Unsupervised Alignment of Actions in Video with Text Descriptions

Young Chol Song,[1] **Iftekhar Naim,**[*1] **Abdullah Al Mamun,**[1] **Kaustubh Kulkarni,**[†2]
Parag Singla,[2] **Jiebo Luo,**[1] **Daniel Gildea,**[1] **Henry Kautz**[1]
[1]Department of Computer Science, University of Rochester, Rochester, NY, USA
[2]Indian Institute of Technology Delhi, New Delhi, India

Abstract

Advances in video technology and data storage have made large scale video data collections of complex activities readily accessible. An increasingly popular approach for automatically inferring the details of a video is to associate the spatiotemporal segments in a video with its natural language descriptions. Most algorithms for connecting natural language with video rely on pre-aligned supervised training data. Recently, several models have been shown to be effective for unsupervised alignment of objects in video with language. However, it remains difficult to generate good spatiotemporal video segments for **actions** that align well with language. This paper presents a framework that extracts higher level representations of low-level action features through hyperfeature coding from video and aligns them with language. We propose a two-step process that creates a high-level action feature codebook with temporally consistent motions, and then applies an unsupervised alignment algorithm over the action codewords and verbs in the language to identify individual activities. We show an improvement over previous alignment models of objects and nouns on videos of biological experiments, and also evaluate our system on a larger scale collection of videos involving kitchen activities.

1 Introduction

With advances in video technology, we have seen an increase in the availability of large scale video datasets. However, as video data becomes abundant, the work required in generating accurately segmented, aligned, and labeled data for these sets also increases in difficulty. Unlike the smaller action datasets used in the past, with carefully crafted domains and annotations that have been defined to the level of individual frames, longer term activity datasets are becoming more common with annotations in the form of natural language descriptions and only loosely associated with language.

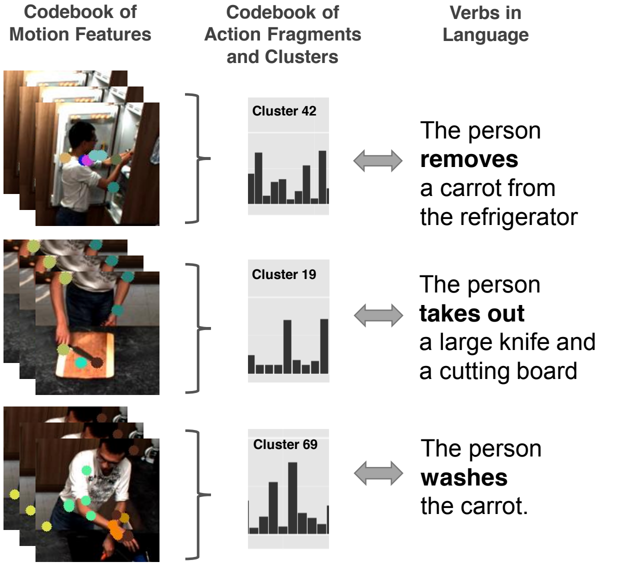
Codebook of Motion Features
Codebook of Action Fragments and Clusters
Verbs in Language
Cluster 42
The person **removes** a carrot from the refrigerator
Cluster 19
The person **takes out** a large knife and a cutting board
Cluster 69
The person **washes** the carrot.

Figure 1: An overview of our text and video alignment framework. Codebooks of motion features (left) are accumulated over multiple frames and encoded into action fragments and clusters (center), and associated with verbs in text (right).

While manually labeling datasets requires a great amount of effort on the part of the annotators, annotations in the form of natural language descriptions can often be retrieved as a byproduct of data collection without human intervention. For example, during a video of a person performing a multiple step kitchen activity (*e.g.*, baking a cake), it is relatively easy for the person to add their own descriptions through speech with the intent of teaching someone how to conduct the activity. Also, if the person is following instructions according to a recipe, the text in the recipe is a good resource for describing the video. It would be beneficial to use indirect descriptions that are readily available, rather than having to manually go through and tag each video segment.

For actions in particular, generating labels directly from text descriptions becomes more crucial. Conventionally, the act of manual annotation assumes the domain of labels is either known or defined in advance. While this fact may hold for datasets collected in a controlled setting, defining the do-

[*] Iftekhar is now at Google
[†] Kaustubh is now at Yahoo Japan

main of actions from data collected in a live and complex scene is much more ambiguous. Unlike objects, which are generally well defined, identical motions may belong to different actions in a hierarchy, and the boundaries of actions may overlap or not be strictly defined. In this case, the associated verbs in language can point us to what actions are relevant for that particular activity. By creating labels for actions directly from verbs in the language, we naturally focus only on the events stemming from text, and assume the labels of the most relevant actions come from the descriptions in the language themselves.

In this paper, we consider the problem of recognizing actions from text descriptions and multiple instances of video. In particular, we focus on the problem of mapping sentences in language to their corresponding segments in video, and also mapping verbs to their corresponding action representations. We aim to expand on the work on unsupervised alignment by Naim et al. [2014; 2015] which introduces a framework to ground words in language with objects in the video. While their work exploited co-occurrences of video objects with nouns and verbs in the text, they did not use information on activities in the video. We argue that while objects have a fixed spatial boundary and directly connect with constituents in language, actions are generally more free-form both spatially and temporally with ambiguous boundaries, making it harder to form a common visual vocabulary and requiring a more robust form of representation.

Unlike previous papers which assume a set of labeled actions [Bojanowski *et al.*, 2014], classify a set of actions from training data [Regneri *et al.*, 2013], or try to directly associate low-level motion features with language [Bojanowski *et al.*, 2015], we introduce and evaluate a process that groups temporally consistent motion features together using hyperfeature coding that provides a natural mapping from action clusters to verbs without any supervision. By evaluating the actions in two different datasets of varying domains, we show that it is possible to describe actions in the video in detail, without the need to explicitly label each action individually.

2 Related Work

Our system integrates work from a variety of fields including action recognition, unsupervised clustering, and work on aligning video and language. We give an overview of literature in each of these related areas relevant to our framework.

2.1 Features for Action Recognition

Space-time motion descriptors based on local video features have been widely accepted in human action recognition. Many of these low-level features usually consist of two parts: a detector that locates points of interest, and a descriptor that captures space-time information on these points, generally independent of shift in space-time and background noise. Harris3D, Cuboid, and Hessian detectors are commonly used [Wang *et al.*, 2009], while descriptors are extensions of commonly used image descriptors, such as HOG/HOF [Laptev, 2005] and MBH [Wang *et al.*, 2011]. These features have been well suited for recognizing coarse actions through classification.

A commonly used descriptor which is also used in our evaluations are the space-time interest points (STIPs) by Laptev [2005], which uses a Harris3D detector to localize points of interest, and extracts histograms of gradient (HOG) and histograms of optical flow (HOF) of varying temporal and spatial scales from video segments surrounding these points. The alternative is a dense features approach which samples from regularly spaced points in space and time instead of looking at particular points of interest. In particular, methods based on dense [Wang *et al.*, 2011] and improved [Wang and Schmid, 2013] trajectories extract motion descriptors from densely sampled points in the video and track them, and show improvement over previous descriptors.

There has also been research on using deep architectures for action recognition. In particular, convolutional neural networks (CNNs) [LeCun *et al.*, 2001] have been especially successful in large scale classification of images [Krizhevsky *et al.*, 2012], and have recently been applied to actions. Yang et al. [2015] trained neural network classifiers on a limited set of grasping actions to learn hand manipulation actions on videos collected from the web. Motion features derived from CNN models [Gkioxari and Malik, 2015; Wang *et al.*, 2015] show improvements in the case of action classification, showing that the quality of features generated from CNNs can exceed those of hand-crafted features. We evaluate our system on action features generated by CNN models trained using the UCF101 action recognition dataset [Soomro *et al.*, 2012].

2.2 Clustering and Segmentation of Actions

There is a large pool of literature on unsupervised segmentation of activities. Various techniques, such as change-point detection [Harchaoui *et al.*, 2009] and PCA [Barbič *et al.*, 2004] are used to detect action boundaries based on changes in distributions over time. Other algorithms use repetitive motions as a basis for clustering and segmentation. Aligned cluster analysis (ACA) and its hierarchical variant [Zhou *et al.*, 2013] focus on the problem of conducting joint temporal clustering and segmentation through a generalization of kernel k-means and spectral clustering.

However, unlike the action clustering algorithms mentioned above, we do not have to solve the problem of clustering and segmentation simultaneously, as segmentation is conducted through the alignment of text. Instead, we consider the creation of a flexible high level representation of action features that can be used for alignment and can be learned without supervision. We borrow the notion of aggregating visual features across local patches in images [Coates and Ng, 2012], and choose to perform a hierarchical codebook generation [Agarwal and Triggs, 2006] of motion features at multiple temporal scales.

2.3 Alignment of Video and Language

Given features that roughly correspond to actions in language, we would like to learn the ideal mapping of natural language action expressions with their referents in the video. This is known as grounded language learning [Yu and Ballard, 2004]. Most grounded language learning algorithms are supervised or semi-supervised [Kollar *et al.*, 2010;

Yu and Ballard, 2004; Kate and Mooney, 2007; Tellex *et al.*, 2014], which assume that each sentence in the text description is manually aligned to its corresponding image frame or video clip. Manually acquiring such sentence-level segmentation and alignment can be tedious for large collection of parallel video and text datasets, and hence unsupervised alignment is crucial for scaling to large datasets.

Naim et al. [2014; 2015] introduced several different generative and discriminative models for aligning text with video without the need for individual labeling or segmentation. While they exploited the co-occurrences of the tracked objects in video with the nouns and verbs in the text, they did not consider actions. We extend their alignment model by incorporating the correlations between actions and verbs, which can be applied even if object tracking does not work well. Malmaud et al. [2015] aligned YouTube cooking videos with recipes by matching the words in the recipes with the video-speech transcripts, which may not be available for many domains. Recently, Bojanowski et al. [2015] proposed a discriminative clustering approach using integer quadratic programming to align low level action features with text, whereas our model jointly aligns both objects and actions in video with their corresponding nouns and verbs in the text.

3 Joint Alignment and Matching for Actions

The input to our system is a corpus of videos and associated text descriptions where individuals are performing complex activities. The text can either be a list of instructions that are followed by the user, or a set of descriptions that are annotated after the activity is conducted. We make the assumption that there is a correlation between the temporal progression of video and the order in which the text is written. Our goal is to align the video to sentences in the corresponding text, while simultaneously learning the mapping of each action and object with the corresponding verbs and nouns.

3.1 Multilevel Features for Action Representation

The matching of video segments with text requires that we extract action feature representations from the video that are temporally comparable to those in text. However, commonly used low-level motion features do not provide a good representation of the actions in the larger temporal and spatial contexts that are present in text descriptions.

We adopt the concept of hyperfeatures from Agrawal and Triggs [2006] and apply it to action representations. Motion features are accumulated using a sliding temporal window and quantized over a codebook of action fragments, defined as clusters of commonly occurring motion histogram vectors. These fragments are subsequently aggregated to detect co-occurrences, resulting in higher-level feature representations of actions.

Hyperfeature Codebooks for Motion Features

Many popular image recognition algorithms base their methods on the aggregation of visual features across local image patches [Lowe, 2004; Coates and Ng, 2012]. Similarities can also be found in convolutional neural networks, where local filters are learned codebooks, and pooling is spatial or temporal aggregation. However, unlike neural networks, hyperfeatures are created purely from the bottom up without any supervision; higher-level representations are derived from existing lower layers representing smaller image patches. We draw a similar analogy for representing actions in action classification. Here, the aggregation of motion features can occur over temporal intervals of various lengths. In supervised classification, this interval is given as training data along with the action labels. In our case, we evaluate our system using temporal intervals of various lengths to quantize the action occurring for the interval in question.

A common way of aggregating motion features is by using a bag-of-features approach [Wang *et al.*, 2009; 2011]. Since a motion feature may be of a variable length, we create a codebook of commonly occurring motion features over the entire dataset and conduct vector quantization. From this, we create an action fragment: a normalized histogram of quantized features over a designated window, characterizing the distribution of features over that interval. We use k-means for codebook creation because of its simplicity and scalability to large datasets [Coates and Ng, 2012]. However, our approach is not limited to any particular clustering algorithm. Different values for codebook and window sizes are evaluated in the experiments section.

To capture co-occurrences of action fragments in our data, we conduct vector quantization over all action fragments in the dataset, similar to what we have done for motion features. The code vector results in a higher level representation for actions, which will be used as input to the alignment algorithm. For even higher-level actions, it is also possible to create a hyperfeature stack; repeating the process by using the aggregated fragments as motion features for the next level of the stack.

Algorithm 1 Hyperfeature coding for motion features

```
$\forall(v,t,s), F^{(0)}_{v,t,s} \leftarrow s^{th}$ feature in video $v$ at frame $t$
for $l = 1 \dots L$ do
    cluster $\{F^{(l)}_{v,t,s} \mid \forall(v,t,s)\}$ using k-means with
        $d^{(l)}$ centroids such that a code vector $c^{(l)}_{v,t,i}$
        is generated for each $F^{(l)}_{v,t,s}$
    if $l < L$ then
        $\forall(v,t,s), F^{(l+1)}_{v,t,s} \leftarrow$ accumulate features
                in the neighborhood of window size $w$
                as a histogram of $d^{(l)}$ vectors

        normalize $F^{(l+1)}_{v,t,s}$
    end if
end for
return code vectors $c^{(l)}_{v,t,s}, \forall(v,t,s)$
```

Algorithm 1 describes this process in detail. The total number of levels in the hyperfeature stack is defined by L, where l is the current level. The 0^{th} level feature vector of length s is defined as $F^{(0)}_{v,t,s}$ for each video v and frame t. Once cluster centroids $d^{(l)}$ are created from features, code vectors $c^{(l)}_{v,t,i}$ are generated based on the centroids with i being the i^{th} element of the code vector. Next level features $F_{v,t,s}^{(l+1)}$ are defined by accumulating code vectors of centroids from the previous level over a window defined by w, and are normalized. At the last level L, code vectors $c_{v,t,s}^{(L)}$ for each video v and frame t are returned and provided as input to the alignment process.

3.2 Aligning Language with Video Activities

We detail the process of taking verbs from language and aligning them with hyperfeatures of actions learned from Algorithm 1. We use the Latent-variable Conditional Random Field (LCRF) alignment model by Naim et al. [2015] originally applied to blob and noun alignment, and modify the process to capture the correlations between hyperfeatures generated from the videos and verbs in the text sentences.

Let the input dataset consist of N pairs of observations $\{(\mathbf{x}_i, \mathbf{y_i})\}_{i=1}^{N}$, where $\mathbf{x}_i$ represents the i^{th} text description and $\mathbf{y}_i$ represents the corresponding i^{th} video. Each text description $\mathbf{x}_i$ is a sequence of sentences: $\mathbf{x}_i = [X_{i,1}, \ldots, X_{i,m_i}]$, where m_i is the number of sentences in $\mathbf{x}_i$ and $X_{i,m}$ represents the set of head nouns and verbs extracted from the m^{th} sentence in $\mathbf{x}_i$. The head-nouns and verbs are extracted by parsing each sentence in $\mathbf{x}_i$ using the two-stage Charniak-Johnson parser. We lemmatize each verb to normalize the verbs across different tenses. Each video $\mathbf{y}_i$ is a sequence of short fixed-duration disjoint chunks: $\mathbf{y}_i = [Y_{i,1}, \ldots, Y_{i,n_i}]$, where n_i is the number of chunks in $\mathbf{y}_i$ and $Y_{i,n}$ represents the blobs and actions detected from that chunk. The details of detecting blobs and actions from video are described in the next section. Our goal is to learn the latent alignment $\mathbf{h}_i$ between the sentences in $\mathbf{x}_i$ with their corresponding video chunks in $\mathbf{y}_i$. The latent alignment variable is $\mathbf{h}_i[n] \in \{1, \ldots, m_i\}$, for $1 \leq n \leq n_i$, where $\mathbf{h}_i[n] = m$ indicates that the video segment $Y_{i,n}$ is aligned to the text sentence $X_{i,m}$.

Given a text description $\mathbf{x}_i$ and a video sequence $\mathbf{y}_i$ with lengths $|\mathbf{x}_i| = m_i$ and $|\mathbf{y}_i| = n_i$, the conditional likelihood of the video sequence is defined as:

$$p(\mathbf{y}_i|\mathbf{x}_i, n_i) = \sum_{\mathbf{h}_i} p(\mathbf{y}_i, \mathbf{h}_i|\mathbf{x}_i, n_i). \tag{1}$$

The conditional probability $p(\mathbf{y}_i, \mathbf{h}_i|\mathbf{x}_i, n_i)$ is modeled using a log-linear model:

$$p(\mathbf{y}_i, \mathbf{h}_i|\mathbf{x}_i, n_i) = \frac{\exp \mathbf{w}^T \mathbf{\Phi}(\mathbf{x}_i, \mathbf{y}_i, \mathbf{h}_i)}{Z(\mathbf{x}_i, n_i)}, \tag{2}$$

where $Z(\mathbf{x}_i, n_i) = \sum_{\mathbf{y}} \sum_{\mathbf{h}} \exp \mathbf{w}^T \mathbf{\Phi}(\mathbf{x}_i, \mathbf{y}, \mathbf{h})$. Let $\mathbf{\Phi}(\mathbf{x}_i, \mathbf{y}, \mathbf{h})$ be a feature function that decomposes linearly, similar to a linear-chain graphical model. The model parameter $\mathbf{w}$ represents the feature weights, which are trained by maximizing the following conditional log-likelihood function via stochastic gradient ascent:

$$L(\mathbf{w}) = \sum_{i=1}^{N} \log \sum_{\mathbf{h}_i} p(\mathbf{y}_i, \mathbf{h}_i|\mathbf{x}_i, n_i). \tag{3}$$

Average Alignment Accuracy (%)				
Hand and Object Tracking	LCRF Naim *et al.*	LCRF +STIP	LCRF +DTraj[2]	LCRF +CNN
Vision Tracks	65.59	66.55	**67.77**	66.91
Manual Tracks	85.09	87.10	86.92	**87.38**

Table 1: Average alignment accuracy (% of video chunks aligned to the correct protocol step) for the Wetlab dataset. *Vision Tracks* show results where object and hand tracking was handled by a hand and object recognizer based on color and depth. *Manual Tracks* show results where hand and objects were manually annotated. *LCRF* is the current state-of-the-art model used for the alignment of objects to language without incorporating actions.

We include the standard features used by Naim et al. [2015]: every pair of (*noun*, *blob*) and (*verb*, *blob*), jump size (0 or 1 for monotonic alignment), and diagonal path features to encourage alignment states to be close to diagonal. Furthermore, we capture the co-occurrences between actions and verbs by incorporating new features (a, v) for each action cluster a and verb v.

4 Experiments

This section describes the evaluation of hyperfeature construction and alignment of actions on two multimodal datasets with parallel video and text. The Wetlab dataset [Naim *et al.*, 2014; 2015] has RGB and depth video with text in the form of lab protocols. Participants conduct a variety of biology lab experiments following the steps specified in the protocol. The TACoS corpus [Regneri *et al.*, 2013] has RGB video with text in the form of multiple natural language descriptions collected via crowdsourcing on Amazon Mechanical Turk[1]. Subjects conduct common kitchen activities. Both datasets are focused on conducting a sequence of complex actions to complete a high level activity.

4.1 Features for Action Representation

For each dataset, we extract STIP and CNN motion features. STIPs consist of a 72-element HOG and 90-element HOF descriptor, and each frame can have multiple STIPs. For the CNN motion features, we used a CNN trained on the UCF101 dataset (split 1) [Soomro *et al.*, 2012], consisting of 5 convolutional and 3 fully connected layers. Each frame in the video is resized to a pixel size of 227x227 and optical flow is calculated for each sequential video frame pair. The output from optical flow is then given as input to the CNN, and the output of the first fully connected layer ($k = 4096$) is considered as a feature vector for that frame.

For the Wetlab dataset, we also extract dense trajectories, which consist of 30-element trajectory, 96-element HOG, 108-element HOF and 192-element MBH descriptors. Since the camera was stationary at all times during our experiments, we did not make use of MBH descriptors in our evaluations. For our evaluations, we follow the code of Algorithm 1 to extract hyperfeatures with varying centroids $d^{(l)}$ and window sizes w.

[1] https://www.mturk.com

[2] Dense trajectories

	Avg. Alignment Accuracy (%)
Uniform	34.87
Unsupervised LCRF +STIP	43.07
Unsupervsied LCRF +CNN	**44.14**
Segmented LCRF	**51.93**

Table 2: Average alignment accuracy (% of video chunks aligned to the correct protocol step) for the TACoS dataset. *Uniform* is a complete uniform assignment from text to video. *Unsupervised LCRF* conducts alignment between action clusters and verbs in the language, using *STIP* and *CNN* motion features. *Segmented LCRF* refers to the case when the segmentation is given along with action clusters for the segments, but the action itself is not identified. Unlike the Wetlab dataset, only actions were used for alignment.

Centroids	Window Size w					
	15	75	150	300	450	600
$d^{(1)}$=64	35.17	43.40	**44.14**	42.44	41.58	39.67
$d^{(1)}$=128	37.65	42.52	42.85	43.01	42.01	39.59

Table 3: Average alignment accuracy for different hyperfeature variables $(d^{(1)}, w, d^{(2)}$=64$)$ on the TACoS dataset for *Unsupervised LCRF+CNN.*

4.2 Wetlab Dataset

The Wetlab dataset consists of a set of RGB-Depth videos of different biological experiments being conducted in a wet laboratory setting. A total of 12 videos were collected for 3 protocols, with 4 videos per protocol. Each protocol has a set of natural language instructions which each lab member is expected to follow. The protocols have 9 steps and 24 sentences on average, with 34 unique nouns and 25 unique verbs.

Unsupervised Hand and Object Detection

Color models of gloves were created by manually labeling 20 randomly sampled frames in RGB and LAB color space from a separate wetlab dataset. The 3D coordinates of the hands are extracted by creating a point-cloud of the extracted hands, and calculating the center of mass at each frame. A Kalman filter was used to smooth out jitter and lost frames during tracking.

For objects, the scene is segmented using an adjacency matrix representing the set of connected components that correspond to blobs in the depth video. The connected components are over-segmented by color using a modified version of the SLIC superpixel algorithm [Achanta *et al.*, 2012], and superpixels are grouped using a greedy approach by their color and boundary map [Luo and Guo, 2003]. Hand and object interactions are inferred in 3D space, with interactions within a designated threshold corresponding with object usage. Alignment of nouns and objects are conducted using the LCRF alignment introduced in Naim et al. [2015].

Wetlab Evaluation

We measure the alignment accuracy by the percentage of video chunks that are aligned to the correct protocol step. In order to estimate the error due to alignment and tracking, we apply alignment on both automatically generated tracks and manually labeled tracks via the Anvil video annotation tool [Kipp, 2012].

We experimented with various numbers of clusters and window sizes. In Table 1, we report the average results for three different motion features: STIP, dense trajectories, and CNN features. For hyperfeature variables $\{d^{(1)}, w, d^{(2)}\}$, we achieved best results using $\{64, 150, 32\}$ for STIP, $\{128, 150, 32\}$ for dense trajectory, and $\{128, 150, 64\}$ for CNN features. For all the variations, we train LCRF models by running 200 iterations over the entire dataset. Each iteration per video took an average of 6.6 seconds on a single core of a 2.4GHz Intel Xeon processor with 32GB of RAM. We also compare our results with the state-of-the-art alignment results on the same dataset.

Our results show that using actions and verbs in addition to objects and nouns for alignment produces an improvement in overall average alignment accuracy regardless of the type of motion features used, with the largest improvements shown when using dense trajectory and CNN features. However, we believe the increase in improvement is limited because a majority of actions in the wetlab dataset are synonymous to object-use, therefore the results from Naim et al. [2015] are already highly correlated with a majority of action features. To further evaluate our framework, we use a larger dataset with a wider variety of fine-grained actions without object-use annotations.

4.3 TACoS Dataset

We extend our evaluation to a larger dataset of kitchen activities. The TACoS corpus [Regneri *et al.*, 2013] is a multimodal corpus that consists of 127 videos of 21 people performing 26 types of kitchen tasks. Each kitchen video is annotated with text descriptions by multiple Amazon Mechanical Turk workers, resulting in significantly richer and more diverse language compared to the Wetlab dataset. There are 2204 text descriptions for the 127 kitchen videos. We aim to align the sentences in these text descriptions with their corresponding video segments. The TACoS dataset includes the ground truth manual segmentation and alignments to corresponding text sentences, which we use for evaluating our automated alignment results.

Unlike the Wetlab videos, the assumption of each object having different color distributions is not valid in the TACoS dataset. As a result, object and hand tracking is significantly harder, making it even more crucial to incorporate actions. Moreover, multiple consecutive sentences may indicate different actions with the same object. It is essential to detect the actions to correctly align these sentences to their corresponding video frames.

TACoS Evaluation

To evaluate the performance of the alignment in various scenarios, we consider two different situations with different amounts of prior knowledge:

1. *Segmented:* The ground truth segmentation is known, but their alignment to the text sentences is unknown. We apply action clusters only on these ground truth segments, and exclude other frames that do not belong to any sentence in the text description.

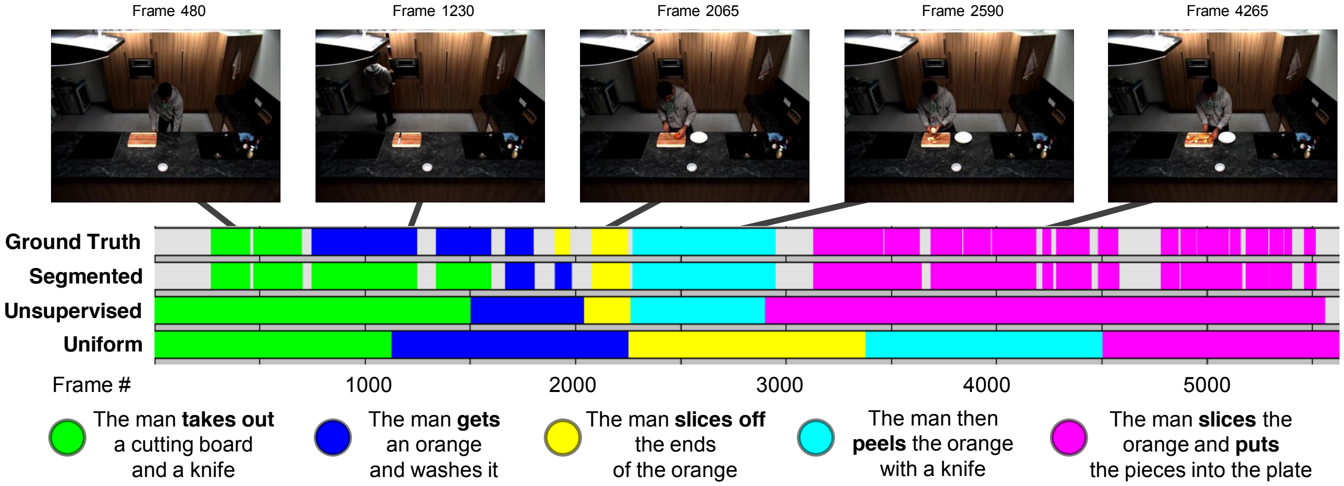

Figure 2: An example of text and video alignment generated by our system on the TACoS corpus for sequence *s13-d28*. Alignments for Ground truth, Segmented, Unsupervised (+STIP), and Uniform are shown. Each color-coded, crowd annotated sentence (bottom) is aligned to a set of video frames.

2. *Unsupervised:* A fully unsupervised setting where both the segmentation and alignment are unknown. We apply a fixed-duration sliding window on the video, and assign each window to one of the action clusters. All the frames in a video are considered for clustering and alignment. However, when we measure the alignment accuracy, we exclude the video frames that do not align to any text sentences in the ground truth data.

We compare the results with a uniform/diagonal baseline alignment, which assigns an equal number of video chunks to each of the text sentences. Figure 2 shows a detailed alignment output for one of the sequences in the dataset. Each sentence is aligned with a set of video frames, shown as color-coded intervals. In the segmented case, action intervals from the ground truth are given, and the alignment matches each action interval to its most likely sentence. In the unsupervised case, the alignment also provides a segmentation of the sequence. It is important to note that while ground truth annotations are used for evaluations, action intervals are often ambiguous and there are cases where system generated alignments provide better results than the ground truth.

Table 2 shows the alignment accuracy of actions on the TACoS dataset. We list our results using hyperfeature variables $\{d^{(1)}, w, d^{(2)}\} = \{128, 75, 64\}$ for STIP, and $\{64, 150, 64\}$ for CNN features. Each LCRF iteration per video took an average of 125 seconds on a single core of a 2.4GHz Intel Xeon processor with 32GB of RAM. The alignment accuracy is significantly higher for the segmented case, since different actions take different amounts of time, and a fixed duration window may split an activity in the wrong places and introduce errors. However, the unsupervised fixed-window approach outperforms the baseline by a large margin.

Table 3 looks at the effects of varying hyperfeature variables, in particular the window size w. While different actions in the TACoS dataset are of varying lengths, our top accuracy is at w=150 frames or 5 seconds, with accuracy rapidly dropping off at extremely short or long window intervals. This is consistent with the timespan of an average action in the TACoS dataset, which is around 5 seconds.

While our experiments are a step in showing that our system is capable of generating high-level hyperfeatures for actions in an unsupervised manner, there are limitations in our method that still need addressing. The ideal number of clusters will vary depending on the number of actions at each point of aggregation, and ideal window size will vary on the length of the action. We use k-means and fixed windows for efficiency and have tested our dataset on a variety of different hyperfeature settings, but results may improve by evaluating over different methods of clustering and aggregation.

5 Conclusion and Future Work

In this paper, we present a framework for recognizing actions from text descriptions and multiple instances of video. While previous approaches for unsupervised alignment of language and video primarily exploited the co-occurrences between nouns and blobs, we extend prior work by including the alignment of actions with verbs and show an improvement in overall alignment accuracy. Furthermore, incorporating actions allows us to align text with complex videos, for which object tracking is extremely difficult (*e.g.*, TACoS dataset).

We introduce the concept of hyperfeatures for actions, where we use low-level action features combined with unsupervised clustering to generate temporally consistent action fragments and clusters, which we then use as input to the alignment system. We evaluate our framework on two activity datasets, and demonstrate the effectiveness of generating action labels from weakly supervised datasets. We expect our framework to be effective in various domains with even larger datasets and overlapping actions. We also plan on extending our model to different parts of speech (*e.g.*, prepositions), and recognizing spatial and temporal relationships between objects and/or actions.

Acknowledgements

This work was supported by the Intel Science & Technology Center for Pervasive Computing (ISTC-PC), NSF award #1319378, DOD-SBIR award #N00014-12-C-0263, ONR award #N00014-11-10417, and the NYS Center of Excellence in Data Science.

References

[Achanta *et al.*, 2012] Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, and Sabine Susstrunk. Slic superpixels compared to state-of-the-art superpixel methods. *IEEE Trans. Pattern Anal. Mach. Intell.*, 34(11):2274–2282, November 2012.

[Agarwal and Triggs, 2006] Ankur Agarwal and Bill Triggs. Hyperfeatures - multilevel local coding for visual recognition. In Ales Leonardis, Horst Bischof, and Axel Pinz, editors, *ECCV*, volume 3951, pages 30–43, Graz, Autriche, 2006. Springer.

[Barbič *et al.*, 2004] Jernej Barbič, Alla Safonova, Jia-Yu Pan, Christos Faloutsos, Jessica K. Hodgins, and Nancy S. Pollard. Segmenting motion capture data into distinct behaviors. In *Proceedings of Graphics Interface*, pages 185–194, 2004.

[Bojanowski *et al.*, 2014] Piotr Bojanowski, Rémi Lajugie, Francis Bach, Ivan Laptev, Jean Ponce, Cordelia Schmid, and Josef Sivic. Weakly supervised action labeling in videos under ordering constraints. In *Proceedings of ECCV*, 2014.

[Bojanowski *et al.*, 2015] Piotr Bojanowski, Rémi Lajugie, Edouard Grave, Francis Bach, Ivan Laptev, Jean Ponce, and Cordelia Schmid. Weakly-Supervised Alignment of Video With Text. In *Proceedings of ICCV*, Santiago, Chile, 2015.

[Coates and Ng, 2012] Adam Coates and Andrew Y. Ng. Learning feature representations with k-means. In *Neural Networks: Tricks of the Trade - Second Edition*, pages 561–580. Springer, 2012.

[Gkioxari and Malik, 2015] G. Gkioxari and J. Malik. Finding action tubes. In *Proceedings of CVPR*, 2015.

[Harchaoui *et al.*, 2009] Zaïd Harchaoui, Eric Moulines, and Francis R. Bach. Kernel change-point analysis. In D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, editors, *Advances in Neural Information Processing Systems 21*, pages 609–616. Curran Associates, Inc., 2009.

[Kate and Mooney, 2007] Rohit J Kate and Raymond J Mooney. Learning language semantics from ambiguous supervision. In *AAAI*, volume 7, pages 895–900, 2007.

[Kipp, 2012] M Kipp. Anvil: A universal video research tool. *Handbook of Corpus Phonology. Oxford University Press*, 2012.

[Kollar *et al.*, 2010] Thomas Kollar, Stefanie Tellex, Deb Roy, and Nicholas Roy. Toward understanding natural language directions. In *5th ACM/IEEE International Conference on Human-Robot Interaction*, pages 259–266. IEEE, 2010.

[Krizhevsky *et al.*, 2012] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C.J.C. Burges, L. Bottou, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012.

[Laptev, 2005] Ivan Laptev. On space-time interest points. *Int. J. Comput. Vision*, 64(2-3):107–123, September 2005.

[LeCun *et al.*, 2001] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. In S. Haykin and B. Kosko, editors, *Intelligent Signal Processing*, pages 306–351. IEEE Press, 2001.

[Lowe, 2004] David G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 60(2):91–110, November 2004.

[Luo and Guo, 2003] Jiebo Luo and Cheng-en Guo. Perceptual grouping of segmented regions in color images. *Pattern Recognition*, 36(12):2781–2792, 2003.

[Malmaud *et al.*, 2015] Jonathan Malmaud, Jonathan Huang, Vivek Rathod, Nicholas Johnston, Andrew Rabinovich, and Kevin Murphy. What's cookin'? interpreting cooking videos using text, speech and vision. In *Proceedings of NAACL HLT*, pages 143–152, Denver, CO, USA, 2015.

[Naim *et al.*, 2014] Iftekhar Naim, Young Chol Song, Qiguang Liu, Henry A. Kautz, Jiebo Luo, and Daniel Gildea. Unsupervised alignment of natural language instructions with video segments. In *Proceedings of AAAI*, pages 1558–1564, Québec City, Québec, Canada, 2014.

[Naim *et al.*, 2015] Iftekhar Naim, Young Chol Song, Qiguang Liu, Henry A. Kautz, Jiebo Luo, and Daniel Gildea. Discriminative unsupervised alignment of natural language instructions with corresponding video segments. In *Proceedings of NAACL HLT*, Denver, CO, USA, 2015.

[Regneri *et al.*, 2013] Michaela Regneri, Marcus Rohrbach, Dominikus Wetzel, Stefan Thater, Bernt Schiele, and Manfred Pinkal. Grounding action descriptions in videos. *Transactions of the Association for Computational Linguistics (TACL)*, 1:25–36, 2013.

[Soomro *et al.*, 2012] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. UCF101: A dataset of 101 human actions classes from videos in the wild. In *CRCV-TR-12-01*, 2012.

[Tellex *et al.*, 2014] Stefanie Tellex, Pratiksha Thaker, Joshua Joseph, and Nicholas Roy. Learning perceptually grounded word meanings from unaligned parallel data. *Machine Learning*, 94(2):151–167, 2014.

[Wang and Schmid, 2013] Heng Wang and Cordelia Schmid. Action Recognition with Improved Trajectories. In *Proceedings of ICCV*, pages 3551–3558, Sydney, Australia, December 2013. IEEE.

[Wang *et al.*, 2009] Heng Wang, Muhammad Muneeb Ullah, Alexander Klaser, Ivan Laptev, and Cordelia Schmid. Evaluation of local spatio-temporal features for action recognition. In *Proceedings of the British Machine Vision Conference*, pages 124.1–124.11. BMVA Press, 2009.

[Wang *et al.*, 2011] Heng Wang, A. Klaser, C. Schmid, and Cheng-Lin Liu. Action recognition by dense trajectories. In *Proceedings of CVPR*, pages 3169–3176, Washington, DC, USA, 2011. IEEE.

[Wang *et al.*, 2015] Limin Wang, Yu Qiao, and Xiaoou Tang. Action recognition with trajectory-pooled deep-convolutional descriptors. In *Proceedings of CVPR*, pages 4305–4314, 2015.

[Yang *et al.*, 2015] Yezhou Yang, Yi Li, Cornelia Fermuller, and Yiannis Aloimonos. Robot learning manipulation action plans by Watching unconstrained videos from world wide web. In *Proceedings of AAAI*, Austin, US, 2015.

[Yu and Ballard, 2004] Chen Yu and Dana H Ballard. On the integration of grounding language and learning objects. In *Proceedings of AAAI*, volume 4, pages 488–493, 2004.

[Zhou *et al.*, 2013] Feng Zhou, Fernando De la Torre, and Jessica K. Hodgins. Hierarchical aligned cluster analysis for temporal clustering of human motion. *IEEE Transactions Pattern Analysis and Machine Intelligence (PAMI)*, 35(3):582–596, 2013.