# *Speech2Action:* Cross-modal Supervision for Action Recognition

Arsha Nagrani[1,2] Chen Sun[2] David Ross[2]
Rahul Sukthankar[2] Cordelia Schmid[2] Andrew Zisserman[1,3]
[1]VGG, Oxford [2]Google Research [3]DeepMind

https://www.robots.ox.ac.uk/~vgg/research/speech2action/

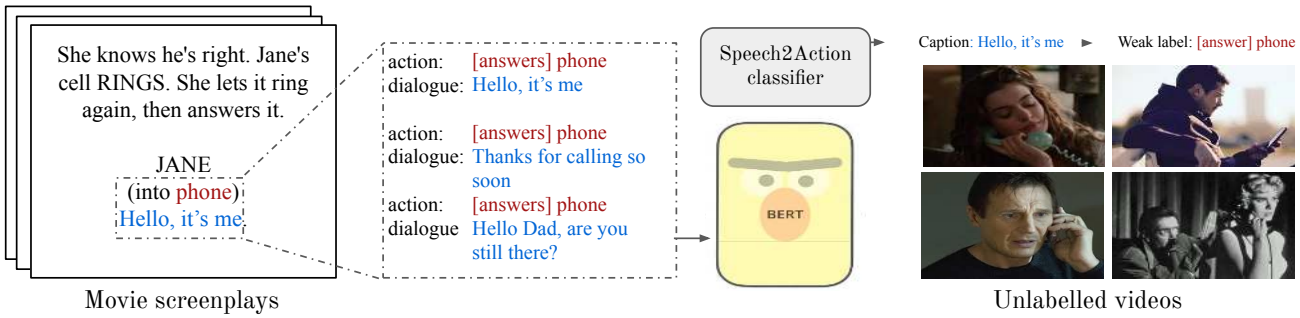


Figure 1. **Weakly Supervised Learning of Actions from Speech Alone:** The co-occurrence of speech and scene descriptions in movie screenplays (text) is used to learn a `Speech2Action` model that predicts actions from transcribed speech *alone*. Weak labels for visual actions can then be obtained by applying this model to the speech in a large *unlabelled* set of movies.

## Abstract

*Is it possible to guess human action from dialogue alone? In this work we investigate the link between spoken words and actions in movies. We note that movie screenplays describe actions, as well as contain the speech of characters and hence can be used to learn this correlation with no additional supervision. We train a BERT-based* `Speech2Action` *classifier on over a thousand movie screenplays, to predict action labels from transcribed speech segments.*

*We then apply this model to the speech segments of a large* `unlabelled` *movie corpus (188M speech segments from 288K movies). Using the predictions of this model, we obtain weak action labels for over 800K video clips. By training on these video clips, we demonstrate superior action recognition performance on standard action recognition benchmarks, without using a single manually labelled action example.*

## 1. Introduction

Often, you can get a sense of human activity in a movie by listening to the dialogue alone. For example, the sentence *'Look at that spot over there'*, is an indication that somebody is pointing at something. Similarly, the words *'Hello, thanks for calling'*, is a good indication that somebody is speaking on the phone. Could this be a valuable source of information for learning good action recognition models?

Obtaining large scale human labelled video datasets to train models for visual action recognition is a notoriously challenging task. While large datasets, such as Kinetics [19] or Moments in Time [29] consisting of individual short clips (e.g. 10s) are now available, these datasets come at formidable human cost and effort. Furthermore, many such datasets suffer from heavily skewed distributions with long tails – i.e. it is difficult to obtain manual labels for *rare* or *infrequent* actions [14].

Recently, a number of works have creatively identified *certain domains* of videos, such as *narrated instructional videos* [27, 38, 51] and *lifestyle vlogs* [11, 17] that are available in huge numbers (e.g. on YouTube) and often contain narration with the explicit intention of explaining the visual content on screen. In these video domains, there is a direct link between the action being performed, and the speech accompanying the video – though this link, and the visual supervision it provides, can be quite weak and 'noisy' as the speech may refer to previous or forthcoming visual events, or be about something else entirely [27].

In this paper we explore a complementary link between speech and actions in the more general domain of movies and TV shows (not restricted to instructional videos and vlogs). We ask: is it possible given only a speech sentence to predict whether an action is happening, and, if so, what

the action is? While it appears that in some cases the speech is correlated with action – *'Raise your glasses to . . . '*, in the more general domain of movies and TV shows it is *more* likely that the speech is completely uncorrelated with the action – *'How is your day going?'*. Hence in this work, we *explicitly* learn to identify when the speech is discriminative. While the supervision we obtain from the speech–action correlation is still noisy, we show that at scale it can provide sufficient weak supervision to train visual classifiers (see Fig. 1).

Luckily, we have a large amount of literary content at our disposal to learn this correlation between speech and actions. Screenplays can be found for hundreds of movies and TV shows and contain rich descriptions of the identities of people, their actions and interactions with one another and their dialogue. Early work has attempted to *align* these screenplays to the videos themselves, and use that as a source of weak supervision [2, 8, 22, 25]. However, this is challenging due to the lack of explicit correspondence between scene elements in video and their textual descriptions in screenplays [2], and notwithstanding alignment quality, is also fundamentally limited in scale to the amount of aligned movie screenplays available. Instead we learn from *unaligned* movie screenplays. We *first* learn the correlation between speech and actions from written material *alone* and use this to train a `Speech2Action` classifier. This classifier is then applied to the speech in an unlabelled, unaligned set of videos to obtain visual samples corresponding to the actions confidently predicted from the speech (Fig. 1). In this manner, the correlations can provide us with an effectively infinite source of weak training data, since the audio is freely available with movies.

Concretely, we make the following four contributions: (i) We train a `Speech2Action` model from literary screenplays, and show that it is possible to predict certain actions from transcribed speech *alone* without the need for any manual labelling; (ii) We apply the `Speech2Action` model to a large unlabelled corpus of videos to obtain weak labels for video clips from the speech alone; (iii) We demonstrate that an action classifier trained with these weak labels achieves state of the art results for action classification when fine-tuned on standard benchmarks compared to other weakly supervised/domain transfer methods; (iv) Finally, and more interestingly, we evaluate the action classifier trained only on these weak labels with *no* fine-tuning on the mid and tail classes from the AVA dataset [14] in the zero-shot and few-shot setting, and show a large boost over fully supervised performance for some classes without using a *single* manually labelled example.

## 2. Related Works

**Aligning Screenplays to Movies:** A number of works have explored the use of screenplays to learn and automatically annotate character identity in TV series [5, 9, 30, 35, 39]. Learning human actions from screenplays has also been attempted [2, 8, 22, 25, 26]. Crucially, however, all these works rely on aligning these screenplays to the actual videos themselves, often using the speech (as subtitles) to provide correspondences. However, as noted by [2], obtaining supervision for actions in this manner is challenging due to the lack of explicit correspondence between scene elements in video and their textual descriptions in screenplays.

Apart from the imprecise temporal localization inferred from subtitles correspondences, a major limitation is that this method is not scalable to all movies and TV shows, since screenplays with stage directions are simply not available at the same order of magnitude. Hence previous works have been limited to a small scale, no more than *tens* of movies or a season of a TV series [2, 8, 22, 25, 26]. A similar argument can be applied to works that align books to movies [40, 52]. In contrast, we propose a method that can exploit the richness of information in a modest number of screenplays, and then be applied to a virtually limitless set of edited video material with no alignment or manual annotation required.

**Supervision for Action Recognition:** The benefits of learning from large scale supervised video datasets for the task of action recognition are well known, with the introduction of datasets like Kinetics [19] spurring the development of new network architectures yielding impressive performance gains, e.g. [4, 10, 41, 43, 44, 47]. However, as described in the introduction, such datasets come with an exorbitant labelling cost. Some work has attempted to reduce this labeling effort through heuristics [50] (although a human annotator is required to clean up the final labels) or by procuring weak labels in the form of accompanying meta data such as hashtags [12].

There has *also* been a recent growing interest in using *cross-modal supervision* from the audio streams readily available with videos [1, 20, 31, 32, 49]. Such methods, however, focus on *non-speech* audio, e.g. 'guitar playing', the 'thud' of a bouncing ball or the 'crash' of waves at the seaside, rather the transcribed speech. As discussed in the introduction, transcribed speech is used only in certain narrow domains, e.g. instruction videos [27, 38, 51] and lifestyle vlogs [11, 17], while in contrast to these works, we focus on the domain of movies and TV shows (where the link between speech and actions is less explicit). Further, such methods use most or *all* the speech accompanying a video to learn a better overall visual embedding, whereas we note that often the speech is completely uninformative of the action. Hence we *first* learn the correlation between speech and actions from written material, and then apply this knowledge to an unlabelled set of videos to obtain video clips that can be used directly for training.

## 3. Speech2Action Model

In this section we describe the steps in data preparation, data mining and learning, required to train the Speech2Action classifier from a large scale dataset of screenplays. We then assess its performance in predicting visual actions from transcribed speech segments.

### 3.1. The IMSDb Dataset

Movie screenplays are a rich source of data that contain both stage directions (*'Andrew walked over to open the door'*) and the dialogues spoken by the characters (*'Please come in'*). Since stage directions often contain described actions, we use the co-occurrence of dialogue and stage directions in screenplays to learn the relationship between 'actions' and dialogue (see Fig. 1). In this work, we use a corpus of screenplays extracted from IMSDb (www.imsdb.com). In order to get a wide variety of different actions ('push' and 'kick' as well as 'kiss' and 'hug') we use screenplays covering a range of different genres[1]. In total our dataset consists of 1,070 movie screenplays (statistics of the dataset can be seen in Table 1). We henceforth refer to this dataset as the IMSDb dataset.

**Screenplay Parsing:** While screenplays (generally) follow a standardized format for their parts (e.g., stage direction, dialogue, location, timing information etc.), they can be challenging to parse due to discrepancies in layout and format. We follow the grammar created by Winer et al. [45] which is based on 'The Hollywood Standard' [33], to parse the scripts and separate out various screenplay elements. The grammar provided by [45] parses scripts into the following four different elements, (1) Shot Headings, (2) Stage Directions (which contain mention of actions), (3) Dialogue and (4) Transitions. More details are provided in Sec. **??** of the suppl. material.

In this work we extract only (2) Stage Directions and (3) Dialogue. We extract over 500K stage directions and over 500K dialogue utterances (see Table 1). It is important to note that since screenplay parsing is done using an automatic method, and sometimes hand-typed screenplays follow completely non-standard formats, this extraction is not perfect. A quick manual inspection of 100 randomly extracted dialogues shows that around 85% of these are actually dialogue, with the rest being stage directions that have been wrongly labelled as dialogue.

**Verb Mining the Stage Directions:** Not all actions will be correlated with speech – e.g. actions like 'sitting' and 'standing' are difficult to distinguish based on speech alone, since they occur commonly with all types of speech. Hence our first endeavour is to automatically determine verbs rendered 'discriminative' by speech alone. For this we use the IMSDb dataset described above. We first take all the stage directions in the dataset, and break up each sentence into clean word tokens (devoid of punctuation). We then determine the part of speech (PoS) tag for each word using the NLTK toolkit [24] and obtain a list of all the verbs present. Verbs occurring fewer than 50 times (includes many spelling mistakes) or those occurring too frequently, i.e. the top 100 most frequent verbs (these are stop words like 'be' etc.) are removed. For each verb, we then group together all the conjugations and word forms for a particular word stem (e.g. the stem *run* can appear in many different forms – running, ran, runs etc.), using the manually created verb conjugations list from the UPenn XTag project[2]. All such verb classes are then used in training a BERT-based speech to action classifier, described next.

### 3.2. BERT-based Speech Classifier

Each stage direction is then parsed for verbs belonging to the verb classes identified above. We obtain *paired* speech-action data using proximity in the movie screenplays as a clue. Hence, the nearest speech segment to the stage direction (as illustrated in Fig. 1) is assigned a label for every verb in the stage direction (more examples in Fig. 2 of the suppl. material). This gives us a dataset of speech sentences matched to verb labels. As expected, this is a very noisy dataset. Often, the speech has no correlation with the verb class it is assigned to, and the same speech segment can be assigned to many different verb classes. To learn the correlation between speech and action, we train a classifier with 850 movies and use the remaining ones for validation. The classifier used is a pretrained BERT [7] model with an additional classification layer, finetuned on the dataset of speech paired with weak 'action' labels. Exact model details are described below.

**Implementation Details:** The model used is BERT-Large Cased with Whole-Word Masking (L=24, H=1024, A=16, Total Parameters=340M) [7] pretrained only on English data (BooksCorpus (800M words, [52]) and the Wikipedia corpus (2,500M words)), since the IMSDb dataset consists only of movie screenplays in English[3]. We use WordPiece embeddings [46] with a $30,000$ token vocabulary. The first token of every sequence is always a special classification token ([CLS]). We use the final hidden vector $C \in \mathbb{R}^H$ corresponding to the first input token ([CLS]) as the aggregate representation. The only new parameters introduced during fine-tuning are classification layer weights $W \in \mathbb{R}^{K \times H}$ where K is the number of classes. We use the standard cross-entropy loss with $C$ and $W$,

[1] Action, Adventure, Animation, Biography, Comedy, Crime, Drama, Family, Fantasy, Film-Noir, History, Horror, Music, Musical, Mystery, Romance, Sci-Fi, Short, Sport, Thriller, War, Western

[2] http://www.cis.upenn.edu/~xtag/

[3] The model can be found here: https://github.com/google-research/bert

| # movies | # scene descriptions | # speech segs | # sentences | # words | # unique words | # genres |
|---|---|---|---|---|---|---|
| 1,070 | 539,827 | 595,227 | 2,570,993 | 21,364,357 | 590,959 | 22 |

Table 1. **Statistics of the IMSDb dataset of movie screenplays.** This dataset is used to learn the correlation between speech and verbs. We use 850 screenplays for training and 220 for validation. Statistics for sentences and words are from the entire text of the screenplays.

| | | | | | |
|---|---|---|---|---|---|
| phone | Hello, it's me.<br>May I have the number for Dr George Shannan<br>Honey I asked you not to call unless what why<br>hey, it's me<br>Hello, it's me.<br>Hello? | kiss | One more kiss<br>Give me a kiss<br>Good night my darling<br>I love you my darling<br>Noone had ever kissed me there before<br>Goodnight angel my sweet boy | drink | To us<br>Raise your glasses to Charlie<br>Heres a toast<br>You want some water<br>Drink deep and live<br>Drink up its party time |
| dance | Shes a beautiful dancer<br>Waddaya say you wanna dance<br>Come on Ill take a break and well all dance<br>Ladies and Gentlemen the first dance<br>Excuse me would you care for this dance<br>Hattie do you still dance | drive | So well drop Rudy off at the bus<br>Ill drive her<br>just parking it out of the way<br>all you have to do is drop me off at the bank<br>Wait down the road<br>He drove around for a long long time driving | point | Officer Van Dorn is right down that hall<br>OK Print that one<br>the Metroplitan Museum of Art is right there<br>Over there<br>And her<br>The one with the black spot |

Figure 2. **Examples of the top ranked speech samples for six verb categories.** Each block shows the action verb on the left, and the speech samples on the right. All speech segments are from the validation set of the IMSDb dataset of movie screenplays.

i.e., $\log(\text{softmax}(W^T C))$. We use a batch size of 32 and finetune the model end-to-end on the IMSDb dataset for 100,000 iterations using the Adam solver with a learning rate of $5 \times 10^5$.

**Results:** We evaluate the performance of our model on the 220 movie screenplays in the val set. We plot the precision-recall curves using the softmax scores obtained from the `Speech2Action` model (Fig. 1 in suppl. material). Only those verbs that achieve an average precision (AP) higher than 0.01 are inferred to be correlated with speech. The highest performing verb classes are 'phone', 'open' and 'run', whereas verb classes like 'fishing' and 'dig' achieve a very low average precision. We finally conclude that there is a strong correlation for 18 verb classes.[4] Qualitative examples of the most confident predictions (using softmax score as a measure of confidence) for 6 verb classes can be seen in Fig. 2. We note here that we have learnt the correlation between action verb and speech from the movie screenplays using a purely data-driven method. The key assumption is that if there is a *consistent* trend of a verb appearing in the screenplays before or after a speech segment, and our model is able to exploit this trend to minimise a classification objective, we infer that the speech is correlated with the action verb. Because the evaluation is performed purely on the basis of the proximity of speech to verb class in the stage direction of the movie screenplay, it is *not* a perfect ground truth indication of whether an action will actually be performed in a *video* (which is impossible to say only from the movie scripts). We use the stage directions in this case as *pseudo* ground truth, i.e. if the stage direction contains an action and the actor then says a particular sentence, we infer that these two must be related. As a sanity check, we also manually annotate some videos in order to better assess the performance of the `Speech2Action` model. This is described in Sec. 4.2.3.

[4]The verb classes are: 'open', 'phone', 'kiss', 'hug', 'push', 'point', 'dance', 'drink', 'run', 'count', 'cook', 'shoot', 'drive', 'enter', 'fall', 'follow', 'hit', 'eat'.

## 4. Mining Videos for Action Recognition

Now that we have learned the `Speech2Action` model to map from transcribed speech to actions (from *text* alone), in this section we demonstrate how this can be applied to video. We use the model to automatically mine video examples from large, unlabelled corpora (the corpus is described in Sec. 4.1), and assign them with weak labels from the `Speech2Action` model prediction. Armed with this weakly labelled data, we then train models directly for the downstream task of visual action recognition. Detailed training and evaluation protocols for the mining are described in the following sections.

### 4.1. Unlabelled Data

In this work, we apply the `Speech2Action` model to a large internal corpus of movies and TV shows. The corpus consists of $222,855$ movies and TV show episodes. For these videos, we use the closed captions (note that this can be obtained from the audio track directly using automatic speech recognition). The total number of closed captions for this corpus is $188,210,008$, which after dividing into sentences gives us a total of $390,791,653$ (almost 400M) sentences. While we use this corpus in our work, we would like to stress here that there is no correlation between the text data used to train the `Speech2Action` model and this unlabelled corpus (other than both belonging to the movie domain), and such a model can be applied to any other corpus of unlabelled, edited film material.

### 4.2. Obtaining Weak Labels

In this section, we describe how we obtain weak action labels for short clips from the speech alone. We do this in two ways, (i) using the `Speech2Action` model, and (ii) using a simple keyword spotting baseline described below.

#### 4.2.1 Using `Speech2Action`

The `Speech2Action` model is applied to a single sentence of speech, and the prediction is used as a weak label if the confidence (softmax score) is above a certain threshold. The threshold is obtained by taking the confidence value at a precision of 0.3 on the IMSDb validation set, with some manual adjustments for the classes of 'phone', 'run' and 'open' (since these classes have a much higher recall, we increase the threshold in order to prevent a huge imbalance of retrieved samples). More details are provided in Sec. **??** in the suppl. material. We then extract the visual frames for a 10 second clip centered around the midpoint of the timeframe spanned by the caption, and assign the `Speech2Action` label as the weak label for the clip. Ultimately, we successfully end up mining $837,334$ video clips for 18 action classes. While this is a low yield, we still end up with a large number of mined clips, greater than the manually labelled Kinetics dataset [19] (600K).

We also discover that the verb classes that have high correlation with speech in the IMSDb dataset tend to be *infrequent* or *rare* actions in other datasets [14] – as shown in Fig. 3, we obtain two orders of magnitude more data for certain classes in the AVA training set [14]. Qualitative examples of mined video clips with action labels can be seen in Fig. 4. Note how we are able to retrieve clips with a wide variety in background and actor, simply from the speech alone. Refer to Fig. 5 in the suppl. material for more examples showing diversity in objects and viewpoint.

#### 4.2.2 Using a Keyword Spotting Baseline

In order to validate the efficacy of our `Speech2Action` model trained on movie screenplays, we also compare to a simple keyword spotting baseline. This involves searching for the action verb in the speech directly – a speech segment like *'Will you eat now?'* is directly assigned the label 'eat'. This itself is a very powerful baseline, e.g. speech segments such as *'Will you dance with me'*, are strongly indicative of the action 'dance'. To implement this baseline, we search for the presence of the action verb (or its conjugations) in the speech segment directly, and if the verb is present in the speech, we assign the action label to the video clip directly. The fallacy of this method is that there is no distinction between the different semantic meanings of a verb, e.g. the speech segment *'You've missed the point entirely'* will be weakly labelled with the verb 'point' using this baseline, which is indicative of a different semantic meaning to the physical action of 'pointing'. Hence as we show in the results, this baseline performs poorly compared to our `Speech2Action` mining method (Tables 4 and 3). More examples of speech labelled using this keyword spotting baseline can be seen in Table 1 in the suppl. material.

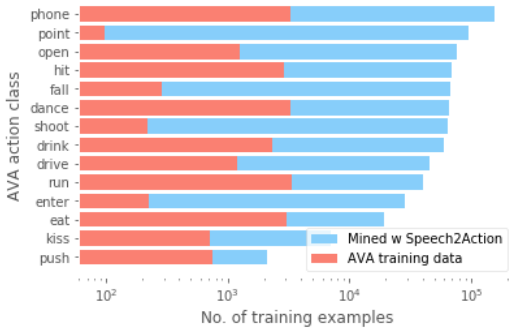


Figure 3. **Distribution of training clips mined using `Speech2Action`.** We compare the distribution of mined clips to the number of samples in the AVA training set. Although the mined clips are noisy, we are able to obtain far more, in some cases up to *two* orders of magnitude more training data (note the **log scale** in the x-axis).

| dance | phone | kiss | drive | eat | drink | run | point | hit | shoot |
|---|---|---|---|---|---|---|---|---|---|
| 42 | 68 | 18 | 41 | 27 | 51 | 83 | 52 | 18 | 27 |

Table 2. **Number of true positives for 100 randomly retrieved samples for 10 classes.** These estimates are obtained through manual inspection of video clips that are labelled with `Speech2Action`. While the true positive rate for some classes is low, the other samples still contain valuable information for the classifier. For example, although there are only 18 true samples of 'kiss', many of the other videos have two people with their lips very close together, or even if they are not 'eating' strictly, many times they are holding food in their hands.

#### 4.2.3 Manual Evaluation of `Speech2Action`

We now assess the performance of `Speech2Action` applied to videos. Given a speech segment, we check whether a prediction made by the model on the speech translates to the action being performed visually in the frames aligned to the speech. To assess this, we do a manual inspection of a random set of 100 retrieved video clips for 10 of the verb classes, and report the true positive rate (number of clips for which the action is visible) in Table 2. We find that a surprising number of samples actually contain the action during the time frame of 10 seconds, with some classes noisier than others. The high purity of the classes 'run' and 'phone' can be explained by the higher thresholds used for mining,


Figure 4. **Examples of clips mined automatically using the `Speech2Action` model applied to *speech alone* for 8 AVA classes.** We show only a single frame from each video. Note the diversity in background, actor and view point. We show false positives for eat, phone and dance (last in each row, enclosed in a red box). Expletives are censored. More examples are provided in the supplementary material.

as explained in Sec. 4.2.1. Common sources of false positives are actions performed off screen, or actions performed at a temporal offset (either much before or much after) the speech segment. We note that at no point do we ever actually use any of the manual labels for training, these are purely for evaluation and as a sanity check.

## 5. Action Classification

Now that we have described our method to obtain weakly labelled training data, we train a video classifier with the S3D-G [47] backbone on these noisy samples for the task of action recognition. We first detail the training and testing protocols, and then describe the datasets used in this work.

### 5.1. Evaluation Protocol

We evaluate our video classifier for the task of action classification in the following two ways:
First, we follow the typical procedure adopted in the video understanding literature [4]: pre-training on a large corpus of videos weakly labelled using our `Speech2Action` model, followed by fine-tuning on the training split of a labeled target dataset ('test bed'). After training, we evaluate the performance on the test set of the target dataset. In this work we use HMDB-51 [21], and compare to other state of the art methods on this dataset. We also provide results for the UCF101 dataset [36] in Sec. **??** of the suppl. material.
Second, and perhaps more interestingly, we apply our method by training a video classifier on the mined video clips for some action classes, and evaluating it *directly* on the test samples of *rare* action classes in the target dataset (in this case we use the AVA dataset [14]). Note: At this point we also manually verified that there is no overlap between the movies in the IMSDb dataset and the AVA dataset (not surprising since AVA movies are older and more obscure – these are movies that are freely available on YouTube). Here not a single manually labelled training example is used, since there is no finetuning (we henceforth refer to this as zero-shot[5]). We also report performance for the few-shot learning scenario, where we fine-tune our model on a *small* number of labelled examples. We note that in this case, we can only evaluate on the classes that directly overlap with the verb classes in the IMSDb dataset.

[5]In order to avoid confusion with the strict meaning of this term, we clarify that in this work we use it to refer to the case where not a single *manually labelled* example is available for a particular class. We do however train on multiple weakly labelled examples.

### 5.2. Datasets and Experimental Details

**HMDB51:** HMDB51 [21] contains 6,766 realistic and varied video clips from 51 action classes. Evaluation is performed using average classification accuracy over three train/test splits from [16], each with 3,570 train and 1,530 test videos.

**AVA:** The AVA dataset [14] is collected by exhaustively manually annotating videos and exhibits a strong imbalance in the number of examples between the common and rare classes. Eg. a common action, like 'stand', has 160K training and 43K test examples, compared to 'drive' (1.18K train and 561 test) and 'point' (only 96 train and 32 test). As a result, methods relying on full supervision struggle on the categories in the middle and the end of the tail. We evaluate on the 14 AVA classes that overlap with the classes present in the IMDSDb dataset (all from the middle and tail). While the dataset is originally a detection dataset, we repurpose it simply for the task of action classification, by assigning each frame the union of labels from all bounding box annotations. We then train and test on samples from these 14 action classes, reporting per-class average precision (AP).

**Implementation Details:** We train the S3D with gating (S3D-G) [47] model as our visual classifier. Following [47], we densely sample 64 frames from a video, resize input frames to $256 \times 256$ and then take random crops of size $224 \times 224$ during training. During evaluation, we use all frames and take $224 \times 224$ center crops from the resized frames. Our models are implemented with TensorFlow and optimized with a vanilla synchronous SGD algorithm with momentum of 0.9. For models trained from scratch, we train for 150K iterations with a learning rate schedule of $10^2$, $10^3$ and $10^4$ dropping after 80K and 100K iterations, and for finetuning we train for 60K iterations using a learning rate of $10^2$.

**Loss functions for training:** We try both the softmax cross-entropy and per-class sigmoid loss, and find that the performance was relatively stable with both choices.

### 5.3. Results

**HMDB51:** The results on HMDB51 can be seen in Table 3. Training on videos labelled with `Speech2Actions` leads to a significant 17% improvement over from-scratch training. For reference, we also compare to other self-supervised and weakly supervised works (note that these methods differ both in architecture and training objective). We show a 14% improvement over previous self-supervised works that use *only* video frames (no other modalities). We also compare to Korbar *et al.* [20] who pretrain using audio and video synchronisation on AudioSet, DisInit [13], which distills knowledge from ImageNet into Kinetics videos, and simply pretraining on ImageNet and then inflating 2D convolutions to our S3D-G model [19]. We improve over these works by 3-4% – which is impressive given that the latter two methods rely on access to a large-scale manually labelled image dataset [6], whereas ours relies only on 1000 unlabelled movie scripts. Another point of interest (and perhaps an unavoidable side-effect of this stream of self- and weak-supervision) is that while all these previous methods do not use labels, they still pretrain on the Kinetics data, which has been carefully curated to cover a wide diversity of over 600 different actions. In contrast, we mine our training data directly from movies, without the need for any manual labelling or careful curation, and our pretraining data was mined for only 18 classes.

| Method | Architecture | Pre-training | Acc. |
|---|---|---|---|
| Shuffle&Learn [28]⋆ | S3D-G (RGB) | UCF101† [36] | 35.8 |
| OPN [23] | VGG-M-2048 | UCF101† [36] | 23.8 |
| ClipOrder [48] | R(2+1)D | UCF101† [36] | 30.9 |
| Wang et al. [42] | C3D | Kinetics† [36] | 33.4 |
| 3DRotNet [18]⋆ | S3D-G (RGB) | Kinetics† | 40.0 |
| DPC [15] | 3DResNet18 | Kinetics† | 35.7 |
| CBT [37] | S3D-G (RGB) | Kinetics† | 44.6 |
| DisInit (RGB) [13] | R(2+1)D-18 [41] | Kinetics** | 54.8 |
| Korbar et al [20] | I3D (RGB) | Kinetics† | 53.0 |
| - | S3D-G (RGB) | Scratch | 41.2 |
| Ours | S3D-G (RGB) | `KSB-mined` | 46.0 |
| Ours | S3D-G (RGB) | `S2A-mined` | **58.1** |
| Supervised pretraining | S3D-G (RGB) | ImageNet | 54.7 |
| Supervised pretraining | S3D-G (RGB) | Kinetics | 72.3 |

Table 3. **Action classification results on HMDB51.** Pre-training on videos labelled with `Speech2Action` leads to a 17% improvement over training from scratch and also outperforms previous self-supervised and weakly supervised works. **KSB-mined:** video clips mined using the keyword spotting baseline. **S2A-mined:** video clips mined using the `Speech2Action` model. †videos without labels. **videos with labels distilled from ImageNet. When comparing to [20], we report the number achieved by their I3D (RGB only) model which is the closest to our architecture. For ⋆, we report the reimplementations by [37] using the S3D-G model (same as ours). For the rest, we report performance directly from the original papers.

**AVA-scratch:** The results on AVA for models trained from scratch with *no* pretraining, can be seen in Table 4 (top 4 rows). We compare the following: training with the AVA training examples (Table 4, top row), training only with our mined examples, and training jointly with both. For 8 out of 14 classes, we exceed fully supervised performance without a single AVA training example, in some cases ('drive' and 'phone') almost by 20%.

**AVA-finetuned:** We also show results for pre-training on `Speech2Action` mined clips first, and then fine-tuning on a gradually increasing number of AVA labelled training samples per class (Table 4, bottom 4 rows). Here we keep all the weights from the fine-tuning, including the classification layer weights, for initialisation, and fine-tune only for a single epoch. With 50 training samples per class, we exceed fully supervised performance for all classes (except for

| Data | Per-Class AP | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | drive | phone | kiss | dance | eat | drink | run | point | open | hit | shoot | push | hug | enter |
| AVA (fully supervised) | 0.63 | 0.54 | 0.22 | 0.46 | 0.67 | 0.27 | 0.66 | 0.02 | 0.49 | 0.62 | 0.08 | 0.09 | 0.29 | 0.14 |
| KS-baseline † | 0.67 | 0.20 | 0.12 | 0.53 | 0.67 | 0.18 | 0.37 | 0.00 | 0.33 | 0.47 | 0.05 | 0.03 | 0.10 | 0.02 |
| S2A-mined (zero-shot) | 0.83 | 0.79 | 0.13 | 0.55 | 0.68 | 0.30 | 0.63 | 0.04 | 0.52 | 0.54 | 0.18 | 0.04 | 0.07 | 0.04 |
| S2A-mined + AVA | 0.84 | 0.83 | 0.18 | 0.56 | 0.75 | 0.40 | 0.74 | 0.05 | 0.56 | 0.64 | 0.23 | 0.07 | 0.17 | 0.04 |
| AVA (few-shot)-20 | 0.82 | 0.83 | 0.22 | 0.55 | 0.69 | 0.33 | 0.64 | 0.04 | 0.51 | 0.59 | 0.20 | 0.06 | 0.19 | 0.13 |
| AVA (few-shot)-50 | 0.82 | 0.85 | 0.26 | 0.56 | 0.70 | 0.37 | 0.69 | 0.04 | 0.52 | 0.65 | 0.21 | 0.06 | 0.19 | 0.15 |
| AVA (few-shot)-100 | 0.84 | 0.86 | 0.30 | **0.58** | 0.71 | 0.39 | **0.75** | **0.05** | 0.58 | 0.73 | 0.25 | **0.13** | 0.27 | 0.15 |
| AVA (all) | **0.86** | **0.89** | **0.34** | **0.58** | **0.78** | **0.42** | **0.75** | 0.03 | **0.65** | **0.72** | **0.26** | **0.13** | **0.36** | **0.16** |

Table 4. **Per-class average precision for 14 AVA mid and tail classes.** These actions occur *rarely*, and hence are harder to get manual supervision for. For 8 of the 14 classes, we exceed fully supervised performance without a single manually labelled training example (highlighted in pink, best viewed in colour). `S2A-mined`: Video clips mined using `Speech2Action`. † Keyword spotting baseline. First 4 rows: models are trained from scratch. Last 4 rows: we pre-train on video clips mined using `Speech2Action`.

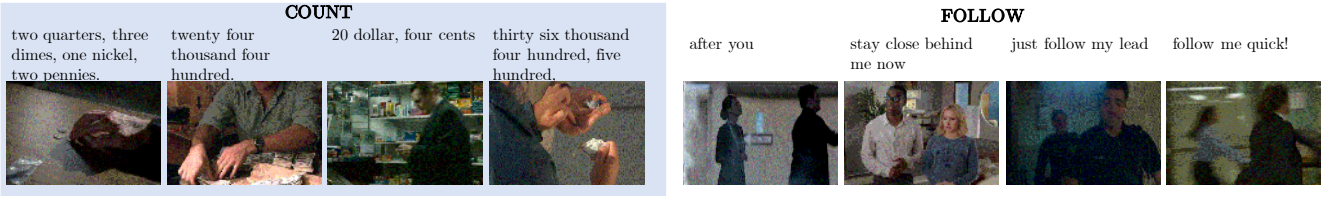


Figure 5. **Examples of clips mined for more abstract actions.** These are actions that are not present in standard datasets like HMDB51 or AVA, but are quite well correlated with speech. Our method is able to automatically mine clips weakly labelled with these actions from unlabelled data.

'hug' and 'push') compared to training from scratch. The worst performance is for the class 'hug' – 'hug' and 'kiss' are often confused, as the speech in both cases tends to be similar – *'I love you'*. A quick manual inspection shows that most of the clips are wrongly labelled as 'kiss', which is why we are only able to mine very few video clips for this class. For completeness, we also pretrain a model with the S2A mined clips (only 14 classes) and then finetune on AVA for *all 60* classes used for evaluation, and get a 40% overall classification acc. vs 38% with training on AVA alone.

**Mining Technique:** We also train on clips mined using the keyword spotting baseline (Table 4). For some classes, this baseline itself exceeds fully supervised performance. Our `Speech2Action` labelling beats this baseline for all classes, indeed the baseline does poorly for classes like 'point' and 'open' – verbs which have many semantic meanings, demonstrating that the semantic information learnt from the IMSDb dataset is valuable. However we note here that it is difficult to measure performance quantitatively for the class 'point' due to idiosyncrasies in the AVA test set (wrong ground truth labels for very few test samples) and hence we show qualitative examples of mined clips in Fig. 4. We note that the baseline comes very close for 'dance' and 'eat', demonstrating that simple keyword matching on speech can retrieve good training data for these actions.

**Abstract Actions:** By gathering data directly from the stage directions in movie screenplays, our action labels are post-defined (as in [11]). This is unlike the majority of the existing human action datasets that use pre-defined labels [3, 14, 29, 34]. Hence we also manage to mine examples for some unusual or *abstract* actions which are quite well correlated with speech, such as 'count' and 'follow'. While these are not present in standard action recognition datasets such as HMDB51 or AVA, and hence cannot be evaluated numerically, we show some qualitative examples of these mined videos in Fig. 5.

## 6. Conclusion

We provide a new data-driven approach to obtain weak labels for action recognition, using speech alone. With only a thousand unaligned screenplays as a starting point, we obtain weak labels automatically for a number of rare action classes. However, there is a plethora of literary material available online, including plays and books, and exploiting these sources of text may allow us to extend our method to predict other action classes, including composite actions of 'verb' and 'object'. We also note that *besides* actions, people talk about physical objects, events and scenes – descriptions of which are also present in screenplays and books. Hence the same principle used here could be applied to mine videos for more general visual content.

**Acknowledgments:** Arsha is supported by a Google PhD Fellowship. We are grateful to Carl Vondrick for early discussions.

## References


[1] Relja Arandjelovic and Andrew Zisserman. Look, listen and learn. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 609–617, 2017. 2

[2] Piotr Bojanowski, Francis Bach, Ivan Laptev, Jean Ponce, Cordelia Schmid, and Josef Sivic. Finding actors and actions in movies. In *Proceedings of the IEEE international conference on computer vision*, pages 2280–2287, 2013. 2

[3] Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 961–970, 2015. 8

[4] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the Kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308, 2017. 2, 6

[5] Timothee Cour, Benjamin Sapp, Chris Jordan, and Ben Taskar. Learning from ambiguously labeled images. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 919–926. IEEE, 2009. 2

[6] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 7

[7] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 3

[8] Olivier Duchenne, Ivan Laptev, Josef Sivic, Francis Bach, and Jean Ponce. Automatic annotation of human actions in video. In *2009 IEEE 12th International Conference on Computer Vision*, pages 1491–1498. IEEE, 2009. 2

[9] Mark Everingham, Josef Sivic, and Andrew Zisserman. "Hello! My name is... Buffy" – automatic naming of characters in TV video. In *BMVC*, 2006. 2

[10] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 6202–6211, 2019. 2

[11] David F Fouhey, Wei-cheng Kuo, Alexei A Efros, and Jitendra Malik. From lifestyle vlogs to everyday interactions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4991–5000, 2018. 1, 2, 8

[12] Deepti Ghadiyaram, Du Tran, and Dhruv Mahajan. Large-scale weakly-supervised pre-training for video action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 12046–12055, 2019. 2

[13] Rohit Girdhar, Du Tran, Lorenzo Torresani, and Deva Ramanan. Distinit: Learning video representations without a single labeled video. *ICCV*, 2019. 7

[14] Chunhui Gu, Chen Sun, David A Ross, Carl Vondrick, Caroline Pantofaru, Yeqing Li, Sudheendra Vijayanarasimhan, George Toderici, Susanna Ricco, Rahul Sukthankar, Cordelia Schmid, and Jitendra Malik. AVA: A video dataset of spatio-temporally localized atomic visual actions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6047–6056, 2018. 1, 2, 5, 6, 7, 8

[15] Tengda Han, Weidi Xie, and Andrew Zisserman. Video representation learning by dense predictive coding. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2019. 7

[16] Haroon Idrees, Amir R Zamir, Yu-Gang Jiang, Alex Gorban, Ivan Laptev, Rahul Sukthankar, and Mubarak Shah. The THUMOS challenge on action recognition for videos "in the wild". *Computer Vision and Image Understanding*, 155:1–23, 2017. 7

[17] Oana Ignat, Laura Burdick, Jia Deng, and Rada Mihalcea. Identifying visible actions in lifestyle vlogs. *arXiv preprint arXiv:1906.04236*, 2019. 1, 2

[18] Longlong Jing and Yingli Tian. Self-supervised spatiotemporal feature learning by video geometric transformations. *arXiv preprint arXiv:1811.11387*, 2018. 7

[19] W. Kay, J. Carreira, K. Simonyan, B. Zhang, C. Hillier, S. Vijayanarasimhan, F. Viola, T. Green, T. Back, P. Natsev, M. Suleyman, and A. Zisserman. The Kinetics human action video dataset. *CoRR*, abs/1705.06950, 2017. 1, 2, 5, 7

[20] Bruno Korbar, Du Tran, and Lorenzo Torresani. Cooperative learning of audio and video models from self-supervised synchronization. In *Advances in Neural Information Processing Systems*, pages 7763–7774, 2018. 2, 7

[21] Hildegard Kuehne, Hueihan Jhuang, Estíbaliz Garrote, Tomaso Poggio, and Thomas Serre. Hmdb: a large video database for human motion recognition. In *2011 International Conference on Computer Vision*, pages 2556–2563. IEEE, 2011. 6, 7

[22] Ivan Laptev, Marcin Marszałek, Cordelia Schmid, and Benjamin Rozenfeld. Learning realistic human actions from movies. In *IEEE Conference on Computer Vision & Pattern Recognition*, 2008. 2

[23] Hsin-Ying Lee, Jia-Bin Huang, Maneesh Singh, and Ming-Hsuan Yang. Unsupervised representation learning by sorting sequences. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 667–676, 2017. 7

[24] Edward Loper and Steven Bird. Nltk: the natural language toolkit. *arXiv preprint cs/0205028*, 2002. 3

[25] Marcin Marszałek, Ivan Laptev, and Cordelia Schmid. Actions in context. In *CVPR 2009-IEEE Conference on Computer Vision & Pattern Recognition*, pages 2929–2936. IEEE Computer Society, 2009. 2

[26] Antoine Miech, Jean-Baptiste Alayrac, Piotr Bojanowski, Ivan Laptev, and Josef Sivic. Learning from video and text via large-scale discriminative clustering. In *Proceedings of the IEEE international conference on computer vision*, 2017. 2

[27] Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. HowTo100M: Learning a Text-Video Embedding by Watching Hundred Million Narrated Video Clips. In *Proceedings of the IEEE international conference on computer vision*, 2019. 1, 2

[28] Ishan Misra, C Lawrence Zitnick, and Martial Hebert. Shuffle and learn: unsupervised learning using temporal order verification. In *European Conference on Computer Vision*, pages 527–544. Springer, 2016. 7

[29] Mathew Monfort, Alex Andonian, Bolei Zhou, Kandan Ramakrishnan, Sarah Adel Bargal, Yan Yan, Lisa Brown, Quanfu Fan, Dan Gutfreund, Carl Vondrick, et al. Moments in time dataset: one million videos for event understanding. *IEEE transactions on pattern analysis and machine intelligence*, 2019. 1, 8

[30] Iftekhar Naim, Abdullah Al Mamun, Young Chol Song, Jiebo Luo, Henry Kautz, and Daniel Gildea. Aligning movies with scripts by exploiting temporal ordering constraints. In *2016 23rd International Conference on Pattern Recognition (ICPR)*, pages 1786–1791. IEEE, 2016. 2

[31] Andrew Owens and Alexei A Efros. Audio-visual scene analysis with self-supervised multisensory features. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 631–648, 2018. 2

[32] Andrew Owens, Jiajun Wu, Josh H McDermott, William T Freeman, and Antonio Torralba. Ambient sound provides supervision for visual learning. In *European conference on computer vision*, pages 801–816. Springer, 2016. 2

[33] Christopher Riley. *The Hollywood standard: the complete and authoritative guide to script format and style*. Michael Wiese Productions, 2009. 3

[34] Gunnar A Sigurdsson, Gül Varol, Xiaolong Wang, Ali Farhadi, Ivan Laptev, and Abhinav Gupta. Hollywood in homes: Crowdsourcing data collection for activity understanding. In *European Conference on Computer Vision*, pages 510–526. Springer, 2016. 8

[35] Josef Sivic, Mark Everingham, and Andrew Zisserman. "who are you?"-learning person specific classifiers from video. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1145–1152. IEEE, 2009. 2

[36] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012. 6, 7

[37] Chen Sun, Fabien Baradel, Kevin Murphy, and Cordelia Schmid. Contrastive bidirectional transformer for temporal representation learning. *arXiv preprint arXiv:1906.05743*, 2019. 7

[38] Yansong Tang, Dajun Ding, Yongming Rao, Yu Zheng, Danyang Zhang, Lili Zhao, Jiwen Lu, and Jie Zhou. Coin: A large-scale dataset for comprehensive instructional video analysis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1207–1216, 2019. 1, 2

[39] Makarand Tapaswi, Martin Bäuml, and Rainer Stiefelhagen. "knock! knock! who is it?" probabilistic person identification in tv-series. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2658–2665. IEEE, 2012. 2

[40] Makarand Tapaswi, Martin Bauml, and Rainer Stiefelhagen. Book2movie: Aligning video scenes with book chapters. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015. 2

[41] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 6450–6459, 2018. 2, 7

[42] Jiangliu Wang, Jianbo Jiao, Linchao Bao, Shengfeng He, Yunhui Liu, and Wei Liu. Self-supervised spatio-temporal representation learning for videos by predicting motion and appearance statistics. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4006–4015, 2019. 7

[43] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks: Towards good practices for deep action recognition. In *ECCV*, 2016. 2

[44] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *CVPR*, 2018. 2

[45] David R Winer and R Michael Young. Automated screenplay annotation for extracting storytelling knowledge. In *Thirteenth Artificial Intelligence and Interactive Digital Entertainment Conference*, 2017. 3

[46] Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*, 2016. 3

[47] Saining Xie, Chen Sun, Jonathan Huang, Zhuowen Tu, and Kevin Murphy. Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 305–321, 2018. 2, 6, 7

[48] Dejing Xu, Jun Xiao, Zhou Zhao, Jian Shao, Di Xie, and Yueting Zhuang. Self-supervised spatiotemporal learning via video clip order prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10334–10343, 2019. 7

[49] Hang Zhao, Chuang Gan, Andrew Rouditchenko, Carl Vondrick, Josh McDermott, and Antonio Torralba. The sound of pixels. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 570–586, 2018. 2

[50] Hang Zhao, Zhicheng Yan, Heng Wang, Lorenzo Torresani, and Antonio Torralba. SLAC: A sparsely labeled dataset for action classification and localization. *arXiv preprint arXiv:1712.09374*, 2017. 2

[51] Luowei Zhou, Chenliang Xu, and Jason J Corso. Towards automatic learning of procedures from web instructional videos. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018. 1, 2

[52] Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *Proceedings of the IEEE international conference on computer vision*, pages 19–27, 2015. 2, 3