,lookup}}^{\text{bags(dict-back)}} = \{A_i \times B_i : (x_i^{\mathcal{D}}, v_i^{\mathcal{D}}) \in \mathcal{D}, \\ A_i = \{j : v_i^{\mathcal{D}} \in \text{Tokenize}(s_k), (x_k, s_k) \in \mathcal{S}, x_j \subseteq x_k, \\ (x_j, v_j) \notin \mathcal{M}\}, B_i = \{l : v_l^{\mathcal{D}} = v_i^{\mathcal{D}}\}\} \quad (8)$$
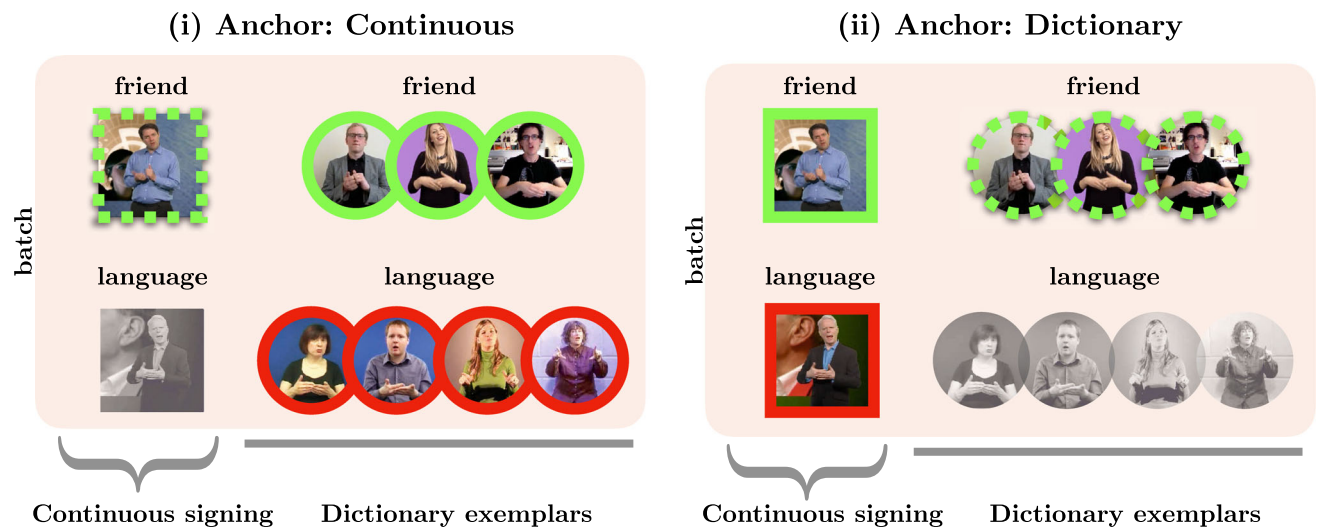
i.e. the bags contain all pairwise combinations of dictionary entries for a given word and segments in continuous signing whose subtitle contains that background word (visualised as top row of Fig. 14 (iv)). We combine these bags with the *Watch-Lookup* positive bags to maximally exploit the available supervisory signal for positives:

$$\mathcal{P}_{\text{watch,read,lookup}}^{\text{bags}} = \mathcal{P}_{\text{watch,lookup}}^{\text{bags}} \cup \mathcal{P}_{\text{watch,read,lookup}}^{\text{bags(seg-back)}} \\ \cup \mathcal{P}_{\text{watch,read,lookup}}^{\text{bags(dict-back)}}. \quad (9)$$

To counterbalance the positives, we use $\mathcal{S}$ in combination with $\mathcal{M}$ and $\mathcal{D}$ to create four kinds of negative bags. Differently to positive sampling, negatives can be constructed across the full minibatch rather than solely from the current (subtitled video, dictionary) pairing. We first anchor negatives bags on foreground segments:

$$\mathcal{N}_{\text{watch,read,lookup}}^{\text{bags(seg-fore)}} = \{\{i\} \times B_i : (x_i^{\mathcal{M}}, v_i^{\mathcal{M}}) \in \mathcal{M},$$

(i) Anchor: Continuous

friend   friend

language   language

Continuous signing   Dictionary exemplars

(ii) Anchor: Dictionary

friend   friend

language   language

Continuous signing   Dictionary exemplars

Fig. 13 Watch-Lookup We illustrate the batch formation and positive/negative sampling for the simplified version of our framework which is not using the subtitles, but only performing Watch-Lookup. We define two sets of positive/negative pairs, anchoring at a different position in each case. Anchor is denoted with dashed lines, positive samples with solid green, negative samples with solid red lines. Gray samples are discarded. (i) anchors at a labelled continuous video, mak-

ing the dictionary samples for the labelled word a positive bag, and all other dictionary samples in the batch a negative bag. (ii) anchors at a bag of dictionary samples, making the corresponding continuous labelled video positive, and all others in the batch negatives. We refer to Fig 14 for the illustration of our Watch-Read-Lookup extension (Color figure online)

$$(x_j^{\mathcal{D}}, v_j^{\mathcal{D}}) \in \mathcal{D}, B_i = \{j : v_j^{\mathcal{D}} \neq v_i^{\mathcal{M}}\}\} \qquad (10)$$

so that they contain pairs between a given foreground segment and all available dictionary videos whose label does not match the segment (visualised in Fig. 14 (i), both rows). We next anchor on the foreground dictionary videos:

$$
\begin{aligned}
\mathcal{N}_{\text{watch,read,lookup}}^{\text{bags(dict-fore)}} = \{A_i \times B_i : (x_i^{\mathcal{D}}, v_i^{\mathcal{D}}) \in \mathcal{D}, \\
A_i = \{j : v_i^{\mathcal{D}} \in \text{Tokenize}(s_k), (x_k, s_k) \in \mathcal{S}, x_j \subseteq x_k, \\
(x_j, v_j) \notin \mathcal{M}\} \cup \{(x_m, v_m) \in \mathcal{M}, v_m \neq v_i\}, \\
B_i = \{l : v_l^{\mathcal{D}} = v_i^{\mathcal{D}}\}\}
\end{aligned}
\qquad (11)
$$

comprising of pairings between the dictionary foreground set and segments within the minibatch that are either labelled with a different word, or can be excluded as a potential match through the subtitles (Fig. 14 (ii), both rows). Next, we anchor on the background continuous segments:

$$
\begin{aligned}
\mathcal{N}_{\text{watch,read,lookup}}^{\text{bags(seg-back)}} = \{\{i\} \times B_i : \exists (x_k, s_k) \in \mathcal{S}, x_i \subseteq x_k, \\
(x_j^{\mathcal{D}}, v_j^{\mathcal{D}}) \in \mathcal{D}, B_i = \{j : v_j^{\mathcal{D}} \notin \text{Tokenize}(s_k)\}\}
\end{aligned}
\qquad (12)
$$

which amounts to the pairings between each background segment and the set of dictionary videos which do not correspond to any of the words in the background subtitles (Fig. 14 (iii), both rows). The fourth negative bag set construction

anchors on the background dictionaries:

$$
\begin{aligned}
\mathcal{N}_{\text{watch,read,lookup}}^{\text{bags(dict-back)}} = \{A_i \times B_i : (x_i^{\mathcal{D}}, v_i^{\mathcal{D}}) \in \mathcal{D}, \\
A_i = \{j : v_i^{\mathcal{D}} \notin \text{Tokenize}(s_k), (x_k, s_k) \in \mathcal{S}, \\
x_j \subseteq x_k, (x_j, v_j) \notin \mathcal{M}\} \cup \{(x_m, v_m) \in \mathcal{M}, v_m \neq v_i\}, \\
B_i = \{l : v_l^{\mathcal{D}} = v_i^{\mathcal{D}}\}\}
\end{aligned}
\qquad (13)
$$

and thus the pairings arise between dictionary examples for a background segment and its corresponding foreground segment, as well all segments from other batch elements (Fig. 14 (iv), both rows). These four sets of bags are combined to form the full negative bag set:

$$
\begin{aligned}
\mathcal{N}_{\text{watch,read,lookup}}^{\text{bags}} = \mathcal{N}_{\text{watch,read,lookup}}^{\text{bags(seg-fore)}} \cup \mathcal{N}_{\text{watch,read,lookup}}^{\text{bags(seg-dict)}} \\
\cup \mathcal{N}_{\text{watch,read,lookup}}^{\text{bags(seg-back)}} \cup \mathcal{N}_{\text{watch,read,lookup}}^{\text{bags(dict-back)}}.
\end{aligned}
\qquad (14)
$$

In the main paper, these bag formulations are used through Eq. (1) (the MIL-NCE loss function) to guide learning. Concretely, the *Watch-Lookup* framework defines positive and negative bags via $\mathcal{P}^{\text{bags}} = \mathcal{P}_{\text{watch,lookup}}^{\text{bags}}$, $\mathcal{N}^{\text{bags}} = \mathcal{N}_{\text{watch,lookup}}^{\text{bags}}$ and the *Watch-Read-Lookup* formulation instead defines the positive and negative bags via $\mathcal{P}^{\text{bags}} = \mathcal{P}_{\text{watch,read,lookup}}^{\text{bags}}, \mathcal{N}^{\text{bags}} = \mathcal{N}_{\text{watch,read,lookup}}^{\text{bags}}$.
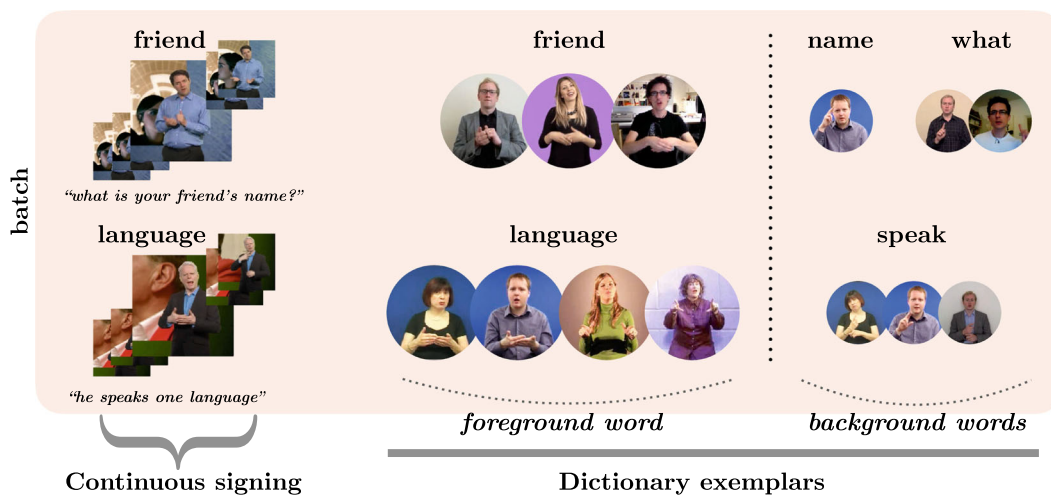
(a) Input: We illustrate an example minibatch formation for our Watch-Read-Lookup framework. We sample continuous videos with only one labelled segment, which we refer to as the 'foreground' word (e.g., *friend, language*). Each continuous video has a subtitle, which we use to sample additional words for which we do not have continuous signing labels, ('background' words), e.g. *name* and *what* for *"what is your friend's name?"*. We sample all the dictionary videos corresponding to these words. Each word has multiple dictionary instances grouped into overlapping circles.
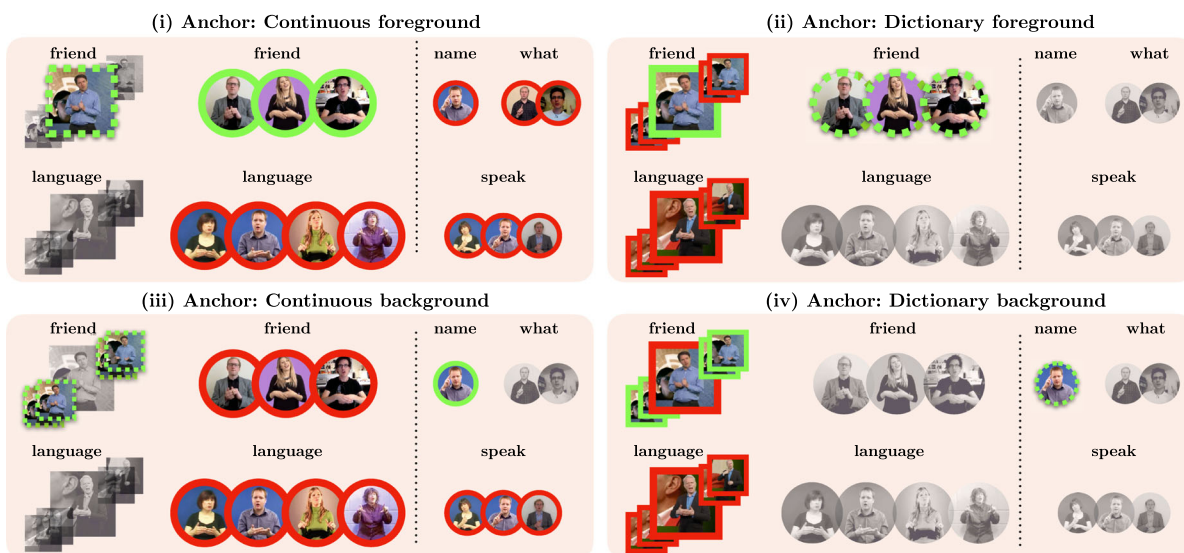
(b) Sampling positive/negative pairs: We anchor at 4 different positions within the batch to determine the pairs. Anchors are denoted with dashed lines, positive samples with solid green, negative samples with solid red lines. Gray samples are discarded. For example, (iii) anchoring at the continuous background marks the dictionary video for *name* positive, because it appears in the subtitle, but it is not within the annotated temporal window. All other dictionary samples *friend, language, speak* become negative to this anchor. We repeat this for each dictionary background, i.e., marking *what* as positive. See text for detailed explanations on each case. We also provide a video animation at our project page to show all possible positive/negative pairs for cases (i) to (iv).

Fig. 14 Watch-Read-Lookup in detail

C.3 Infrastructure and Runtime

Training. The I3D trunk BSL-1K pretraining experiments were performed with four Nvidia M40 graphics cards and took 2-3 days to complete. After freezing the I3D trunk, training the parameters of the MLP with the *Watch-Read-Lookup*

framework took approximately two hours on a single Nvidia M40 graphics card.

Inference. Our sign spotting demo available online (link at our project page) runs at real time in case of GPU availability. A single forward pass from the I3D and MLP layers takes 0.016 s to process 16 video frames on a single Nvidia

M40 GPU, which is roughly 1000 frames per second (much more than the 25 fps real time capture speed). However, our current models (both for spotting and recognition) rely on the I3D model, which is a 3D convolutional neural network with about 15M parameters. Future work can focus on compressing these heavy models into more lightweight architectures for mobile applications.

References

Afouras, T., Chung, J.S., & Zisserman, A. (2018). *LRS3-TED: a large-scale dataset for visual speech recognition.* arXiv preprint arXiv:1809.00496

Agris, U., Zieren, J., Canzler, U., Bauer, B., & Kraiss, K. F. (2008). Recent developments in visual sign language recognition. *Universal Access in the Information Society, 6,* 323–362.

Albanie, S., Varol, G., Momeni, L., Afouras, T., Bull, H., Chowdhury, H., Fox, N., Woll, B., Cooper, R., McParland, A., & Zisserman, A. (2021). *BOBSL: BBC-Oxford british sign language dataset.* arXiv preprint arXiv:2111.03635, https://www.robots.ox.ac.uk/~vgg/data/bobsl

Albanie, S., Varol, G., Momeni, L., Afouras, T., Chung, J.S., Fox, N., & Zisserman, A. (2020). BSL-1K: Scaling up co-articulated sign language recognition using mouthing cues. In *ECCV*

Aldersson, R., & McEntee-Atalianis, L. (2007). A lexical comparison of Icelandic sign language and Danish sign language. *Birkbeck Studies in Applied Linguistics,2.*

Athitsos, V., Neidle, C., Sclaroff, S., Nash, J., Stefan, A., Quan Yuan, & Thangali, A. (2008). The american sign language lexicon video dataset. In *CVPRW.*

Bank, R., Crasborn, O., & Hout, R. (2011). Variation in mouth actions with manual signs in sign language of the Netherlands (ngt). *Sign Language & Linguistics, 14,* 248–270.

Bilge, Y.C., Ikizler-Cinbis, N., & Gokberk Cinbis, R. (2019). Zero-shot sign language recognition: Can textual data uncover sign languages? In K. Sidorov & Yulia Hicks (Eds.), *Proceedings of the British Machine Vision Conference (BMVC)* (pp. 16.1–16.14). BMVA Press.

British sign language dictionary. https://www.signbsl.com/

Buehler, P., Everingham, M., & Zisserman, A. (2009). Learning sign language by watching TV (using weakly aligned subtitles). In *Proceedings of CVPR.*

Camgoz, N.C., Hadfield, S., Koller, O., & Bowden, R. (2017). Sub-UNets: end-to-end hand shape and continuous sign language recognition. In *ICCV.*

Camgoz, N.C., Koller, O., Hadfield, S., & Bowden, R. (2020). Sign language transformers: joint end-to-end sign language recognition and translation. In *CVPR.*

Cao, Z., Hidalgo, G., Simon, T., Wei, S.E., & Sheikh, Y. (2018). *OpenPose: realtime multi-person 2D pose estimation using part affinity fields.* arXiv preprint arXiv:1812.08008

Cao, K., Ji, J., Cao, Z., Chang, C.Y., & Niebles, J.C. (2020). Few-shot video classification via temporal alignment. In *CVPR.*

Cao, Q., Shen, L., Xie, W., Parkhi, O.M., & Zisserman, A. (2018). VGGFace2: A dataset for recognising faces across pose and age. In *Proceedings of International Conference on Automatic Face & Gesture Recognition*

Carreira, J., & Zisserman, A. (2017). Quo vadis, action recognition? A new model and the Kinetics dataset. In *CVPR.*

Chai, X., Wang, H., & Chen, X. (2014). The devisign large vocabulary of chinese sign language database and baseline evaluations. *Technical report VIPL-TR-14-SLR-001. Key lab of intelligent information processing of chinese academy of sciences (CAS),* Institute of Computing Technology, CAS.

Chang, W.G., You, T., Seo, S., Kwak, S., & Han, B. (2019). Domain-specific batch normalization for unsupervised domain adaptation. In *CVPR.*

Chung, J.S., & Zisserman, A. (2016). Signs in time: Encoding human motion as a temporal image. In *Workshop on Brave New Ideas for Motion Representations*, ECCV.

Chung, J.S., Senior, A., Vinyals, O., & Zisserman, A. (2017). Lip reading sentences in the wild. In *CVPR.*

Cooper, H., & Bowden, R. (2009). Learning signs from subtitles: a weakly supervised approach to sign language recognition. In *CVPR.*

Cooper, H., Pugeault, N., & Bowden, R. (2011). Reading the signs: A video based sign dictionary. In *ICCVW.*

Coucke, A., Chlieh, M., Gisselbrecht, T., Leroy, D., Poumeyrol, M., & Lavril, T. (2019). Efficient keyword spotting using dilated convolutions and gating. In *ICASSP.*

Dietterich, T. G., Lathrop, R. H., & Lozano-Pérez, T. (1997). Solving the multiple instance problem with axis-parallel rectangles. *Artificial Intelligence, 89*(1–2), 31–71.

Dutta, A., & Zisserman, A. (2019). The via annotation software for images, audio and video. In *Proceedings of ACMM.* MM 19, vol. 27. ACM, ACM, New York, USA. https://doi.org/10.1145/3343031.3350535, to appear in Proceedings of the 27th ACM international conference on multimedia (MM 19).

Eng-Jon Ong, Koller, O., Pugeault, N., & Bowden, R. (2014). Sign spotting using hierarchical sequential patterns with temporal intervals. In *CVPR.*

Farhadi, A., Forsyth, D.A., & White, R. (2007). Transfer learning in sign language. In *CVPR.*

Feng, Y., Ma, L., Liu, W., Zhang, T., & Luo, J. (2018). Video re-localization. In *ECCV.*

Fenlon, J., Cormier, K., Rentelis, R., Schembri, A., Rowley, K., Adam, R., & Woll, B. (2014). *BSL SignBank: a lexical database and dictionary of British Sign Language (first edition).* London: Deafness, Cognition and Language Research Centre, University College London.

Fillbrandt, H., Akyol, S., & Kraiss, K. (2003). Extraction of 3D hand shape and posture from image sequences for sign language recognition. In *IEEE International SOI Conference.*

Forster, J., Oberdörfer, C., Koller, O., & Ney, H. (2013). Modality combination techniques for continuous sign language recognition. In *Pattern recognition and image analysis.*

Gutmann, M., & Hyvärinen, A. (2010). Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics,* (pp. 297–304).

He, K., Fan, H., Wu, Y., Xie, S., & Girshick, R. (2020). Momentum contrast for unsupervised visual representation learning. In *CVPR.*

Hu, J., Shen, L., Albanie, S., Sun, G., & Wu, E. (2020). Squeeze-and-excitation networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *42*(8), 2011–2023.

Huang, J., Zhou, W., Zhang, Q., Li, H., & Li, W. (2018). Video-based sign language recognition without temporal segmentation. In *AAAI.*

Joze, H.R.V., & Koller, O. (2019). MS-ASL: A large-scale data set and benchmark for understanding american sign language. In *BMVC.*

Kadir, T., Bowden, R., Ong, E.J., & Zisserman, A. (2004). Minimal training, large lexicon, unconstrained sign language recognition. In *Proceedings of BMVC.*

Koller, O., Forster, J., & Ney, H. (2015). Continuous sign language recognition: towards large vocabulary statistical recognition systems handling multiple signers. *Computer Vision and Image Understanding, 141,* 108–125.

Li, D., Opazo, C.R., Yu, X., & Li, H. (2019). Word-level deep sign language recognition from video: a new large-scale dataset and methods comparison. In *WACV*.

Li, D., Yu, X., Xu, C., Petersson, L., & Li, H. (2020). Transferring cross-domain knowledge for video sign language recognition. In *CVPR*.

Miech, A., Alayrac, J.B., Smaira, L., Laptev, I., Sivic, J., & Zisserman, A. (2020). End-to-end learning of visual representations from uncurated instructional videos. In *CVPR*.

Momeni, L., Afouras, T., Stafylakis, T., Albanie, S., & Zisserman, A. (2020). Seeing wake words: Audio-visual keyword spotting. In *BMVC*.

Momeni, L., Varol, G., Albanie, S., Afouras, T., & Zisserman, A. (2020). Watch, read and lookup: learning to spot signs from multiple supervisors. In *ACCV*.

Motiian, S., Jones, Q., Iranmanesh, S.M., & Doretto, G. (2017). Few-shot adversarial domain adaptation. In *NeurIPS*.

Ong, E., Cooper, H., Pugeault, N., & Bowden, R. (2012). Sign language recognition using sequential pattern trees. In *CVPR*.

Oord, A.v.d., Li, Y., & Vinyals, O. (2018). *Representation learning with contrastive predictive coding*. arXiv preprint arXiv:1807.03748.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., & Chintala, S. (2019). Pytorch: an imperative style, high-performance deep learning library. In *NeurIPS*.

Pfister, T., Charles, J., & Zisserman, A. (2013). Large-scale learning of sign language by watching tv (using co-occurrences). In *BMVC*.

Pfister, T., Charles, J., & Zisserman, A. (2014). Domain-adaptive discriminative one-shot learning of gestures. In *Proceedings of ECCV*.

Schembri, A., Fenlon, J., Rentelis, R., & Cormier, K. (2017). *British sign language corpus project: a corpus of digital video data and annotations of British Sign Language 2008-2017 (Third Edition)*, http://www.bslcorpusproject.org.

Schembri, A., Fenlon, J., Rentelis, R., Reynolds, S., & Cormier, K. (2013). Building the British sign language corpus. *Language Documentation & Conservation, 7,* 136–154.

SignumMcKee, D., & Kennedy, G. (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*.

Stafylakis, T., & Tzimiropoulos, G. (2018). Zero-shot keyword spotting for visual speech recognition in-the-wild. In *ECCV*.

Starner, T. (1995). *Visual recognition of American sign language using hidden markov models*. Master's thesis, Massachusetts Institute of Technology.

Sutton-Spence, R. (2007). Mouthings and simultaneity in British sign language. In *Simultaneity in signed languages: form and function,* (pp. 147–162). John Benjamins.

Sutton-Spence, R., & Woll, B. (1999). *The Linguistics of British sign language: an introduction*. Cambridge: Cambridge University Press.

Tamura, S., & Kawasaki, S. (1988). Recognition of sign language motion images. *Pattern Recognition, 21*(4), 343–353.

Varol, G., Momeni, L., Albanie, S., Afouras, T., & Zisserman, A. (2021). Read and attend: temporal localisation in sign language videos. In *CVPR*.

Véniat, T., Schwander, O., & Denoyer, L. (2019). Stochastic adaptive neural architecture search for keyword spotting. In *ICASSP*.

Viitaniemi, V., Jantunen, T., Savolainen, L., Karppa, M., & Laaksonen, J. (2014). S-pot - a benchmark in spotting signs within continuous signing. In *LREC*.

von Agris, U., Knorr, M., & Kraiss, K. (2008). The significance of facial features for automatic sign language recognition. In *2008 8th IEEE international conference on automatic face gesture recognition*.

Wilbur, R.B., & Kak, A.C. (2006). Purdue RVL-SLLL American sign language database. *School of electrical and computer engineering technical report*, TR-06-12, Purdue University, W. Lafayette, IN 47906.

Wu, Z., Xiong, Y., Yu, S.X., & Lin, D. (2018). Unsupervised feature learning via non-parametric instance discrimination. In *CVPR*.

Xie, S., Sun, C., Huang, J., Tu, Z., & Murphy, K. (2018). Rethinking spatiotemporal feature learning for video understanding. In *ECCV*.

Yang, H., He, X., & Porikli, F. (2018). One-shot action localization by learning sequence matching network. In *CVPR*.

Ye, Y., Tian, Y., Huenerfauth, M., & Liu, J. (2018). Recognizing american sign language gestures from within continuous videos. In *CVPRW*.

Zhang, J., Chen, Z., Huang, J., Lin, L., & Zhang, D. (2019). Few-shot structured domain adaptation for virtual-to-real scene parsing. In *ICCVW*.

Zhou, H., Zhou, W., Zhou, Y., & Li, H. (2020). Spatial-temporal multi-cue network for continuous sign language recognition. CoRR **abs/2002.03187**