Learning Discriminative Prototypes with Dynamic Time Warping

Xiaobin Chang[1,2], Frederick Tung[2], Greg Mori[1,2]
Simon Fraser University[1], Borealis AI[2]
xiaobin_chang@sfu.ca frederick.tung@borealisai.com mori@cs.sfu.ca

Abstract

Dynamic Time Warping (DTW) is widely used for temporal data processing. However, existing methods can neither learn the discriminative prototypes of different classes nor exploit such prototypes for further analysis. We propose Discriminative Prototype DTW (DP-DTW), a novel method to learn class-specific discriminative prototypes for temporal recognition tasks. DP-DTW shows superior performance compared to conventional DTWs on time series classification benchmarks[1]*. Combined with end-to-end deep learning, DP-DTW can handle challenging weakly supervised action segmentation problems and achieves state of the art results on standard benchmarks. Moreover, detailed reasoning on the input video is enabled by the learned action prototypes. Specifically, an action-based video summarization can be obtained by aligning the input sequence with action prototypes.*

1. Introduction

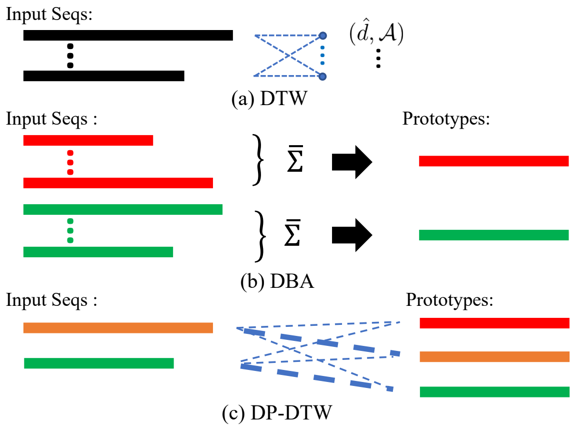
Temporal data is a common data form and widely exists in different domains [15], e.g., finance, industrial processes and video sequences. Analyzing temporal sequences is thus an important task. However, a significant challenge arises when comparing two sequences as they are not guaranteed to be aligned. They can be varied in temporal length and/or observation speed. Therefore, alignment is essential before comparisons, e.g., computing their discrepancy value. Naive pre-processing such as interpolation, cyclic repeat extension, place-holder insertion and down-sampling are used to align sequences with different lengths. These methods either modify the original data distribution or suffer from data loss and fail to handle the issue of varied speeds.

Dynamic Time Warping (DTW) [30, 2] was proposed to handle misalignment issues in temporal data. The optimal monotonic alignment between two input sequences is provided by a dynamic programming procedure. DTW is thus robust to inputs with varied temporal lengths and observation speeds. The discrepancy value between the two sequences can then be computed based on the alignment. With DTW and its discrepancy, a prototype over a set of sequences can be obtained by averaging. This technique is known as DTW barycenter averaging (DBA) [27] and enables several tasks, e.g., clustering and classification. However, DBA considers the intra-class samples only and neglects the inter-class ones in learning the class-specific prototypes for time series classification (TSC). Discriminative prototypes thus fail to be obtained and classification performance is negatively affected.

Besides the conventional multi-class single label TSC setting [9], we focus on solving the weakly supervised action segmentation problem in video data [21, 3]. Three major challenges are highlighted. First, the video data is captured from realistic scenarios, such as daily activities and movies. It thus contains sophisticated spatial-temporal dynamics. Secondly, multiple actions are performed sequentially in each video. Last but not least, instead of labelling the action at each frame, only the action order is provided as weak supervision. To better handle the complex temporal structure of video inputs, deep models are widely adopted to extract frame-wise deep feature representations. However, training deep models with such weak supervision is not straightforward. Existing methods [29, 12, 6, 22] follow very similar paradigms. Specifically, deep models first provide the action predictions of each frame. Different algorithms are then proposed to encode the frame-wise predictions with the given action transcript and result in different learning objectives. For example, the dynamic programming procedure of DTW is exploited by D^{3}TW [6] as its encoding algorithm. To obtain a differentiable DTW loss for deep learning, a relaxation technique, as in Soft-DTW [8] can be adopted. No existing work attempts to learn discriminative prototype sequences of different actions nor use them for action segmentation.

In this paper, we propose a novel DTW method, Discriminative Prototype DTW (DP-DTW), for temporal recognition problems. In the TSC setting, each sequence corresponds to a single class. Instead of averaging the sequences

[1] Code available at https://github.com/BorealisAI/TSC-Disc-Proto



(a) DTW

(b) DBA

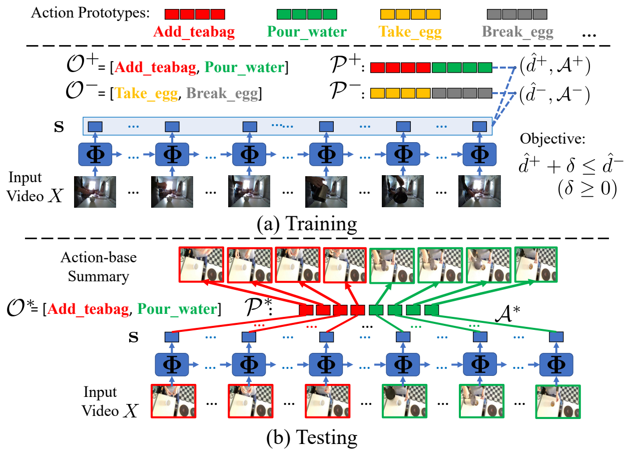
(c) DP-DTW

Figure 1. Bold lines represent sequences with varied temporal lengths. (a) DTW computes the discrepancy $\hat{d}$ and alignment $\mathcal{A}$ between a pair of sequences. (b) DBA computes a prototype by averaging (denoted as $\bar{\Sigma}$) the samples within a class. Different classes are indicated by colors. (c) DP-DTW focuses on the inter-class variance. Each input should be closest (shown as bold dashed line) to the prototype sequence of the same class (color).

within a class as in DBA [27], DP-DTW computes the discrepancies between an input and the prototypes of different classes and then is supervised by discriminative loss. Class-specific distinctive temporal dynamics are thus represented by such learned prototypes. Illustrations of DTW, DBA and the proposed DP-DTW are shown in Figure 1.

More importantly, DP-DTW can handle temporal recognition of a sequence of multiple classes, as in the weakly supervised action segmentation problem. Specifically, each action is a class and has its prototype sequence in DP-DTW. An input video can contain multiple actions performed one after another. Only the action ordering is recorded in the transcript as weak supervision. By concatenating the action prototypes in order, each transcript has its ordering sequence representation in DP-DTW. During training, discriminative losses are applied on the DTW discrepancies between the deep representation of the input video and the ordering sequences. As a result, the discriminative action prototypes are learned. With the retrieved or given transcript of a testing video, the action segmentation is obtained based on the DTW alignment between the input and ordering sequences. Each frame is assigned to the action (prototype) it aligns with. As a by-product, action-based key frames can be selected by the learned prototypes and used as a summarization of the input video. The process of DP-DTW mentioned above is illustrated in Figure 2.

The contributions of the proposed method are three-fold. (1) DP-DTW learns discriminative class-specific prototypes for TSC. (2) By modeling each action with a temporal sequence as a prototype, the training and inference of DP-DTW for weakly supervised action segmentation are unified under DTW. With the distinctive action dynamics captured by the learned prototypes, action segmentation can then benefit from an optimal temporal alignment. (3) Action-based video summarization is obtained as a detailed analysis and by-product of the discriminative prototypes learned by DP-DTW. DP-DTW is evaluated on different temporal recognition tasks. On the TSC benchmarks [9], DP-DTW outperforms the competitive DTW baselines. The effectiveness of DP-DTW on weakly supervised action segmentation is demonstrated by state of the art results on two challenging datasets [21, 3]. Detailed analysis, i.e., action-based summarization, on such videos is enabled by DP-DTW.


Action Prototypes:
Add_teabag
Pour_water
Take_egg
Break_egg
...
$\mathcal{O}^+$= [Add_teabag, Pour_water]
$\mathcal{P}^+$:
$(\hat{d}^+, \mathcal{A}^+)$
$\mathcal{O}^-$= [Take_egg, Break_egg]
$\mathcal{P}^-$:
$(\hat{d}^-, \mathcal{A}^-)$
s
Φ
Input Video X
Objective:
$\hat{d}^+ + \delta \leq \hat{d}^-$
$(\delta \geq 0)$
Action-base Summary
$\mathcal{O}^*$= [Add_teabag, Pour_water]
$\mathcal{P}^*$
$\mathcal{A}^*$
s
Φ
Input Video X

(a) Training

(b) Testing

Figure 2. DP-DTW for weakly supervised action segmentation. Each action, indicated by a color, is represented by a prototype sequence with temporal length 4. The frame-wise deep representation $\mathbf{s}$ of input video X is extracted by Φ, e.g., GRU. An action transcript $\mathcal{O}$ has its ordering sequence $\mathcal{P}$. Evaluated by the DTW discrepancies, a training sample should be closer to its ground truth $\mathcal{O}^+$ than a negative transcript $\mathcal{O}^-$. Hinge loss is used as the discriminative objective. The testing transcript $\mathcal{O}^*$ is retrieved or given with its sequence $\mathcal{P}^*$. Based on the DTW alignment $\mathcal{A}^*$ between $\mathbf{s}$ and $\mathcal{P}^*$, an action segmentation, indicated by the colored box on the frame, is obtained. Moreover, the action-based key frames are selected as a video summary.

2. Related Work

Dynamic Time Warping (DTW). DTW [30, 2] computes the discrepancy value between two sequences based on their optimal alignment from dynamic programming. Different DTW variants have been proposed. By relaxing the global alignment constraint in DTW, a local optimal matching algorithm [31] can be obtained. Shapelet methods [35, 16, 25] aim to capture local discriminative temporal dynamics by learning from pre-segmented sub-sequences. With a prediction ensemble [1, 23] from DTW and other domains such as frequency, time series classification (TSC) performance can

	Prototype		mini-batch	Weak-Sup.
	Cls-Spec.	Discri.	SGD	Act. Seg.
DBA [27]	✓	×	×	×
Soft-DTW [8]	✓	×	✓	×
DTWNet [5]	×	✓	✓	×
D^{3}TW† [6]	×	×	✓	✓
DP-DTW	✓	✓	✓	✓

Table 1. Comparisons of different DTW models. Temporal classification is the default task. 'Cls-Spec.' and 'Discri.' stand for 'Class-Specific' and 'Discriminative' correspondingly. 'Weak-Sup. Act. Seg.' means weakly supervised action segmentation and † means D^{3}TW is specified for this task only.

be further boosted. DTW also has been adopted for different applications, such as heterogeneous sequence alignment [13, 32], time series forecasting [34] and temporal pattern transform [24].

Learning Prototypes with DTW. Prototypes capture global temporal patterns over whole input sequences. With all training samples as class-specific prototypes, the one nearest-neighbour (1-NN) classifier with DTW discrepancy as distance can achieve competitive TSC results [11]. However, such models are not efficient. DBA [27] learns a few class-specific prototypes by averaging over the sequences of each class and iterative refinement. Its variant, Soft-DTW [8], smooths the dynamic programming procedure [26] and results in a differentiable loss optimized with mini-batch SGD. However, without considering inter-class variance, the prototypes learned by DBA and Soft-DTW are not discriminative. Beyond TSC tasks, prototypes can also be found in robust data or mid-level feature extractors, i.e. a DTW-layer, as proposed in DTWNet [5, 18]. The learned prototypes in a DTW-layer can be discriminative but latent, i.e., with no explicit correspondence to a specific class. None of these methods learns class-specific discriminative prototypes as in the proposed DP-DTW. Comparisons of different prototype learning methods are shown in Table 1.

Weakly Supervised Action Segmentation. Instead of labeling the action in every single frame, this setting only provides the action ordering of each video for training. Existing methods for this challenging task follow similar paradigms, encoding frame-wise action predictions with the action ordering to construct different objectives. Relevant models include D^{3}TW [6], where Soft-DTW [8] is adopted for encoding and loss computation. However, no prototype sequence is learned by D^{3}TW. An extended connectionist temporal classification model is proposed in [17], with frame-to-frame visual similarity as a regularization for frame labels. Soft boundary assignment on the initial frame predictions and iterative optimization are done in [12]. The Viterbi algorithm is another encoding option [29]. To combine a hidden markov model with deep networks, a loss is proposed by discriminating the energy of all possible frame labelings [22]. The proposed DP-DTW is different from existing methods in two aspects. Action prototype sequences are learned by our method and without frame-wise action prediction. As a by-product of the learned prototypes, action-based video summarizations can be obtained.

Video Summarization. To identify the key frames in a video as a summarization, existing methods assess diversity and/or representativeness and pick distinctive frames. Unsupervised learning methods [33, 37, 36] use selection criteria from intrinsic temporal structures. With explicit annotation of key frames, supervised learning algorithms [19, 14] have been developed. Under the weakly supervised setting, the category label of each video is provided and used as privileged information to improve summarization [7, 4]. DP-DTW can obtain a type of implicit summarization as a by-product of its alignment process.

3. Methodology

We develop a method for learning discriminative prototypes. We start by providing a recap of the Dynamic Time Warping (DTW) mechanism. Subsequently, the details of our DP-DTW are described.

3.1. Preliminaries

DTW Mechanisms. DTW can be treated as a function that takes in two temporal sequences and returns their optimal temporal alignment and the corresponding discrepancy. An input sequence $\mathbf{s} \in \mathbb{R}^{m\times\tau}$ has feature dimension m and temporal length τ. $\mathbf{s}[t] \in \mathbb{R}^m$ indicates the feature at time step t. Given two sequences $\mathbf{s}_1 \in \mathbb{R}^{m\times\tau_1}$ and $\mathbf{s}_2 \in \mathbb{R}^{m\times\tau_2}$, DTW can be expressed as,

$$\mathcal{A}, \hat{d} = \mathrm{DTW}(\mathbf{s}_1, \mathbf{s}_2), \tag{1}$$

where $\mathcal{A}$ represents the temporal alignment between $(\mathbf{s}_1, \mathbf{s}_2)$ and $\hat{d}$ is the discrepancy value. The temporal alignment $\mathcal{A}$ is an optimal solution from dynamic programming and obeys the DTW warping constraints [2], e.g., monotonicity and continuity. $\mathcal{A}$ is a list with length $max(\tau_1, \tau_2) \leq |\mathcal{A}| \leq \tau_1 + \tau_2 - 1$,

$$\mathcal{A} = \{a_1, ..., a_{|\mathcal{A}|}\}, \tag{2}$$

where $a_i = (t_{1,i}, t_{2,i}), i \in \{1, ..., |\mathcal{A}|\}$ indicates the ith alignment between $\mathbf{s}_1[t_{1,i}]$ and $\mathbf{s}_2[t_{2,i}]$. The DTW discrepancy value $\hat{d}$ accumulates along the aligned distances,

$$\hat{d} = \sum_{i=1}^{|\mathcal{A}|} ||\mathbf{s}_1[t_{1,i}] - \mathbf{s}_2[t_{2,i}]||_2. \tag{3}$$

A toy example of DTW, with $m = 1$, $\tau_1 = 10$, $\tau_2 = 10$, $|\mathcal{A}| = 13$, is illustrated in Figure 3.

Notation in DP-DTW. Based on DTW, the proposed DP-DTW aims to learn class-specific discriminative prototypes for temporal recognition tasks. We assume $\mathcal{K}$ classes are

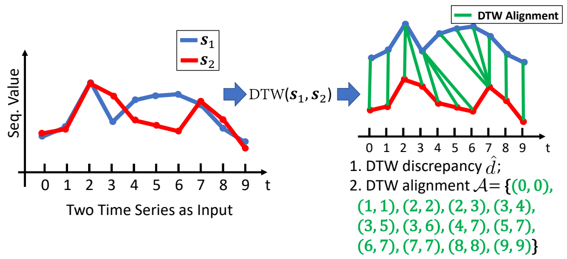

s_1
s_2
Seq. Value
0 1 2 3 4 5 6 7 8 9 t
Two Time Series as Input
DTW(s_1, s_2)
DTW Alignment
0 1 2 3 4 5 6 7 8 9 t
1. DTW discrepancy $\hat{d}$;
2. DTW alignment $\mathcal{A}$= {(0, 0), (1, 1), (2, 2), (2, 3), (3, 4), (3, 5), (3, 6), (4, 7), (5, 7), (6, 7), (7, 7), (8, 8), (9, 9)}

Figure 3. An illustration of DTW inputs and outputs. For example, the ninth alignment $a_9 = (5, 7)$ indicates the aligned moments of $\mathbf{s}_1[5]$ and $\mathbf{s}_2[7]$.

defined in the problem and a set of input sequences with $\mathcal{N}$ samples is denoted as $\mathcal{S} = \{\mathbf{s}_1, \ldots, \mathbf{s}_\mathcal{N}\}$. An input sample $\mathbf{s}_n \in \mathbb{R}^{m\times\tau_n}$, $n \in \{1, ..., \mathcal{N}\}$, has feature dimension m and temporal length τ_n (temporal lengths across inputs can be different). DP-DTW learns one prototype for each class[2]. For a specific class $k \in \{1, ..., \mathcal{K}\}$, its corresponding action prototype is $\mathbf{p}^k \in \mathbb{R}^{m\times\tau_p}$, with temporal length fixed at τ_p across different prototypes. The feature of a temporal sequence at time step t is indexed by $\mathbf{s}_n[t]$ or $\mathbf{p}^k[t]$. Moreover, DP-DTW is optimised with mini-batch SGD. To simplify the derivation, one input sequence $\mathbf{s}_n$ is considered. Generalization to other batch sizes is straightforward.

3.2. DP-DTW for TSC

To capture the distinctive temporal dynamics of different classes, DP-DTW focuses on optimizing the inter-class distance between an input sequence and different class-specific prototype sequences. In the time series classification (TSC) setting, an input sequence $\mathbf{s}_n$ comes from a single class and is labelled with $y_n \in \{1, ..., \mathcal{K}\}$. For each input $\mathbf{s}_n$, DP-DTW computes its DTW outputs with each learned class-specific prototype $\mathbf{p}^k$, $k \in \{1, ..., \mathcal{K}\}$,

$$\mathcal{A}^k_n, \hat{d}^k_n = \mathrm{DTW}(\mathbf{p}^k, \mathbf{s}_n). \tag{4}$$

In order to use these for classification, a softmax function is applied on the *negative* discrepancy values of all different classes for the logits σ^k_n, $k \in \{1, ..., \mathcal{K}\}$.

For learning the prototypes, we include two objectives. First, a cross entropy loss $\mathcal{L}_{CE} = -\log(\sigma^{y_n}_n)$ is used, for the purpose of enlarging inter-class distances from prototypes, leading to correct classification. Moreover, we wish to ensure each prototype represents a class well. Hence, the discrepancy values $\hat{d}^{y_n}_n$ form another loss $\mathcal{L}_D$. Therefore, the overall loss function $\mathcal{L}_{TSC}$ consists of two parts,

$$\begin{aligned} \mathcal{L}_{TSC} &= -\log(\sigma^{y_n}_n) + \lambda \cdot \hat{d}^{y_n}_n \\ &= \mathcal{L}_{CE} + \lambda \cdot \mathcal{L}_D, \end{aligned} \tag{5}$$

with a balancing hyper-parameter $\lambda \geq 0$.

Optimization. DP-DTW aims to learn a discriminative prototype for each class via the objective:

$$\min_{\{\mathbf{p}^1,\ldots,\mathbf{p}^\mathcal{K}\}} \mathcal{L}_{TSC}, \tag{6}$$

$\mathcal{L}_{TSC}$ is differentiable and optimized with mini-batch SGD.

Inference. Once the prototypes $\mathbf{p}^k, k \in \{1, .., \mathcal{K}\}$ are learned against Eq. 6, the predicted class label $\tilde{y}$ of a testing input $\mathbf{s}$ is obtained by using the 1 nearest neighbour (1-NN) classifier with DTW discrepancy,

$$\tilde{y} = \operatorname*{arg\,min}_{k\in\{1,\ldots,\mathcal{K}\}} \hat{d}^k. \tag{7}$$

With one discriminative prototype per class learned by DP-DTW, its inference is much more efficient than the baseline 1-NN classifier in which all training samples must be matched to a test sample.

3.3. DP-DTW for Weakly Supervised Action Segmentation

In weakly supervised action segmentation, only the action ordering, not frame-level labeling, is provided. By modeling each action as a prototype sequence, the action ordering can also be represented as a sequence in DP-DTW. Based on the discrepancy and temporal alignment between the video and ordering sequences, the training and inference of DP-DTW are unified under DTW. Specifically, a discriminative objective is built on the DTW discrepancy for the learning of action prototypes. The action segmentation can then be inferred and benefit from the optimal alignment by DTW. Moreover, with the learned action prototypes, action-based video summarization can be obtained by DP-DTW as a by-product.

To handle a video input, a deep neural network (DNN) can be used to extract its feature sequence. Let X_n denote the raw frames or pre-computed spatial-temporal features of the n-th video with τ_n as its temporal length. A DNN $\Phi(\cdot; \theta)$ is exploited to extract a frame-wise deep representation $\mathbf{s}_n$ from X_n,

$$\mathbf{s}_n = \Phi(X_n; \theta), \tag{8}$$

where θ denotes the DNN model parameters. A deep representation $\mathbf{s}_n[t]$ corresponds to the exact moment of input $X_n[t]$, $t \in \{1, ..., \tau_n\}$. Moreover, an action ordering can be represented by its corresponding sequence in DP-DTW. Denoting the transcript of the n-th sample as $\mathcal{O}_n = [o_{n,1}, ..., o_{n,l_n}]$ with $l_n = |\mathcal{O}_n|$ actions recorded. Each $o_{n,i}$, $i \in \{1, ..., l_n\}$, specifies the i-th action appearing in the video. Assuming the i-th action is $k \in \{1, ..., \mathcal{K}\}$, $o_{n,i} = k$, and it thus corresponds to the action prototype $\mathbf{p}^k$. Therefore, the ordering sequence $\mathcal{P}_n$ is generated by concatenating the action prototypes $\{\mathbf{p}^1, ..., \mathbf{p}^\mathcal{K}\}$ according

[2] This is for simplicity of presentation. DP-DTW can also support multiple prototypes for each class.

to the action ordering recorded in the transcript $\mathcal{O}_n$. This procedure is denoted as Π,

$$\begin{aligned}\mathcal{P}_n &= \Pi(\mathcal{O}_n; \{\mathbf{p}^1, ..., \mathbf{p}^{\mathcal{K}}\})\\ &= \mathrm{TempCat}([\mathbf{p}^{o_{n,1}}, ..., \mathbf{p}^{o_{n,l_n}}]),\end{aligned} \tag{9}$$

where $\mathrm{TempCat}(\cdot)$ concatenates the list of action prototypes on their temporal dimension and returns the sequence $\mathcal{P}_n \in \mathbb{R}^{m\times\Gamma}$ with temporal length $\Gamma = l_n \cdot \tau_p$.

Training. To learn action prototypes with ordering only as weak supervision, both the positive and negative action transcripts are required in building the discriminative learning objective. For the n-th training sample, its ground-truth action ordering is recorded by its positive transcript $\mathcal{O}_n^+$ while all other orderings different from the ground-truth can be in its negative transcripts. However, it is not feasible to consider all possible negative ones, and most of them are not meaningful or seldom appear. Therefore, a reference ordering set $\mathcal{R}$ is constructed by aggregating all positive transcripts from the training split and keeping the unique orderings. The feasible negative set of the n-th sample is $\mathcal{R} \setminus \mathcal{O}_n^+$. Instead of considering all of them, Q different negative transcripts, $\{\mathcal{O}_n^{-,1}, ..., \mathcal{O}_n^{-,Q}\} \sim \mathcal{R} \setminus \mathcal{O}_n^+$, are randomly selected at each training step for efficiency. Ordering sequences $\mathcal{P}_n^+$ and $\mathcal{P}_n^{-,q}$, $q \in \{1, ..., Q\}$, of the positive and negative transcripts are from Eq. 9. The DTW outputs between $\mathbf{s}_n$ and $\mathcal{P}_n^+$ or different $\mathcal{P}_n^{-,q}$s are then computed,

$$\begin{cases}\mathcal{A}_n^+, \hat{d}_n^+ &= \mathrm{DTW}(\mathcal{P}_n^+, \mathbf{s}_n);\\ \mathcal{A}_n^{-,q}, \hat{d}_n^{-,q} &= \mathrm{DTW}(\mathcal{P}_n^{-,q}, \mathbf{s}_n), q \in \{1, ..., Q\}.\end{cases} \tag{10}$$

As a discriminative model, the input sequence $\mathbf{s}_n$ should be closer to its positive ordering prototype than any negative one and formulated as,

$$\hat{d}_n^+ < \hat{d}_n^{-,q}, \forall q \in \{1, ..., Q\}. \tag{11}$$

A hinge loss with margin $\delta \geq 0$ is thus used as the discriminative loss for model training,

$$\mathcal{L}_h = \sum_{q=1}^{Q} \max(0, \hat{d}_n^+ - \hat{d}_n^{-,q} + \delta). \tag{12}$$

Moreover, with $\mathcal{P}_n^+$ as a prototype sequence, the $\hat{d}_n^+$ should be reduced to shrink the representation variance around it, which leads to a distance loss,

$$\mathcal{L}_D = \hat{d}_n^+. \tag{13}$$

The overall loss for weakly supervised action segmentation is denoted as $\mathcal{L}_{w\_seg}$,

$$\mathcal{L}_{w\_seg} = \mathcal{L}_h + \lambda \cdot \mathcal{L}_D, \tag{14}$$

with a balancing hyper-parameter $\lambda \geq 0$.

Optimization. The loss $\mathcal{L}_{w\_seg}$ is built on deep representation $\mathbf{s}_n$ and action prototypes $\{\mathbf{p}^1, ..., \mathbf{p}^{\mathcal{K}}\}$. $\mathbf{s}_n$ is extracted by deep model $\Phi(\cdot;\theta)$ from raw video input X_n. Therefore, the learning objective is defined as,

$$\min_{\{\mathbf{p}^1,...,\mathbf{p}^{\mathcal{K}}\};\theta} \mathcal{L}_{w\_seg}, \tag{15}$$

where the action prototypes and DNN parameters are jointly optimized with mini-batch SGD.

Inference. In the *segmentation* setting, only the testing raw input X_n is available. Its frame-wise deep representation $\mathbf{s}_n$ is extracted by the learned DNN Φ as in Eq. 8. Due to the exact temporal correspondence between $\mathbf{s}_n$ and X_n, the analysis, e.g., alignments, on $\mathbf{s}_n$ can be easily tracked back to the raw frames in X_n.

The best matching transcript $\mathcal{O}_n^*$ is retrieved from the reference set $\mathcal{R}$ by,

$$\mathcal{O}_n^* = \arg\min_{\mathcal{O}\in\mathcal{R}} \mathrm{DTW}(\Pi(\mathcal{O}; \{\mathbf{p}^1, ..., \mathbf{p}^{\mathcal{K}}\}), \mathbf{s}_n), \tag{16}$$

where Π is the concatenation procedure defined in Eq. 9 and $\arg\min$ compares DTW discrepancies only. $\mathcal{P}_n^*$ is the ordering sequence of $\mathcal{O}_n^*$. The action assignment on the input sequence $\mathbf{s}_n$ can then be obtained based on the optimal alignment $\mathcal{A}_n^*$ by DTW,

$$\mathcal{A}_n^*, \hat{d}_n^* = \mathrm{DTW}(\mathcal{P}_n^*, \mathbf{s}_n). \tag{17}$$

Specifically, $a_i = (t_{1,i}, t_{2,i})$ is the ith alignment of $\mathcal{A}_n^*$ (as in Eq. 2) and it indicates the aligned pair of $\mathcal{P}_n^*[t_{1,i}]$ and $\mathbf{s}_n[t_{2,i}]$. The action of $\mathbf{s}_n$ at time step $t_{2,i}$ is consistent with the action of $\mathcal{P}_n^*$ at $t_{1,i}$, which can be easily determined with $\mathcal{O}_n^*$ and the action prototype length τ_p. Moreover, a frame in $\mathbf{s}_n$ can align with multiple continuous steps in $\mathcal{P}_n^*$ and its nearest neighbour is chosen for action label assignment. In the *alignment* setting, the ground-truth transcript $\mathcal{O}_n^+$ is given. It can be handled with a similar procedure to that described above by replacing $\mathcal{O}_n^*$ with $\mathcal{O}_n^+$ accordingly.

3.3.1 Action-based Video Summarization

With the learned discriminative prototypes of different actions $\{\mathbf{p}^1, ..., \mathbf{p}^{\mathcal{K}}\}$, a summarization of the input video can be obtained by aggregating the key moments of each action according to the transcript. Specifically, the ground-truth action transcript $\mathcal{O}_n^+$ is provided along with the video input X_n. $\mathcal{P}_n^+$ is the ordering sequence of $\mathcal{O}_n^+$. $\mathcal{P}_n^+ \in \mathbb{R}^{m\times\Gamma}$ is with temporal length $\Gamma = l_n \cdot \tau_p$, where l_n is the number of actions appearing in the transcript $\mathcal{O}_n^+$ and τ_p is the temporal length of each action prototype. Based on the DTW alignment $\mathbf{A}_n^+$ between $\mathbf{s}_n$ and $\mathcal{P}_n^+$, each $\mathcal{P}_n^+[t]$, $t \in \{1, ..., \Gamma\}$, has a nearest neighbour from its aligned $\mathbf{s}_n$ and is denoted as $\mathbf{s}_n[t']$. t' is treated as one of the key or representative moments of the input sequence and there are

Γ key moments in total. In the ideal case, every τ_p key moments under the same action corresponds to the given transcript $\mathcal{O}_n^+$. Therefore, the selected Γ key frames from X_n are the action-based summarization of the video.

4. Experiments

The main characteristic of DP-DTW is computing class-specific discriminative prototypes for temporal recognition tasks. In this section, the effectiveness of the proposed model is verified on time series classification (TSC) and weakly supervised action segmentation. Moreover, with the prototypes learned by DP-DTW, we show that detailed analysis, i.e., summarizing the input videos with action-based key frames, can be achieved.

4.1. TSC

Dataset. **UCR** [9] is a benchmark collection of 128 univariate time series datasets with different application backgrounds such as electronics and biology. The sequence temporal length is the same within each dataset. Each dataset has multiple classes and each sequence belongs to one class.
Implementation Details. In each UCR dataset, the number of prototypes $\mathcal{K}$ in DP-DTW is set as the number of classes in the dataset, i.e. one prototype per class. The temporal length τ_p of each prototype is fixed as the input sequence length. The feature dimension m of both input and prototype sequences is 1 in the UCR datasets. DP-DTW is directly applied on the raw input sequences of each UCR dataset to learn the discriminative prototypes of classes. The prototypes are initialized with the medoids of different classes. 20% of the training data in each dataset forms a mini-batch and 60 epochs are used for learning. We optimise with Adam [20] and cross-validate the hyper-parameters such as learning rate.
Competitors. The proposed DP-DTW is compared with the four baselines: (1) *1-NN + ED* is the 1 nearest neighbour (1-NN) classifier with Euclidean distance (ED) for TSC. (2) *1-NN + DTW* is 1-NN with DTW discrepancy for TSC. (3) *1-NN + DTW(W)* is 1-NN with DTW discrepancy (by tuning its window size constraint to optimal) for TSC. (4) *DBA* [27] averages the time series within each class as prototype. The first three methods are the common baselines of UCR datasets with their results listed on the project page[3]. Moreover, the first three 1-NN based methods are strong baselines with all training samples as reference. On the contrary, the prototype learning methods, e.g., DP-DTW and DBA, are with one prototype learned for each class.
Results. To collectively compare classification performance of the five TSC methods over the 128 datasets, critical difference diagram [10] can be exploited, as shown in Figure 4. The averaged (over 128 datasets) ranks of different classifiers are compared. The better method is with the lower rank. DP-DTW is thus the best model among them with the lowest averaged rank ($\sim$1.6) achieved. Different 1-NN classifiers come after with higher ranks. Moreover, the differences among these methods are statistically significant as the gaps of their averaged ranks are all greater than the critical difference (CD) 0.539. For more targeted analysis, different methods can be compared in a pair. A model that achieves superior results to its competitor on the majority of datasets is the better one. The pairwise comparisons between our DP-DTW and the four competitors are as follows. DP-DTW achieves no worse results than 1-NN + ED on 86.7% $(111/128)$ of all datasets. Comparing with the competitive DTW based 1-NN classifiers, DP-DTW still has no worse classification performance on 82.8% $(106/128)$ and 74.2% $(95/128)$ of all datasets to 1-NN + DTW and 1-NN + DTW(W) respectively. A scatter plot is also used to show the details of comparison between DP-DTW and 1-NN + DTW, as in Figure 5. DP-DTW clearly outperforms the averaging based DBA [27] with no worse results on all 128 datasets. Both the critical difference diagram of ranking and pairwise comparisons demonstrate the importance of learning discriminative prototypes for different classes.

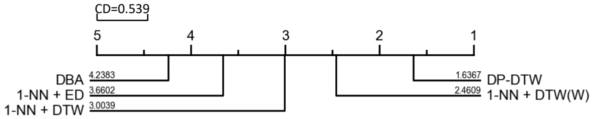


Figure 4. Critical difference diagram on the UCR 128 datasets with five TSC algorithms compared. The critical difference (CD) is 0.539 with significance level at 0.05.

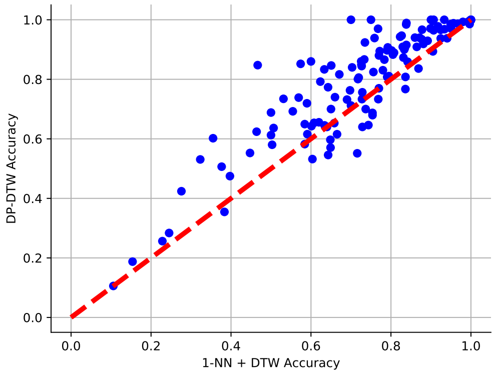


Figure 5. Each point corresponds to two TSC test accuracies on a UCR dataset by DP-DTW (y-axis) and 1-NN + DTW (x-axis). The points above or on the diagonal red dashed line mean DP-DTW achieves no worse results than 1-NN + DTW on such datasets. This is the case for 82.8% of the UCR datasets.

[3] https://www.cs.ucr.edu/~eamonn/time_series_data_2018/, we also reproduce them as a sanity check.

	Breakfast			Hollywood		
	F-acc.	IoU	IoD	F-acc.	IoU	IoD
HMM+RNN [28]	33.3	-	-	-	11.9	-
TCFPN [12]	38.4	24.2	40.6	28.7	12.6	18.3
NN-Viterbi [29]	43.0	-	-	-	-	-
D^{3}TW [6]	45.7	-	-	33.6	-	-
CDFL [22]	50.2	33.7	**45.4**	45.0	19.5	25.8
DP-DTW	**50.8**	**35.6**	45.1	**55.6**	**33.2**	**43.3**

Table 2. Comparisons among DP-DTW and competitors under the segmentation setting.

	Breakfast			Hollywood		
	F-acc.	IoU	IoD	F-acc.	IoU	IoD
HMM+RNN [28]	-	-	47.3	-	-	46.3
TCFPN [12]	53.5	35.3	52.3	57.4	22.3	39.6
NN-Viterbi [29]	-	-	-	-	-	48.7
D^{3}TW [6]	57.0	-	56.3	59.4	-	50.9
CDFL [22]	63.0	45.8	63.9	64.3	40.5	52.9
DP-DTW	**67.7**	**50.8**	**66.5**	**66.4**	**46.8**	**61.7**

Table 3. Comparisons among DP-DTW and competitors under the alignment setting.

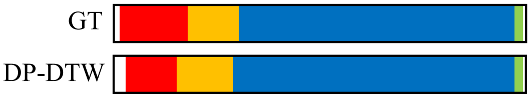


Figure 6. The illustrated test video is 'P39_cam02_P39_friedegg' from Breakfast dataset. The retrieved transcript is the same as ground-truth (GT). Different actions are represented by different colors, i.e., butter_pan, crack_egg, fry_egg, put_egg2plate, and white indicates background. Best viewed in color.

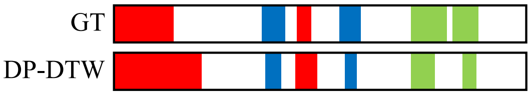


Figure 7. Illustration of video action alignment. The test video consists of multiple actions, i.e., Run, SitDown, StandUp and white as background. The ground truth action transcript is given. Best viewed in color.

4.2. Weakly Supervised Action Segmentation

Datasets. Breakfast [21] contains 1,712 videos of 48 actions related to the preparation of breakfast. Such videos are recorded from multiple views under real-life scenarios, e.g., from 18 different home kitchens. On average, $\sim$7 action instances are performed one after another in each video. We follow the data splits in [21] and the averaged results are reported. **Hollywood Extended** [3] comprises 16 action classes and 937 videos clipped from Hollywood movies. There are 2 to 11 actions in each video and 2.5 actions on average. For a fair comparison, the pre-computed frame-level features and the data split criteria in [28, 29] are used. In both datasets, the action ordering of each video is recorded in a transcript as weak supervision.

Settings and Metrics. *Segmentation* and *alignment* are two sub-tasks in weakly supervised action segmentation. In the segmentation task, the ground-truth action transcript is not available during evaluation. The best matching action ordering is first retrieved from the reference set $\mathcal{R}$. The label assignment procedure is then applied. In the alignment task, the ground-truth transcript is provided. Action labels are assigned following the ground-truth action order. Three standard metrics are adopted to evaluate both settings. The first is the frame accuracy (**F-acc.**), the percentage of frames that are correctly labeled. The other two are the intersection over union (**IoU**) and the intersection over detection (**IoD**). Given a ground-truth action assignment I^* and the predicted assignment I, IoU $= |I \cap I^*|/|I \cup I^*|$ and IoD $= |I \cap I^*|/|I|$.

Implementation Details. The deep model Φ is a single-layer GRU with 256 hidden units. Therefore, the feature dimension $m = 256$. $\mathcal{K} = 17$ (16 actions plus 1 background class) for the Hollywood dataset and $\mathcal{K} = 48$ (with background class included) for Breakfast. The temporal length τ_p of prototype is 6 and 8 for Hollywood and Breakfast respectively. The number of randomly selected negative transcripts Q (in Eq. 10) is 50 during training. The hinge loss margin $\delta = 1$ in Eq. 12. To minimize objective $\mathcal{L}_{w\_seg}$ in Eq. 14 with balancing $\lambda = 0.1$, an Adam optimizer is used with mini-batch size 64 and initial learning rate at 0.001 for $10,000$ training steps.

Competitors. The state of the art methods for weakly supervised action segmentation are compared to: (1) A new loss is proposed by *CDFL* [22] to discriminate the energy of all valid and invalid action assignment paths. (2) In *D^{3}TW* [6], the dynamic programming of DTW is exploited as an optimal encoding and a discriminative DTW loss is proposed. (3) *NN-Viterbi* [29] proposes a Viterbi-based loss that enables online learning. (4) *TCFPN* [12] is based on the frame-wise label prediction. Network updates are then performed iteratively for better efficiency. (5) In *HMM+RNN* [28], an HMM ensures the assignments obey the action order while an RNN makes the frame-wise predictions. These methods are mainly based on encoding frame predictions with an action ordering.

Segmentation Results. The proposed DP-DTW is compared with different competitors under the segmentation setting, as shown in Table 2. DP-DTW achieves one of the best results on the Breakfast dataset. On the Hollywood Extented, it clearly outperforms the state of the art, CDFL [22], with 10.6%, 13.7% and 17.5% improvements on frame accuracy, IoU and IoD respectively. Comparing with the DTW based method, D^{3}TW [6], DP-DTW also achieves superior performance to it with clear margins, 5.1% better frame accuracy on Breakfast and 22.0% better on Hollywood. These results demonstrate the effectiveness and ne-

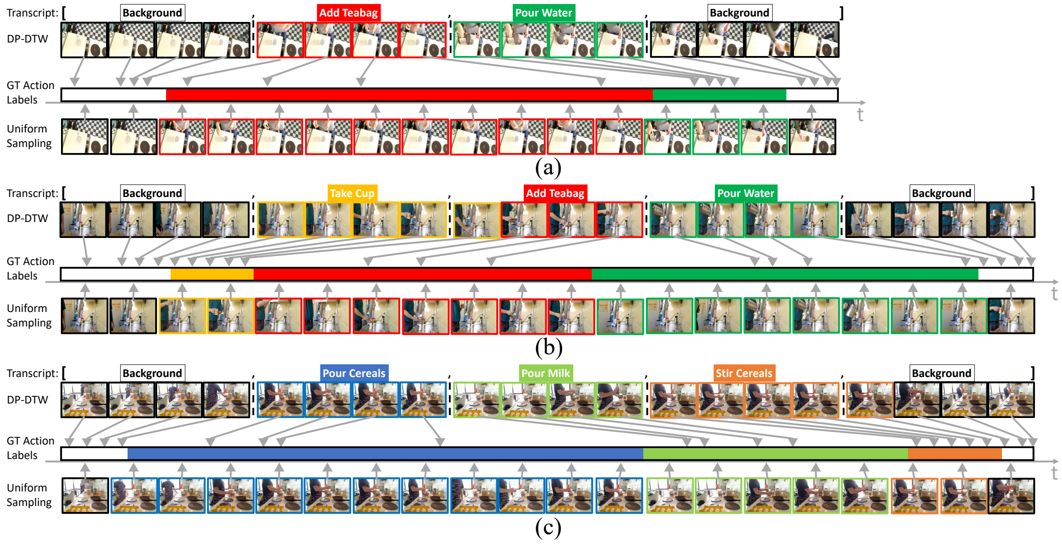


Figure 8. Summarizations of three videos by DP-DTW and uniform sampling. Different actions are indicated by different colors. The length of a color bar reflects the relative duration of the corresponding action in a video. Selected moments of the frames are also indicated. The action performed in a selected frame is the same as its selected moment and indicated by its frame box color. In DP-DTW summarizations, dashed lines split the summary frames according to the ideal action-based case where the actions of selected frames within each split should be consistent with the action in transcript.

cessity of explicitly learning discriminative prototypes of different actions for the weakly supervised action segmentation problems. The qualitative comparison between action segmentation by DP-DTW and ground-truth labels is illustrated in Figure 6. The majority of sequential actions with varied temporal lengths can be localized by DP-DTW. However, it is challenging to accurately determine the true start and end moments across actions.

Alignment Results. Comparing with the segmentation task, the alignment one is less challenging due to the ground-truth action transcripts being provided during testing. As a result, a model's performance under the alignment setting is generally much better than the segmentation one, as shown in Table 3. DP-DTW achieves the best results on both datasets. Comparing with D^{3}TW [6], DP-DTW still consistently achieves better results with clear ($\geq 7.0\%$) margin over all criteria. The action alignment of a video (*0652*) from Hollywood Extended dataset is illustrated in Figure 7. Good action alignments can be achieved by DP-DTW even with frequent action transitions.

4.3. Action-Based Video Summarization

With the action prototypes learned and the transcripts provided in weakly supervised action segmentation, the action-based key frames can be selected by DP-DTW as a by-product and used to summarize the input video, as detailed in Sec. 3.3.1. For illustration purposes, a DP-DTW model with prototype temporal length $\tau_p = 4$ is trained on Breakfast. Four key frames are thus selected by each prototype sequence as the summary of an action. In the ideal case, such frames should all belong to the corresponding action and an action-based video summarization can be obtained according to the action transcript. As shown in Figure 8, the key frames selected by a prototype are often distinctive moments across the time span of the corresponding action. The DP-DTW summarization is action-based and thus robust to action duration variation in the video. On the contrary, summarizing a video by uniform sampling can only reflect the duration of different actions. By comparing the action labels of the selected frames with the ideal action-based summarization case (inferred from the action transcript and τ_p), the matching rate (accuracy) can be obtained. The summarization by DP-DTW achieves 62.5% accuracy while uniform sampling achieves 40.8%.

5. Conclusion

We proposed Discriminative Prototype DTW (DP-DTW), the first model to explicitly learn class-specific discriminative prototypes for temporal recognition. Different from existing methods, DP-DTW focuses on enlarging the inter-class difference among prototypes via a discriminative objective. We develop an algorithm for weakly supervised segmentation based on DP-DTW prototype sequences, with end-to-end learning. DP-DTW outperforms competitive baselines on TSC benchmarks. On two challenging weakly supervised action segmentation datasets, DP-DTW achieves state of the art results. Action-based video summarization is also enabled by DP-DTW via alignment to the input video.

References

[1] Anthony Bagnall, Jason Lines, Jon Hills, and Aaron Bostrom. Time-series classification with cote: the collective of transformation-based ensembles. *IEEE Transactions on Knowledge and Data Engineering*, 27(9):2522–2535, 2015. 2

[2] Donald J Berndt and James Clifford. Using dynamic time warping to find patterns in time series. In *KDD workshop*, volume 10, pages 359–370. Seattle, WA, USA:, 1994. 1, 2, 3

[3] 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 *European Conference on Computer Vision*, pages 628–643. Springer, 2014. 1, 2, 7

[4] Sijia Cai, Wangmeng Zuo, Larry S Davis, and Lei Zhang. Weakly-supervised video summarization using variational encoder-decoder and web prior. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 184–200, 2018. 3

[5] Xingyu Cai, Tingyang Xu, Jinfeng Yi, Junzhou Huang, and Sanguthevar Rajasekaran. Dtwnet: a dynamic time warping network. In *Advances in Neural Information Processing Systems*, pages 11640–11650, 2019. 3

[6] Chien-Yi Chang, De-An Huang, Yanan Sui, Li Fei-Fei, and Juan Carlos Niebles. D3tw: Discriminative differentiable dynamic time warping for weakly supervised action alignment and segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3546–3555, 2019. 1, 3, 7, 8

[7] Yiyan Chen, Li Tao, Xueting Wang, and Toshihiko Yamasaki. Weakly supervised video summarization by hierarchical reinforcement learning. In *Proceedings of the ACM Multimedia Asia*, pages 1–6, 2019. 3

[8] Marco Cuturi and Mathieu Blondel. Soft-dtw: a differentiable loss function for time-series. *arXiv preprint arXiv:1703.01541*, 2017. 1, 3

[9] Hoang Anh Dau, Anthony Bagnall, Kaveh Kamgar, Chin-Chia Michael Yeh, Yan Zhu, Shaghayegh Gharghabi, Chotirat Annh Ratanamahatana, and Eamonn Keogh. The ucr time series archive. *IEEE/CAA Journal of Automatica Sinica*, 6(6):1293–1305, 2019. 1, 2, 6

[10] Janez Demšar. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine learning research*, 7(Jan):1–30, 2006. 6

[11] Hui Ding, Goce Trajcevski, Peter Scheuermann, Xiaoyue Wang, and Eamonn Keogh. Querying and mining of time series data: experimental comparison of representations and distance measures. *Proceedings of the VLDB Endowment*, 1(2):1542–1552, 2008. 3

[12] Li Ding and Chenliang Xu. Weakly-supervised action segmentation with iterative soft boundary assignment. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6508–6516, 2018. 1, 3, 7

[13] Mark Everingham, Josef Sivic, and Andrew Zisserman. Hello! my name is... buffy"–automatic naming of characters in tv video. In *BMVC*, volume 2, page 6, 2006. 3

[14] Jiri Fajtl, Hajar Sadeghi Sokeh, Vasileios Argyriou, Dorothy Monekosso, and Paolo Remagnino. Summarizing videos with attention. In *Asian Conference on Computer Vision*, pages 39–54. Springer, 2018. 3

[15] Tak-chung Fu. A review on time series data mining. *Engineering Applications of Artificial Intelligence*, 24(1):164–181, 2011. 1

[16] Josif Grabocka, Nicolas Schilling, Martin Wistuba, and Lars Schmidt-Thieme. Learning time-series shapelets. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 392–401, 2014. 2

[17] De-An Huang, Li Fei-Fei, and Juan Carlos Niebles. Connectionist temporal modeling for weakly supervised action labeling. In *European Conference on Computer Vision*, pages 137–153. Springer, 2016. 3

[18] Brian Kenji Iwana, Volkmar Frinken, and Seiichi Uchida. Dtw-nn: A novel neural network for time series recognition using dynamic alignment between inputs and weights. *Knowledge-Based Systems*, 188:104971, 2020. 3

[19] Yudong Jiang, Kaixu Cui, Bo Peng, and Changliang Xu. Comprehensive video understanding: Video summarization with content-based video recommender design. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 0–0, 2019. 3

[20] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[21] Hilde Kuehne, Ali Arslan, and Thomas Serre. The language of actions: Recovering the syntax and semantics of goal-directed human activities. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 780–787, 2014. 1, 2, 7

[22] Jun Li, Peng Lei, and Sinisa Todorovic. Weakly supervised energy-based learning for action segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 6243–6251, 2019. 1, 3, 7

[23] Jason Lines, Sarah Taylor, and Anthony Bagnall. Hive-cote: The hierarchical vote collective of transformation-based ensembles for time series classification. In *2016 IEEE 16th international conference on data mining (ICDM)*, pages 1041–1046. IEEE, 2016. 2

[24] Suhas Lohit, Qiao Wang, and Pavan Turaga. Temporal transformer networks: Joint learning of invariant and discriminative time warping. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 12426–12435, 2019. 3

[25] Qianli Ma, Wanqing Zhuang, Sen Li, Desen Huang, and Garrison W Cottrell. Adversarial dynamic shapelet networks. In *AAAI*, pages 5069–5076, 2020. 2

[26] Arthur Mensch and Mathieu Blondel. Differentiable dynamic programming for structured prediction and attention. *arXiv preprint arXiv:1802.03676*, 2018. 3

[27] François Petitjean, Germain Forestier, Geoffrey I Webb, Ann E Nicholson, Yanping Chen, and Eamonn Keogh. Dynamic time warping averaging of time series allows faster and more accurate classification. In *2014 IEEE international*

conference on data mining, pages 470–479. IEEE, 2014. 1, 2, 3, 6

[28] Alexander Richard, Hilde Kuehne, and Juergen Gall. Weakly supervised action learning with rnn based fine-to-coarse modeling. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 754–763, 2017. 7

[29] Alexander Richard, Hilde Kuehne, Ahsan Iqbal, and Juergen Gall. Neuralnetwork-viterbi: A framework for weakly supervised video learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7386–7395, 2018. 1, 3, 7

[30] Hiroaki Sakoe and Seibi Chiba. Dynamic programming algorithm optimization for spoken word recognition. *IEEE transactions on acoustics, speech, and signal processing*, 26(1):43–49, 1978. 1, 2

[31] Yasushi Sakurai, Christos Faloutsos, and Masashi Yamamuro. Stream monitoring under the time warping distance. In *2007 IEEE 23rd International Conference on Data Engineering*, pages 1046–1055. IEEE, 2007. 2

[32] Pramod Sankar, CV Jawahar, and Andrew Zisserman. Subtitle-free movie to script alignment. In *Proc. Brit. Mach. Vis. Conf*, pages 121–1, 2009. 3

[33] Yair Shemer, Daniel Rotman, and Nahum Shimkin. Ilssumm: Iterated local search for unsupervised video summarization. *arXiv preprint arXiv:1912.03650*, 2019. 3

[34] LE Vincent and Nicolas Thome. Shape and time distortion loss for training deep time series forecasting models. In *Advances in Neural Information Processing Systems*, pages 4189–4201, 2019. 3

[35] Lexiang Ye and Eamonn Keogh. Time series shapelets: a new primitive for data mining. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 947–956, 2009. 2

[36] Bin Zhao and Eric P Xing. Quasi real-time summarization for consumer videos. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2513–2520, 2014. 3

[37] Kaiyang Zhou, Yu Qiao, and Tao Xiang. Deep reinforcement learning for unsupervised video summarization with diversity-representativeness reward. *AAAI*, 2018. 3