Connectionist Temporal Modeling for Weakly Supervised Action Labeling

De-An Huang, Li Fei-Fei, Juan Carlos Niebles

Computer Science Department, Stanford University

Abstract. We propose a weakly-supervised framework for action labeling in video, where only the order of occurring actions is required during training time. The key challenge is that the per-frame alignments between the input (video) and label (action) sequences are unknown during training. We address this by introducing the Extended Connectionist Temporal Classification (ECTC) framework to efficiently evaluate all possible alignments via dynamic programming and explicitly enforce their consistency with frame-to-frame visual similarities. This protects the model from distractions of visually inconsistent or degenerated alignments without the need of temporal supervision. We further extend our framework to the semi-supervised case when a few frames are sparsely annotated in a video. With less than 1% of labeled frames per video, our method is able to outperform existing semi-supervised approaches and achieve comparable performance to that of fully supervised approaches.

1 Introduction

With the rising popularity of video sharing sites like YouTube, there is a large amount of visual data uploaded to the Internet. This has stimulated recent developments of large-scale action understanding in videos [1–5]. Supervised learning methods can be effective for action recognition, but fully annotating actions in videos at large scale is costly and time-consuming in practice. An alternative is to develop methods that require weak supervision, which may be automatically extracted from movie scripts [6–9] or instructional videos [10–12] at a lower cost.

In this work, we address the problem of weakly-supervised action labeling in videos. In this setting, only incomplete temporal localization of actions is available during training, and the goal is to train models that can be applied in new videos to annotate each frame with an action label. This is challenging as the algorithm must reason not only about whether an action occurs in a video, but also about its exact temporal location. Our setting contrasts with most existing works [13–17] for action labeling that require fully annotated videos with accurate per frame action labels for training. Here, we aim at achieving comparable temporal action localization *without* temporal supervision in training.

The setting of our work is illustrated in Figure 1. During training, only the order of the occurring actions is given, and the goal is to apply the learned model to unseen test videos. As no temporal localization is provided during training, the first challenge of our task is that there is a large number of possible

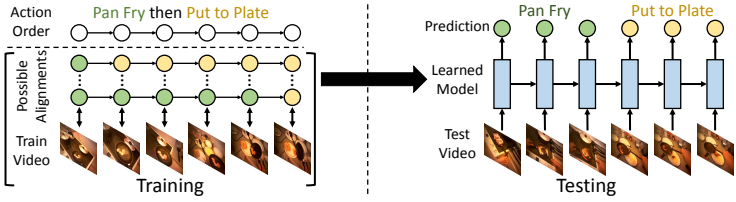

Fig. 1. We tackle the problem of weakly supervised action labeling where only the order of the occurring actions is given during training (left). We train a temporal model by maximizing the probability of all possible frame-to-label alignments. At testing time (right), no annotation is given. As our learned model already encodes the temporal structure of videos, it predicts the correct actions without further information.

alignments (or correspondences) between action labels and video frames, and it is infeasible to naively search through all of these alignments. We address this challenge by first introducing Connectionist Temporal Classification (CTC) [18], originally designed for speech recognition, to our video understanding task. CTC efficiently evaluates all of the possible alignments using dynamic programming.

Directly applying the original CTC framework to our weakly-supervised action labeling could not fully address the challenge of a large space of possible frame to action label alignments. Note that the duration of an action could be hundreds of frames, which is much longer than the duration of phonetic states in speech recognition. As a result, we are required to align videos of thousands of frames to only dozens of actions. This poses a unique challenge in comparison to speech, as our space of possible alignments is much larger and contains degenerated alignments that can deteriorate performance. We address this challenge by proposing the Extended Connectionist Temporal Classification (ECTC) framework, which explicitly enforces the alignments to be consistent with frame-to-frame visual similarities. The incorporation of similarity allows us to (1) explicitly encourage the model to output visually consistent alignments instead of fitting to the giant space of all possible alignments (2) down-weight degenerated paths that are visually inconsistent. In addition, we extend the forward-backward algorithm of [18] to incorporate visual similarity, which allows us to efficiently evaluate all of the possible alignments while explicitly enforcing their consistency with frame-to-frame similarities at the same time.

While our main focus is the weakly supervised setting, we also show how to extend our approach to incorporate supervision beyond action ordering. To this end, we introduce the *frame-level* semi-supervised setting, where action labels are temporally localized in a few annotated video frames. This supervision could be extracted from movie scripts [8, 9] or by asking annotators to label actions for a small number of frames in the video, which is less costly than precisely annotating temporal boundaries of all actions. We model such supervision as a frame to label alignment constraints and naturally incorporate it in our ECTC framework to efficiently prune out inconsistent alignments. This significantly reduces the alignment space and boosts the performance of our approach.

The main contributions of our work can be summarized as: (1) We first introduce CTC to our video understanding task, as a way to efficiently evaluate all frame to action alignments. (2) We propose ECTC to explicitly enforce the consistency of alignments with visual similarities, which protects the model from distractions of visually inconsistent alignments without the need of temporal supervision. (3) We extend ECTC to incorporate *frame-level* semi-supervision in a unified framework to significantly reduce the space of possible alignments. (4) We test our model on long and complex activity videos from the Breakfast Actions Dataset [13] and a subset of the Hollywood2 dataset [7], and show that our method achieves state-of-the-art performance with less than 1% of supervision.

2 Related Work

As significant progress has been made on categorizing temporally trimmed video clips, recent research of human activity analysis is shifting towards a higher level understanding in real-world settings [5, 16, 19–25]. Two tasks of action labeling have been explored extensively. The first is video classification, where the goal is to categorize each video to a discrete action class. Challenging datasets including UCF101 [1], HMDB51 [2], Sports1M [3], THUMOS [4], and ActivityNet [5] exemplify this. Deep neural networks trained directly from videos [26, 27] have shown promising results on this task [28]. The second is dense action labeling, where the goal is to label each frame with the occurring actions [13–17], and the fully annotated temporal boundaries of actions are given during training.

In this paper, we aim to achieve action labeling with a weaker level of supervision that is easier to obtain than accurately time-stamped action labels. A similar goal has been explored in video to action label alignment [7, 11]. The closest to our work is the ordering constrained discriminative clustering (OCDC) approach of Bojanowski *et al.* [7], where the goal is to align video frames to an ordered list of actions. By exploiting the ordering constraint during the alignment, OCDC extends previous work [8] to deal with multiple actions in a video clip. As their focus is on video to action label alignment, their method can assume that the ordering of actions is always available, both at training and testing time. Our approach aims at a more general scenario, where the learned model is applied to unseen test videos that come without information about the actions that appear in the video. When applied to this more general scenario, OCDC is equivalent to a frame-by-frame action classifier that was implicitly learned during the training alignment. Therefore, this form of model does not fully exploit temporal information at test time, since it does not need to encapsulate the temporal relationships provided by the ordering supervision. This may limit its applicability to temporally structured complex activities. On the other hand, our temporal modeling exploits the temporal structure of actions in videos, such as the duration of actions and transitions between actions, by capturing them during training and leveraging at test time.

Our work is also related to recent progress on using instructional videos or movie scripts [8, 10, 11, 22, 29–31] as supervision for video parsing. These ap-

proaches also tackle the case when some text is available for alignment at testing time, and focus more on the natural language processing side of understanding the text in the instructions or the scripts. In this paper, we focus on training a temporal model that is applicable to unseen test videos that come without associated text. Our supervision could potentially be obtained with some of these text processing approaches, but this is not the focus of our work.

Our goal of understanding the temporal structure of video is related to [32–38]. In contrast to their goal of classifying the whole video to a single action, our goal is to utilize the temporal structure of videos to guide the training of an action labeling model that can predict the occurring action at every frame in the unseen test video. Our use of visual similarities in the training is related to unsupervised video parsing [21, 39, 40], where frames are grouped into segments based on visual or semantic cues. We integrate visual similarity with weak supervision as a soft guidance of the model and go beyond just grouping video frames.

The core of our model builds upon Recurrent Neural Networks (RNN), which have been proved effective for capturing the temporal dependencies in data, and have been applied to challenging computer vision tasks including image captioning [26, 41, 42], video description [26, 43, 44], activity recognition [26, 45], dense video labeling [17]. However, in the above tasks, accurate temporal localization of actions is either ignored or requires pre-segmented training data. Our ECTC framework enables learning recurrent temporal models with weak supervision, and we show empirically its effectiveness on the video action labeling task.

3 Ordering Constrained Video Action Labeling

Our goal is to train a temporal model to assign action labels to every frame of unseen test videos. We use a Recurrent Neural Network (RNN) at the core of our approach, as it has been successfully applied to label actions in videos [17, 26]. While RNNs have been generally trained with full supervision in previous work, we aim to train them with weak supervision in the form of an ordered list of occurring actions. We address this challenge by proposing the Extended Connectionist Temporal Classification (ECTC) framework that efficiently evaluates all possible frame to action alignments and weights them by their consistency with the visual similarity of consecutive frames. The use of visual similarities sets our approach apart from the direct application of CTC [18] and alleviates the problem caused by visually inconsistent alignments. ECTC incorporates a frame dependent binary term on top of the original unary based model, and we show that this can be efficiently handled by our forward-backward algorithm.

3.1 Extended Connectionist Temporal Classification

The biggest challenge of our task is that only the order of the actions is given during training. Formally, given a training set consisting of video examples $X = [x_1, \cdots, x_T] \in \mathbb{R}^{d \times T}$ represented by d-dimensional features x_t extracted from each of their T frames, our goal is to infer the associated action labels $a =$

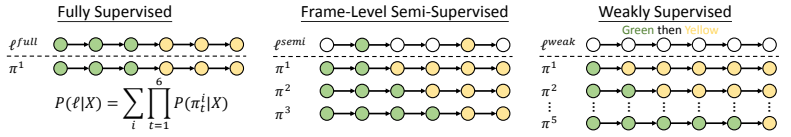

Fig. 2. Comparison of different levels of supervision (first row). Blank circles indicate frames without annotated action. The probability of ℓ is given by the sum of the probabilities of all the paths π^i that are consistent with it.

$[a_1, \cdots, a_T] \in \mathcal{A}^{1\times T}$, where $\mathcal{A}$ is a fixed set of possible actions. Note that a is not available for our training examples. Instead, the supervision we have for each video is the order of actions $\ell = \mathcal{B}(a)$, where $\mathcal{B}$ is the operator that removes the repeated labels. For example, $\mathcal{B}([b, b, c, c, c]) = [b, c]$. Our goal is to learn a temporal model using this supervision, and apply it to unseen test videos for which neither ℓ nor a are available. We build our temporal models with an RNN at the core. Let $Y = [y_1, \cdots, y_T] \in \mathbb{R}^{A\times T}$ be the RNN output at each frame, where $A = |\mathcal{A}|$ is the number of possible actions. We normalize the output vectors y_t using a softmax to get $z_t^k = P(k, t|X) = e^{y_t^k}/\sum_{k'} e^{y_t^{k'}}$, which can be interpreted as the probability of emitting action k at time t.

In the original CTC formulation [18], the conditional independence assumption states that the probability of a label sequence $\pi = [\pi_1, \cdots, \pi_T]$ is:

$$P(\pi|X) = \prod_{t=1}^{T} z_t^{\pi_t}, \tag{1}$$

which corresponds to the stepwise product of $z_t^{\pi_t}$ at each frame. Note that we distinguish a *path* π that indicates per-frame label information from the *label sequence* $\ell = \mathcal{B}(\pi)$ which only contains the ordering of actions and no precise temporal localization of labels. Label sequence ℓ is computed from path π by $\mathcal{B}(\pi)$, which removes all the consecutive label repetitions. We can compute the probability of emitting a label sequence ℓ, by summing the probability of all paths π that can be reduced to ℓ using the operator $\mathcal{B}$:

$$P(\ell|X) = \sum_{\{\pi|\mathcal{B}(\pi)=\ell\}} P(\pi|X). \tag{2}$$

Given the label sequence ℓ for each training video X, model learning is formulated as minimizing $\mathcal{L}(\ell, X) = -\log P(\ell|X)$, the negative log likelihood of emitting ℓ. The intuition is that, because we do not have the exact temporal location of a label, we sum over all the frame to label alignments that are consistent with ℓ [46]. One drawback of this original CTC formulation in Eq.(1) is that it does not take into account the fact that consecutive frames in the video are highly correlated, especially visually similar ones. This is important as the sum in Eq.(2) might thus include label paths π that are visually inconsistent with the video contents and thus deteriorate the performance. In the following, we discuss how our ECTC uses visual similarity to reweight the possible paths.

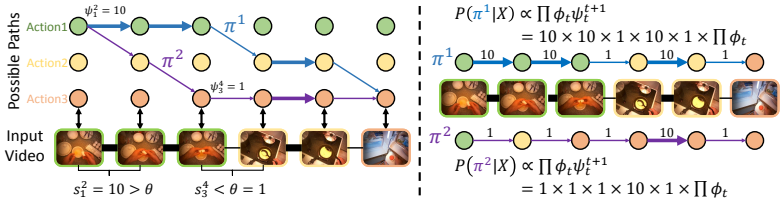

Fig. 3. Our ECTC framework uses the binary term ψ_t^{t+1} to re-weight paths. In this example, an input video has 6 frames and 3 annotated actions. Thicker connections between frames indicate higher similarity. In ECTC, π^1 has higher weight than π^2 since it stays in the same action for similar frames. In the example, π^1 actually matches the ground truth actions. In contrast, both paths are weighted equally in CTC.

We introduce the Extended CTC framework to address such limitations. To illustrate our framework, assume that $z_t^a = z_t^b$ for all t in a short clip of visually similar frames. In this example, the probability of the path $[a, b, a, b]$ will be the same as $[a, a, b, b]$ using Eq.(1). Clearly the latter path should be more probable, as action labels are usually smooth and stay the same for visually similar frames. Such intuition, however, is not compatible with Eq.(1). While our RNN could implicitly encode such pattern from training observations, we reformulate Eq.(1) to explicitly enforce the incorporation of visual similarity between consecutive frames by rewarding visually consistent paths:

$$P(\pi|X) \propto \prod_{t=1}^{T} \phi_t \psi_t^{t+1}, \quad \phi_t = z_t^{\pi_t}, \quad \psi_t^{t+1} = \begin{cases} \max(\theta, s_t^{t+1}) & \pi_t = \pi_{t+1} \\ \theta & \pi_t \neq \pi_{t+1} \end{cases}. \tag{3}$$

The path probability now includes both a unary term ϕ_t and a binary term ψ_t^{t+1}. The unary term is defined as $z_t^{\pi_t}$ and represents the original formulation. We introduce the binary term ψ_t^{t+1} to explicitly capture the correlation between consecutive frames, where θ is a predefined minimum similarity, and $s_t^{t+1} = \text{sim}(x_t, x_{t+1})$ is the similarity between frames. When $\pi_t = \pi_{t+1}$ and $s_t^{t+1} > \theta$ (the two frames are similar), $\psi_t^{t+1} = s_t^{t+1}$ can be seen as a reward for staying at the same action. Effectively, our binary term explicitly rewards the paths that have the same action for visually similar frames, which further encourages the model to generate visually consistent action labels. On the other hand, frames with low similarity are not penalized for having the same action. When $\pi_t = \pi_{t+1}$ and $s_t^{t+1} < \theta$ (low similarity), $\psi_t^{t+1} = \theta$ is simply the same for all and has no effect on the path probability after normalization. Consider an extreme case when $s_t^{t+1} = \infty$. This effectively imposes the constraint that $\pi_t = \pi_{t+1}$, as the probability of paths with $\pi_t \neq \pi_{t+1}$ will be zero after normalization. As we will show in the experiment, our explicit modeling of the frame-to-frame correlation with the binary term plays an important role to the success of our model, as it allows us to avoid visually inconsistent and trivial paths in our task. Figure 3 shows an example of how our ECTC reweights the paths using visual consistency.

3.2 ECTC Forward-Backward Algorithm

At first sight, the summation in Eq.(2) seems problematic, as the number of paths grows exponentially with the length of the input sequence. This is further complicated by the fact that our formulation in Eq.(3) involves a binary term ψ_t^{t+1} that depends on both frame t and $t+1$. We address this by proposing the ECTC forward-backward algorithm that extends the approach in [18] and naturally incorporates the visual similarity function in a unified framework. We will show how the proposed algorithm is still able to efficiently evaluate all of the possible paths using dynamic programming despite the introduction of the binary term in Eq.(3) to explicitly capture the correlation between consecutive frames. We define our *forward variable* as

$$\alpha(s,t) = \sum_{\{\pi_{1:t}|\mathcal{B}(\pi_{1:t})=\ell_{1:s}\}} P(\pi_{1:t}|X) \tag{4}$$

$$\propto \sum_{\{\pi_{1:t}|\mathcal{B}(\pi_{1:t})=\ell_{1:s}\}} \prod_{t'=1}^{t} \Psi_{t'}^{\pi_{t'}} z_{t'}^{\pi_{t'}}, \quad \Psi_t^k = \begin{cases} \max(\theta, s_{t-1}^t) & k = \pi_{t-1} \\ \theta & k \neq \pi_{t-1} \end{cases}, \tag{5}$$

which corresponds to the sum of probabilities of paths with length t $\pi_{1:t} = [\pi_1, \cdots, \pi_t]$ that satisfy $\mathcal{B}(\pi_{1:t}) = \ell_{1:s}$, where $\ell_{1:s}$ is the first s elements of the label sequence ℓ. We also introduce a new variable Ψ_t^k for explicitly modeling the dependence between consecutive frames and encourage the model to output visually consistent path. This makes the the original CTC forward-backward algorithm not directly applicable to our formulation. By deriving all $\pi_{1:t}$ that satisfy $\mathcal{B}(\pi_{1:t}) = \ell_{1:s}$ from $\pi_{1:t-1}$, the forward recursion is formulated as:

$$\alpha(s,t) = \hat{z}_t^{\pi_t} \alpha(s, t-1) + \tilde{z}_t^{\pi_t} \alpha(s-1, t-1), \tag{6}$$

where

$$\hat{z}_t^{\pi_t} = \frac{\Psi_t^{\pi_t} z_t^{\pi_t}}{\sum_{k=1}^{A} \Psi_t^k z_t^k} = \frac{\max(\theta, s_{t-1}^t) z_t^{\pi_t}}{\max(\theta, s_{t-1}^t) z_t^{\pi_t} + \theta(1 - z_t^{\pi_t})} \tag{7}$$

$$\tilde{z}_t^{\pi_t} = \frac{\Psi_t^{\pi_t} z_t^{\pi_t}}{\sum_{k=1}^{A} \Psi_t^k z_t^k} = \frac{\theta z_t^{\pi_t}}{\max(\theta, s_{t-1}^t) z_t^{\pi_{t-1}} + \theta(1 - z_t^{\pi_{t-1}})}. \tag{8}$$

The key difference between our algorithm and that of [18] is the renormalization of z_t^k using frame similarity Ψ_t^k, which in turn gives the renormalized $\hat{z}_t^{\pi_t}$ and $\tilde{z}_t^{\pi_t}$. This efficiently incorporates visual similarity in the dynamic programming framework and encourages the model towards visually consistent paths. The first term in Eq.(6) corresponds to the case when $\pi_t = \pi_{t-1}$. Based on the definition, we have $\Psi_t^{\pi_t} = \Psi_t^{\pi_{t-1}} = \max(\theta, s_{t-1}^t)$. Intuitively, this reweighting using Ψ_t^k will reward and raise $z_t^{\pi_t}$ to $\hat{z}_t^{\pi_t}$ for having the same action label for similar consecutive frames. On the other hand, the second term in Eq.(6) is for the case when $\pi_t \neq \pi_{t-1}$, and thus $\Psi_t^{\pi_t} = \theta$. In this case, the probability is taken from $\tilde{z}_t^{\pi_t}$ to reward $\tilde{z}_t^{\pi_{t-1}}$, and thus $\tilde{z}_t^{\pi_t}$ will be smaller than $z_t^{\pi_t}$.

The *backward variable* is similarly defined as:

$$\beta(s,t) = \sum_{\{\pi_{t:T}|\mathcal{B}(\pi_{t:T})=\ell_{s:S}\}} P(\pi_{t:T}|X) \propto \sum_{\{\pi_{t:T}|\mathcal{B}(\pi_{t:T})=\ell_{s:S}\}} \prod_{t'=t}^{T} \tilde{\Psi}_{t'}^{\pi_{t'}} z_{t'}^{\pi_{t'}}, \tag{9}$$

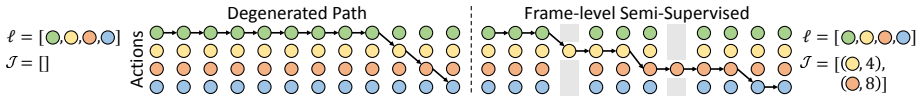

Fig. 4. Example of a degenerated path and a semi-supervised path. On the right, gray blocks constrain the path to be consistent with the two supervised frames. This significantly reduces the space of possible paths and prevents degenerated paths.

the sum of the probability of all paths starting at t that will complete ℓ when appending from $t+1$ to any path of $\alpha(s,t)$. We also introduce $\tilde{\Psi}_t^k$ in the same way as Ψ_t^k, but by decomposing Eq.(3) backward rather than forward. The backward recursion to compute $\beta(s,t)$ can be derived similarly to the forward recursion in Eq.(6), but by deriving $\pi_{t:T}$ from $\pi_{t+1:T}$. Based on the definition of forward and backward variables, we have $P(\ell|X) = \sum_{s=1}^{S} \frac{\alpha(s,t)\beta(s,t)}{z_t^{\ell_s}}$.

Optimization. With this forward-backward algorithm, we are able to compute the gradient of the loss function $\mathcal{L}(\ell, X)$ w.r.t. the recurrent neural network output y_t^k, the response of label k at time t. The gradient is given by:

$$\frac{\partial \mathcal{L}(\ell, X)}{\partial y_t^k} = z_t^k - \frac{1}{P(\ell|X)} \sum_{s:\ell_s=k} \frac{\alpha(s,t)\beta(s,t)}{z_t^{\ell_s}}, \tag{10}$$

where the first term is the softmax output. The second term can be seen as the softmax target.The second term can be intuitively interpreted as $P(\pi_t = k|\mathcal{B}(\pi) = \ell, X)$, which is the probability of choosing action k at time t for paths that are consistent with the sequence label ℓ (reweighted by ψ_t^{t+1}). The recurrent neural network can then be optimized through back propagation [47].

4 Extension to Frame-level Semi-Supervised Learning

When only the ordering supervision is available, all of the paths π that are consistent with ℓ are considered in Eq.(2). A quick observation, however, shows that some undesirable or degenerate paths shown in Figure 4 are also considered in the summation. This challenge is unique to our task as the length of the label sequence ℓ is usually much shorter than the number of frames, which is not the case in speech recognition. We have shown how our ECTC can be used in this case as soft constraints to down-weight such visually inconsistent paths and reward the ones that have consistent labels for visually similar frames. Nevertheless, when supervision beyond ordering is available, we can derive harder constraints for the paths and effectively remove undesirable paths from the summation.

In this section, we show that sparse temporal supervision can also be naturally incorporated in our framework and efficiently prune out the degenerated paths. We introduce the *frame-level* semi-supervised setting, where only a few frames in the video are annotated with the ground truth action. Such supervision

could be automatically extracted from movie scripts [8, 9] or by annotating a few frames of the video, which is much easier than finding the exact temporal boundaries of all the actions. Formally, the supervision we consider is a list of frames with the corresponding action labels: $\mathcal{J} = [(a_1, t_1), \cdots, (a_m, t_m), \cdots (a_M, t_M)]$, where each element of the list is a pair of frame index t_m and the corresponding action label a_m. This can significantly reduce the number of possible paths when combined with the order of the actions. For example, assuming that we have $\mathcal{J} = [(a, 2), (b, 4)]$ and $\ell = [a, b]$ for a video of length 6, then there are only two possible paths ($[a, a, b, b, b, b]$ and $[a, a, a, b, b, b]$) that are consistent with the supervision. This not only significantly reduces the space of consistent paths, but also avoids undesirable paths like $[a, a, a, a, a, b]$. Figure 4 also shows an example of the effect of the frame-level semi-supervision. This supervision can be naturally incorporated by extending the recursion in Eq.(6) as:

$$\alpha(s,t) = \begin{cases} 0, & \exists (a_m, t_m) \in \mathcal{J}, \text{ s.t. } t = t_m \text{ but } s \neq a_m \\ \hat{z}_t^{\pi_t} \alpha(s, t-1) + \tilde{z}_t^{\pi_t} \alpha(s-1, t-1), & \text{otherwise} \end{cases}, \tag{11}$$

where an extra checking step is applied to ensure that the path is consistent with the given semi-supervision. We will show that, with less than 1% of frames being labeled, our approach can perform comparably to fully supervised model.

5 Experiments

We evaluate our model on two challenging tasks and datasets. The first is segmentation of cooking activity video in the Breakfast Actions Dataset [13]. The output action labeling divides the video into temporal segments of cooking steps. Because of the dependencies between temporally adjacent actions in cooking activities, the capacity of the model to handle temporal dynamics is especially important. The second task is action detection on videos in a subset of the Hollywood2 dataset [9], with a setting introduced by [7]. Our action labeling framework can be applied to action detection by considering an additional background label $\emptyset$ to indicate frames without actions of interest.

5.1 Implementation Details

Network Architecture. We use 1-layer Bidirectional LSTM (BLSTM) [48] with 256 hidden units for our approach. We cross-validate the learning rate and the weight decay. For the optimization, we use SGD with batch size 1. We clip gradients elementwise at 5 and scale gradients using RMSProp [49].

Visual Similarity. For our ECTC, we combine two types of visual similarity functions. The first is clustering of visually similar and temporally adjacent frames. We apply k-means clustering to frames in a way similar to SLIC [50] to over-segment the video. We initialize $\frac{T}{M}$ centers uniformly for a video, where T is the video length, and M is the average number of frames in a cluster. We empirically pick $M = 20$, which is much shorter than the average length of an

action (∼400 frames in the Breakfast Dataset) to conservatively over segment the video and avoid grouping frames that belong to different actions. The resulting grouping is in the form of constraints such as $\pi_t = \pi_{t+1}$, which can be easily incorporated in our ECTC by setting s_t^{t+1} to ∞. We thus set $s_t^{t+1} = \infty$ if the video frames x_t and x_{t+1} are in the same cluster and $s_t^{t+1} = 0$ otherwise. The second visual similarity function we consider is $s_t^{t+1} \propto \frac{x_t \cdot x_{t+1}}{|x_t||x_{t+1}|}$, the cosine similarity of the frames. This formulation will reward paths that assign visually similar frames to the same action and guide the search of alignment during the forward-backward algorithm. We combine the two similarity functions by setting s_t^{t+1} to the cosine similarity at the boundary between clusters instead of 0.

5.2 Evaluating Complex Activity Segmentation

In this task, the goal is to segment long activity videos into actions composing the activity. We follow [13] and define the *action units* as the shorter atomic actions that compose the longer and more complex activity. For example, "Take Cup" and "Pour Coffee" are action units that compose the activity "Make Coffee".

Dataset. We evaluate activity segmentation of our model on the Breakfast dataset [13]. The videos of the dataset were recorded from 52 participants in 18 different kitchens conducting 10 distinct cooking activities. This results in ∼77 hours of videos of preparing dishes such as fruit salad and scrambled eggs.

Metrics. We follow the metrics used in previous work [13] to evaluate the parsing and segmentation of action units. The first is *frame accuracy*, the percentage of frames that are correctly labeled. The second is *unit accuracy*. The output action units sequence is first aligned to the ground truth sequence by dynamic time warping (DTW) before the error rate is computed. For weakly supervised approaches, high frame accuracy is harder to achieve than high unit accuracy because it directly measures the quality of the temporal localization of actions.

Features. We follow the feature extraction steps of [51] and use them for all competing methods. First, the improved dense trajectory descriptor [52] is extracted and encoded by Fisher Vector with GMMs=64. L2 and power normalization, and PCA dimension reduction ($d = 64$) are then applied.

Baselines. We compare our method to three baselines. The first is per-frame Support Vector Machine (**SVM**) [53] with RBF kernels.We are interested in how well discriminative classification can do on the video segmentation task without exploiting the temporal information in the videos. The second is Hidden Markov Model Toolkit (**HTK**) used in previous work for this task [13, 51]. The third is Order Constrained Discriminative Clustering (**OCDC**) of Bojanowski *et al.* [7], which has been applied to align video frames with actions.

Ablation Studies. First we analyze the effect of different components of our approach and compare to the baselines. The results are shown in Table 1. The first variation of our model is "**Uniform**". Instead of using our framework to evaluate all possible paths, the target of Uniform is a single path π given by uniformly distributing the occurring actions among frames. We also show the performance of direct application of **CTC** to our task. Without explicitly imposing the alignments to be consistent with the visual similarity, CTC only has

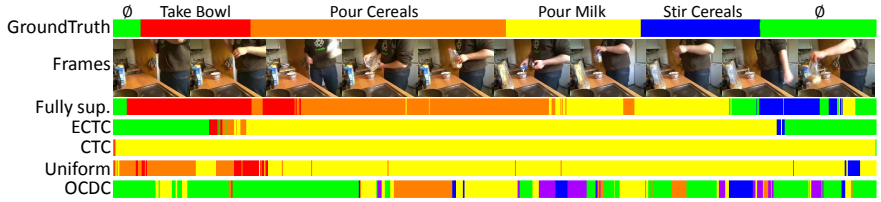

Fig. 5. Qualitative comparison of weakly supervised approaches in a testing video. Fully supervised results using BLSTM are also shown as reference (upper bound of our approach). Colors indicate different actions, and the horizontal axis is time. Per frame classification of OCDC is noisy and contains unrelated actions. The Uniform baseline produces the proper actions, but without alignment and ordering. CTC outputs a degenerated path in this case: while the order is correct, the sequence is dominated by a single action. Our ECTC has better localization and ordering of the actions since we incorporate visual similarity to prune out inconsistent and degenerated paths.

the effect of trading-off frame accuracy for unit accuracy when compared to the Uniform baseline. The reason is that the original CTC objective is actually directly optimizing the unit accuracy, but ignoring the frame accuracy as long as the output action order is correct. The performance of our ECTC with only the clustering similarity is shown as "**ECTC (kmeans)**". This efficiently rules out the paths that are inconsistent with the visual grouping and improve the alignment of actions to frames. Using only the cosine similarity with ECTC ("**ECTC (cosine)**"), we are able to further improve the unit accuracy. Combining both similarities, the last column of Table 1 is our final ECTC model, which further improves the accuracy and outperforms fully supervised baselines. This verifies the advantage of using visual similarity in our ECTC to reward paths that are consistent with it. All variations of our temporal models outperform the linear classifier in OCDC on unseen test videos. Figure 5 shows the qualitative results.

Frame-level Semi-Supervision. Next we study the effect of having more supervision with our model. The results are shown in Figure 7. The x-axis shows the fraction of labeled frames for our frame-level semi-supervision in each video. The minimum supervision we use is when only a single frame is labeled for each occurring action in the video (fraction 0.0025). Fraction 1 indicates our fully supervised performance. The annotation for the Uniform baseline in this case is equally distributed between two sparsely annotated frames. With our approach,

Table 1. Ablation studies of our approach on the Breakfast dataset. Each component introduced in our approach gives an accuracy improvement. Our final ECTC model is able to outperform fully supervised baselines.

Supervision	**Fully Sup.**		**Weakly Sup.**					
Model	SVM [53]	HTK [13]	OCDC [7]	Uniform	CTC	ECTC (kmeans)	ECTC (cosine)	ECTC (Our full model)
Frame Acc.	15.8	19.7	8.9	22.6	21.8	24.5	22.5	**27.7**
Unit Acc.	15.7	20.4	10.4	33.1	36.3	35.0	**36.7**	35.6

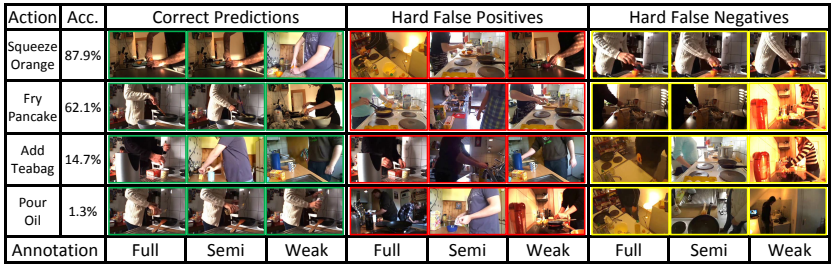

Fig. 6. Example results for the two hardest and easiest actions. Correct Predictions illustrate the most confident correct frame predictions. Hard False Positives show incorrect predictions with high confidence. Our models can be confused by the appearance of objects in the scene (e.g., seeing the teabag box), or by similar motions (e.g., pouring milk instead of oil). Hard false negative show missing detections. We see challenges of viewpoint, illumination, and ambiguities near the annotated boundary between actions.

the frame accuracy is dropping much slower than that of the Uniform baseline, since our approach is able to automatically and iteratively refine the alignment between action and frame during training. Our semi-supervised approach significantly outperforms OCDC with all fractions of labeled frames. The results of HTK, SVM, and our full ECTC are also plotted for reference. As noted earlier, our weakly supervised approach has the highest unit accuracy, as the CTC objective is directly optimizing it. This is consistent with the fact that lower fraction of labeled frames of our approach actually has higher unit accuracy. Another interesting observation is the gap between our weakly supervised model and semi-supervised model. While our weakly supervised model already outperforms several baselines with full supervision, it can be seen that giving only a single frame annotation as an anchor for each segment significantly reduces the space of possible alignments and provides a strong cue to train our temporal model. Figure 6 shows results for different levels of supervision.

Training Set Alignment. While our framework aims at labeling unseen test videos when trained only with the ordering supervision, we also verify whether our action-frame alignment during training also outperforms the baselines. The

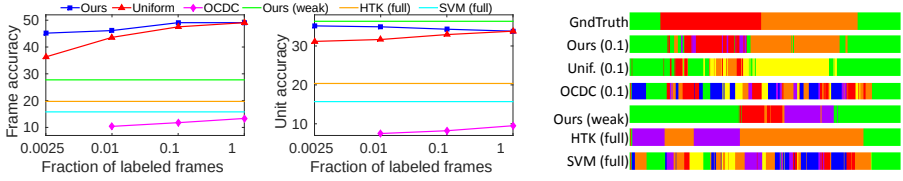

Fig. 7. Frame and unit accuracy in the Breakfast dataset plotted against fraction of labeled data in the frame-level semi-supervised setting. Horizontal lines are either fully-supervised or weakly supervised methods. On the right, qualitative results for one video follow the convention of Figure 5.

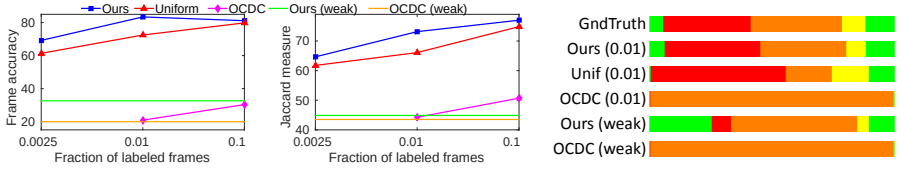

Fig. 8. Frame accuracy, Jaccard measure, and qualitative alignment results on the training set of the Breakfast dataset. Our models also produce good alignments for the training set in addition to the ability to segment unseen test videos.

frame accuracy and Jaccard measure are shown in Figure 8. Jaccard measure is used to evaluate the alignment quality in [7]. OCDC that is directly designed for the alignment problem indeed performs closer to our method in this scenario.

5.3 Evaluating Action Detection

In this task, the goal is to localize actions in the video. This can be formulated as action labeling by introducing the background label $\emptyset$ to indicate frames without actions of interest. One practical challenge of this task is that the videos tend to be dominated by $\emptyset$. This requires the model to deal with unbalanced data and poses a different challenge than the temporal segmentation task.

Dataset and Metrics. We evaluate action detection of our model on the dataset of Bojanowski *et al.* [7], which consists of clips taken from the 69 movies Hollywood2 [9] dataset were extracted. The full time-stamped annotation of 16 actions (e.g. "Open Door" and "Stand Up") are manually added. For metrics, we follow [7] and use mean average precision for evaluating action detection and average Jaccard measure for evaluating the action alignment.

Experimental Setup. We use the extracted features from Bojanowski *et al.* [7] for all the methods. All methods use the same random splitting of the dataset. As we follow the exact setup of [7] for evaluation, we would like to clarify that the semi-supervised here means *video-level* semi-supervised setting, where a fraction

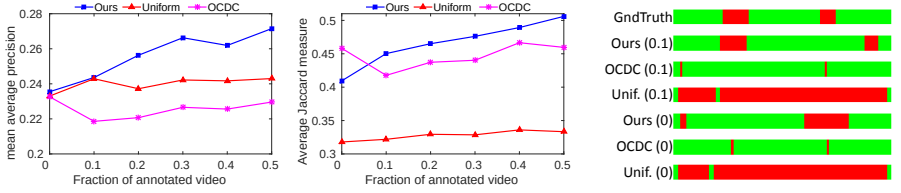

Fig. 9. Left plots mAP for action detection in unseen test videos. Middle plots the average Jaccard measure on the alignment evaluation set. Note that zero fraction of annotated video corresponds to the weakly-supervised setting, where all the videos in training set only have ordering supervision. Our approach consistently outperforms both baselines because of our temporal modeling and efficient evaluation of all possible alignments. On the right, we illustrate qualitative alignment results for all methods.

Fig. 10. Our weakly-supervised action detection results. Color means the output probability of the target action. Our model accurately localizes actions of varied lengths.

of the videos in the *supervised* set has full supervision, while the rest only has ordering as supervision. In this sense, the 0 fraction corresponds exactly to our weakly supervised setting, where all the videos only have ordering supervision. This is different from the *frame-level* semi-supervised setting we have discussed. All experiments are conducted over five random splits of the data.

Detection Results. The action detection results on the held-out testing set are shown in Figure 9 (left). While the occurring actions do not have a strong correlation, the results still demonstrate the importance of temporal modeling for better performance on held-out data. Both of our approaches outperform the OCDC baseline of Bojanowski *et al.* [7] in this scenario. Figure 10 shows the qualitative results of our weakly-supervised action detection model.

Alignment Results. The action alignment result on the evaluation set is shown in Figure 9 (middle). The uniform baseline performs the worst in this scenario, as there is no refinement of the alignment. On the other hand, our ECTC incorporates visual similarity and efficiently evaluates all possible alignments. This allows it to perform the best even for the alignment problem.

6 Conclusions

We have presented ECTC, a novel approach for learning temporal models of actions in a weakly supervised setting. The key technical novelty lies in the incorporation of visual similarity to explicitly capture dependencies between consecutive frames. We propose a dynamic programming based algorithm to efficiently evaluate all of the possible alignments and weight their importance by the consistency with the visual similarity. We further extend ECTC to incorporate frame-level semi-supervision in a unified framework, which significantly reduce the space of possible alignments. We verify the effectiveness of this framework with two applications: activity segmentation and action detection. We demonstrate that our model is able to outperform fully supervised baselines with only weak supervision, and our model achieves comparable results to state-of-the-art fully supervised models with less than 1% of supervision.

Acknowledgement. This work was supported by a grant from the Stanford AI Lab-Toyota Center for Artificial Intelligence Research.

References

1. Soomro, K., Roshan Zamir, A., Shah, M.: UCF101: A dataset of 101 human actions classes from videos in the wild. In: CRCV-TR-12-01. (2012)
2. Kuehne, H., Jhuang, H., Garrote, E., Poggio, T., Serre, T.: Hmdb: a large video database for human motion recognition. In: ICCV. (2011)
3. Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., Fei-Fei, L.: Large-scale video classification with convolutional neural networks. In: CVPR. (2014)
4. Gorban, A., Idrees, H., Jiang, Y.G., Roshan Zamir, A., Laptev, I., Shah, M., Sukthankar, R.: THUMOS challenge: Action recognition with a large number of classes. http://www.thumos.info/ (2015)
5. Heilbron, F.C., Escorcia, V., Ghanem, B., Niebles, J.C.: Activitynet: A large-scale video benchmark for human activity understanding. In: CVPR. (2015)
6. Bojanowski, P., Bach, F., Laptev, I., Ponce, J., Schmid, C., Sivic, J.: Finding actors and actions in movies. In: Proceedings of the IEEE International Conference on Computer Vision. (2013) 2280–2287
7. Bojanowski, P., Lajugie, R., Bach, F., Laptev, I., Ponce, J., Schmid, C., Sivic, J.: Weakly supervised action labeling in videos under ordering constraints. In: ECCV. (2014)
8. Duchenne, O., Laptev, I., Sivic, J., Bach, F., Ponce, J.: Automatic annotation of human actions in video. In: ICCV. (2009)
9. Laptev, I., Marszałek, M., Schmid, C., Rozenfeld, B.: Learning realistic human actions from movies. In: CVPR. (2008)
10. Alayrac, J.B., Bojanowski, P., Agrawal, N., Sivic, J., Laptev, I., Lacoste-Julien, S.: Learning from narrated instruction videos. arXiv preprint arXiv:1506.09215 (2015)
11. Bojanowski, P., Lagugie, R., Grave, E., Bach, F., Laptev, I., Ponce, J., Schmid, C.: Weakly-supervised alignment of video with text. In: ICCV. (2015)
12. Yu, S.I., Jiang, L., Hauptmann, A.: Instructional videos for unsupervised harvesting and learning of action examples. In: ACM Multimedia. (2014)
13. Kuehne, H., Arslan, A., Serre, T.: The language of actions: Recovering the syntax and semantics of goal-directed human activities. In: CVPR. (2014)
14. Lillo, I., Soto, A., Niebles, J.C.: Discriminative hierarchical modeling of spatio-temporally composable human activities. In: CVPR. (2014)
15. Messing, R., Pal, C., Kautz, H.: Activity recognition using the velocity histories of tracked keypoints. In: CVPR. (2009)
16. Rohrbach, M., Amin, S., Andriluka, M., Schiele, B.: A database for fine grained activity detection of cooking activities. In: CVPR. (2012)
17. Yeung, S., Russakovsky, O., Jin, N., Andriluka, M., Mori, G., Fei-Fei, L.: Every moment counts: Dense detailed labeling of actions in complex videos. arXiv preprint arXiv:1507.05738 (2015)
18. Graves, A., Fernández, S., Gomez, F., Schmidhuber, J.: Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In: ICML. (2006)
19. Das, P., Xu, C., Doell, R., Corso, J.: A thousand frames in just a few words: Lingual description of videos through latent topics and sparse object stitching. In: CVPR. (2013)
20. Gkioxari, G., Malik, J.: Finding action tubes. In: CVPR. (2015)
21. Pirsiavash, H., Ramanan, D.: Parsing videos of actions with segmental grammars. In: CVPR. (2014)

22. Sener, O., Zamir, A., Savarese, S., Saxena, A.: Unsupervised semantic parsing of video collections. In: ICCV. (2015)
23. Lan, T., Zhu, Y., Zamir, A.R., Savarese, S.: Action recognition by hierarchical mid-level action elements. In: ICCV. (2015)
24. Vo, N.N., Bobick, A.F.: From stochastic grammar to bayes network: Probabilistic parsing of complex activity. In: CVPR. (2014)
25. Yeung, S., Russakovsky, O., Mori, G., Fei-Fei, L.: End-to-end learning of action detection from frame glimpses in videos. arXiv (2015)
26. Donahue, J., Hendricks, L.A., Guadarrama, S., Rohrbach, M., Venugopalan, S., Saenko, K., Darrell, T.: Long-term recurrent convolutional networks for visual recognition and description. arXiv preprint arXiv:1411.4389 (2014)
27. Simonyan, K., Zisserman, A.: Two-stream convolutional networks for action recognition in videos. In: NIPS. (2014)
28. Xu, Z., Zhu, L., Yang, Y., Hauptmann, A.G.: Uts-cmu at THUMOS. CVPR THUMOS Challenge (2015)
29. Malmaud, J., Huang, J., Rathod, V., Johnston, N., Rabinovich, A., Murphy, K.: What's cookin'? interpreting cooking videos using text, speech and vision. NAACL (2015)
30. Ramanathan, V., Joulin, A., Liang, P., Fei-Fei, L.: Linking people in videos with their names using coreference resolution. In: ECCV. (2014)
31. Zhu, Y., Kiros, R., Zemel, R., Salakhutdinov, R., Urtasun, R., Torralba, A., Fidler, S.: Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In: ICCV. (2015)
32. Fernando, B., Gavves, E., Oramas, J.M., Ghodrati, A., Tuytelaars, T.: Modeling video evolution for action recognition. In: CVPR. (2015)
33. Niebles, J.C., Chen, C.W., Fei-Fei, L.: Modeling temporal structure of decomposable motion segments for activity classification. In: ECCV. (2010)
34. Tang, K., Fei-Fei, L., Koller, D.: Learning latent temporal structure for complex event detection. In: CVPR. (2012)
35. Taylor, G.W., Fergus, R., LeCun, Y., Bregler, C.: Convolutional learning of spatio-temporal features. In: ECCV. (2010)
36. Ramanathan, V., Tang, K., Mori, G., Fei-Fei, L.: Learning temporal embeddings for complex video analysis. In: ICCV. (2015)
37. Ryoo, M.S., Aggarwal, J.K.: Spatio-temporal relationship match: Video structure comparison for recognition of complex human activities. In: ICCV. (2009)
38. Song, Y., Morency, L.P., Davis, R.: Action recognition by hierarchical sequence summarization. In: CVPR. (2013)
39. Niebles, J.C., Wang, H., Fei-Fei, L.: Unsupervised learning of human action categories using spatial-temporal words. International journal of computer vision **79**(3) (2008) 299–318
40. Wu, C., Zhang, J., Savarese, S., Saxena, A.: Watch-n-patch: Unsupervised understanding of actions and relations. In: CVPR. (2015)
41. Chen, X., Zitnick, C.L.: Minds eye: A recurrent visual representation for image caption generation. In: CVPR. (2015)
42. Karpathy, A., Fei-Fei, L.: Deep visual-semantic alignments for generating image descriptions. In: CVPR. (2015)
43. Venugopalan, S., Rohrbach, M., Donahue, J., Mooney, R., Darrell, T., Saenko, K.: Sequence to sequence–video to text. In: ICCV. (2015)
44. Yao, L., Torabi, A., Cho, K., Ballas, N., Pal, C., Larochelle, H., Courville, A.: Video description generation incorporating spatio-temporal features and a soft-attention mechanism. In: ICCV. (2015)

45. Ng, J.Y.H., Hausknecht, M., Vijayanarasimhan, S., Vinyals, O., Monga, R., Toderici, G.: Beyond short snippets: Deep networks for video classification. arXiv preprint arXiv:1503.08909 (2015)
46. Graves, A., Jaitly, N.: Towards end-to-end speech recognition with recurrent neural networks. In: ICML. (2014)
47. Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Learning representations by back-propagating errors. Cognitive modeling **5** (1988) 3
48. Graves, A., Schmidhuber, J.: Framewise phoneme classification with bidirectional lstm and other neural network architectures. Neural Networks **18**(5) (2005) 602–610
49. Tieleman, T., Hinton, G.: Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. COURSERA: Neural Networks for Machine Learning (2012)
50. Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., Susstrunk, S.: Slic superpixels compared to state-of-the-art superpixel methods. Pattern Analysis and Machine Intelligence, IEEE Transactions on **34**(11) (2012) 2274–2282
51. Kuehne, H., Gall, J., Serre, T.: An end-to-end generative framework for video segmentation and recognition. In: WACV. (2016)
52. Wang, H., Schmid, C.: Action recognition with improved trajectories. In: ICCV. (2013)
53. Chang, C.C., Lin, C.J.: LIBSVM: A library for support vector machines. ACM Transactions on Intelligent Systems and Technology **2** (2011) 27:1–27:27 Software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm.