Action Segmentation with Joint Self-Supervised Temporal Domain Adaptation

Min-Hung Chen[1*] Baopu Li[2] Yingze Bao[2] Ghassan AlRegib[1] Zsolt Kira[1]
[1]Georgia Institute of Technology [2]Baidu USA

Abstract

Despite the recent progress of fully-supervised action segmentation techniques, the performance is still not fully satisfactory. One main challenge is the problem of spatio-temporal variations (e.g. different people may perform the same activity in various ways). Therefore, we exploit unlabeled videos to address this problem by reformulating the action segmentation task as a cross-domain problem with domain discrepancy caused by spatio-temporal variations. To reduce the discrepancy, we propose ***Self-Supervised Temporal Domain Adaptation (SSTDA)****, which contains two self-supervised auxiliary tasks (binary and sequential domain prediction) to jointly align cross-domain feature spaces embedded with local and global temporal dynamics, achieving better performance than other Domain Adaptation (DA) approaches. On three challenging benchmark datasets (GTEA, 50Salads, and Breakfast), SSTDA outperforms the current state-of-the-art method by large margins (e.g. for the F1@25 score, from 59.6% to 69.1% on Breakfast, from 73.4% to 81.5% on 50Salads, and from 83.6% to 89.1% on GTEA), and requires only 65% of the labeled training data for comparable performance, demonstrating the usefulness of adapting to unlabeled target videos across variations. The source code is available at* https://github.com/cmhungsteve/SSTDA.

1. Introduction

The goal of action segmentation is to simultaneously segment videos by time and predict an action class for each segment, leading to various applications (e.g. human activity analyses). While action classification has shown great progress given the recent success of deep neural networks [38, 28, 27], temporally locating and recognizing action segments in long videos is still challenging. One main challenge is the problem of *spatio-temporal variations* of human actions across videos [16]. For example, different people may *make tea* in different personalized styles even if the given recipe is the same. The intra-class variations cause degraded performance by directly deploying a model trained with different groups of people.

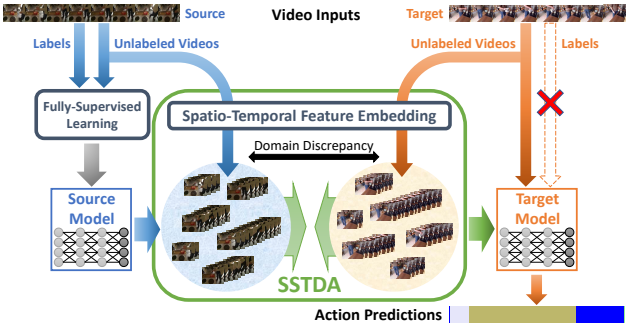

Figure 1: An overview of the proposed Self-Supervised Temporal Domain Adaptation (SSTDA) for action segmentation. "Source" refers to the data with labels, and "Target" refers to the data without access to labels. SSTDA can effectively adapts the source model trained with standard fully-supervised learning to a target domain by diminishing the discrepancy of embedded feature spaces between the two domains caused by spatio-temporal variations. SSTDA only requires unlabeled videos from both domains with the standard transductive setting, which eliminates the need of additional labels to obtain the final target model.

Despite significant progress made by recent methods based on temporal convolution with fully-supervised learning [20, 6, 23, 8], the performance is still not fully satisfactory (e.g. the best accuracy on the Breakfast dataset is still lower than 70%). One method to improve the performance is to exploit knowledge from larger-scale labeled data [2]. However, manually annotating precise frame-by-frame actions is time-consuming and challenging. Another way is to design more complicated architectures but with higher costs of model complexity. Thus, we aim to address the spatio-temporal variation problem with unlabeled data, which are comparatively easy to obtain. To achieve this goal, we propose to diminish the distributional discrepancy caused by spatio-temporal variations by exploiting auxiliary unlabeled videos with the same types of human activities performed by different people. More specifically, to extend the framework of the main video task for exploiting auxiliary

*Work done during an internship at Baidu USA

data [45, 19], we reformulate our main task as an unsupervised domain adaptation (DA) problem with the transductive setting [31, 5], which aims to reduce the discrepancy between source and target domains without access to the target labels.

Recently, adversarial-based DA approaches [10, 11, 37, 44] show progress in reducing the discrepancy for images using a domain discriminator equipped with adversarial training. However, videos also suffer from domain discrepancy along the temporal direction [4], so using image-based domain discriminators is not sufficient for action segmentation. Therefore, we propose **Self-Supervised Temporal Domain Adaptation (SSTDA)**, containing two self-supervised auxiliary tasks: 1) *binary domain prediction*, which predicts a single domain for each frame-level feature, and 2) *sequential domain prediction*, which predicts the permutation of domains for an untrimmed video. Through adversarial training with both auxiliary tasks, SSTDA can jointly align cross-domain feature spaces that embed local and global temporal dynamics, to address the spatio-temporal variation problem for action segmentation, as shown in Figure 1. To support our claims, we compare our method with other popular DA approaches and show better performance, demonstrating the effectiveness for aligning temporal dynamics by SSTDA. Finally, we evaluate our approaches on three datasets with high spatio-temporal variations: GTEA [9], 50Salads [35], and the Breakfast dataset [17]. By exploiting unlabeled target videos with SSTDA, our approach outperforms the current state-of-the-art methods by large margins and achieve comparable performance using only *65%* of labeled training data.

In summary, our contributions are three-fold:

1. **Self-Supervised Sequential Domain Prediction**: We propose a novel self-supervised auxiliary task, which predicts the permutation of domains for long videos, to facilitate video domain adaptation. To the best of our knowledge, this is the first self-supervised method designed for cross-domain action segmentation.

2. **Self-Supervised Temporal Domain Adaptation (SSTDA)**: By integrating two self-supervised auxiliary tasks, *binary* and *sequential domain prediction*, our proposed SSTDA can jointly align local and global embedded feature spaces across domains, outperforming other DA methods.

3. **Action Segmentation with SSTDA**: By integrating SSTDA for action segmentation, our approach outperforms the current state-of-the-art approach by large margins, and achieve comparable performance by using only *65%* of labeled training data. Moreover, different design choices are analyzed to identify the key contributions of each component.

2. Related Works

Action Segmentation methods proposed recently are built upon temporal convolution networks (TCN) [20, 6, 23, 8] because of their ability to capture long-range dependencies across frames and faster training compared to RNN-based methods. With the multi-stage pipeline, MS-TCN [8] performs hierarchical temporal convolutions to effectively extract temporal features and achieve the state-of-the-art performance for action segmentation. In this work, we utilize MS-TCN as the baseline model and integrate the proposed self-supervised modules to further boost the performance *without extra labeled data*.

Domain Adaptation (DA) has been popular recently especially with the integration of deep learning. With the two-branch (source and target) framework for most DA works, finding a common feature space between source and target domains is the ultimate goal, and the key is to design the domain loss to achieve this goal [5].

Discrepancy-based DA [24, 25, 26] is one of the major classes of methods where the main goal is to reduce the distribution distance between the two domains. *Adversarial-based DA* [10, 11] is also popular with similar concepts as GANs [12] by using domain discriminators. With carefully designed adversarial objectives, the domain discriminator and the feature extractor are optimized through min-max training. Some works further improve the performance by assigning pseudo-labels to target data [32, 41]. Furthermore, *Ensemble-based DA* [34, 21] incorporates multiple target branches to build an ensemble model. Recently, *Attention-based DA* [39, 18] assigns attention weights to different regions of images for more effective DA.

Unlike images, video-based DA is still under-explored. Most works concentrate on small-scale video DA datasets [36, 43, 14]. Recently, two larger-scale cross-domain video classification datasets along with the state-of-the-art approach are proposed [3, 4]. Moreover, some authors also proposed novel frameworks to utilize auxiliary data for other video tasks, including object detection [19] and action localization [45]. These works differ from our work by either different video tasks [19, 3, 4] or access to the labels of auxiliary data [45].

Self-Supervised Learning has become popular in recent years for images and videos given the ability to learn informative feature representations without human supervision. The key is to design an auxiliary task (or pretext task) that is related to the main task and the labels can be self-annotated. Most of the recent works for videos design auxiliary tasks based on spatio-temporal orders of videos [22, 40, 15, 1, 42]. Different from these works, our proposed auxiliary task predicts temporal permutation for cross-domain videos, aiming to address the problem of spatio-temporal variations for action segmentation.

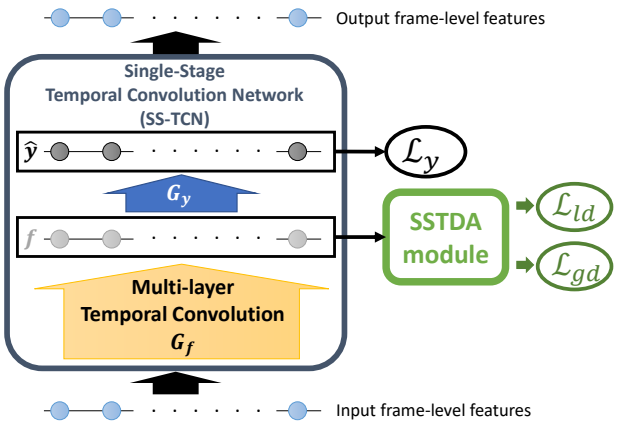

Figure 2: Illustration of the baseline model and the integration with our proposed SSTDA. The frame-level features $\boldsymbol{f}$ are obtained by applying the temporal convolution network G_f to the inputs, and converted to the corresponding predictions $\hat{\mathbf{y}}$ using a fully-connected layer G_y to calculate the prediction loss $\mathcal{L}_y$. The SSTDA module is integrated with $\boldsymbol{f}$ to calculate the local and global domain losses, $\mathcal{L}_{ld}$ and $\mathcal{L}_{gd}$ for optimizing $\boldsymbol{f}$ during training (see details in Section 3.2). Here we only show one stage in our multi-stage model.

3. Technical Approach

In this section, the baseline model which is the current state-of-the-art for action segmentation, MS-TCN [8], is reviewed first (Section 3.1). Then the novel temporal domain adaptation scheme consisting of two self-supervised auxiliary tasks, binary domain prediction (Section 3.2.1) and sequential domain prediction (Section 3.2.2), is proposed, followed by the final action segmentation model.

3.1. Baseline Model

Our work is built on the current state-of-the-art model for action segmentation, multi-stage temporal convolutional network (MS-TCN) [8]. For each stage, a single-stage TCN (SS-TCN) applies a multi-layer TCN, G_f, to derive the frame-level features $\boldsymbol{f} = \{f_1, f_2, ..., f_T\}$, and makes the corresponding predictions $\hat{\mathbf{y}} = \{\hat{y}_1, \hat{y}_2, ..., \hat{y}_T\}$ using a fully-connected layer G_y. By following [8], the prediction loss $\mathcal{L}_y$ is calculated based on the predictions $\hat{\mathbf{y}}$, as shown in the left part of Figure 2. Finally, multiple stages of SS-TCNs are stacked to enhance the temporal receptive fields, constructing the final baseline model, MS-TCN, where each stage takes the predictions from the previous stage as inputs, and makes predictions for the next stage.

3.2. Self-Supervised Temporal Domain Adaptation

Despite the promising performance of MS-TCN on action segmentation over previous methods, there is still a large room for improvement. One main challenge is the problem of *spatio-temporal variations* of human actions [16], causing the distributional discrepancy across domains [5]. For example, different subjects may perform the same action completely differently due to personalized spatio-temporal styles. Moreover, collecting annotated data for action segmentation is challenging and time-consuming. Thus, such challenges motivate the need to learn domain-invariant feature representations without full supervision. Inspired by the recent progress of self-supervised learning, which learns informative features that can be transferred to the main target tasks without external supervision (e.g. human annotation), we propose **Self-Supervised Temporal Domain Adaptation (SSTDA)** to diminish cross-domain discrepancy by designing self-supervised auxiliary tasks using unlabeled videos.

To effectively transfer knowledge, the self-supervised auxiliary tasks should be closely related to the main task, which is cross-domain action segmentation in this paper. Recently, adversarial-based DA approaches [10, 11] show progress in addressing cross-domain image problems using a domain discriminator with adversarial training where domain discrimination can be regarded as a self-supervised auxiliary task since domain labels are self-annotated. However, directly applying image-based DA for video tasks results in sub-optimal performance due to the temporal information being ignored [4]. Therefore, the question becomes: *How should we design the self-supervised auxiliary tasks to benefit cross-domain action segmentation?* More specifically, the answer should address both *cross-domain* and *action segmentation* problems.

To address this question, we first apply an auxiliary task *binary domain prediction* to predict the domain for each frame where the frame-level features are embedded with local temporal dynamics, aiming to address the cross-domain problems for videos in local scales. Then we propose a novel auxiliary task *sequential domain prediction* to temporally segment domains for untrimmed videos where the video-level features are embedded with global temporal dynamics, aiming to fully address the above question. Finally, SSTDA is achieved locally and globally by jointly applying these two auxiliary tasks, as illustrated in Figure 3.

In practice, since the key for effective video DA is to simultaneously align and learn temporal dynamics, instead of separating the two processes [4], we integrate SSTDA modules to multiple stages instead of the last stage only, and the single-stage integration is illustrated in Figure 2.

3.2.1 Local SSTDA

The main goal of action segmentation is to learn frame-level feature representations that encode spatio-temporal information so that the model can exploit information from multiple frames to predict the action for each frame. Therefore,

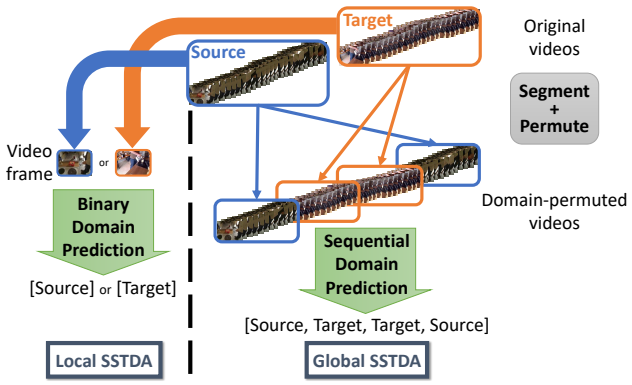

Figure 3: The two self-supervised auxiliary tasks in SSTDA: 1) *binary domain prediction*: discriminate single frame, 2) *sequential domain prediction*: predict a sequence of domains for an untrimmed video. These two tasks contribute to local and global SSTDA, respectively.

we first learn domain-invariant frame-level features with the auxiliary task *binary domain prediction* (Figure 3 left).

Binary Domain Prediction: For a single stage, we feed the frame-level features from source and target domains $\boldsymbol{f^S}$ and $\boldsymbol{f^T}$, respectively, to an additional shallow *binary domain classifier* G_{ld}, to discriminate which domain the features come from. Since temporal convolution from previous layers encodes information from multiple adjacent frames to each frame-level feature, those frames contribute to the binary domain prediction for each frame. Through adversarial training with a gradient reversal layer (GRL) [10, 11], which reverses the gradient signs during back-propagation, G_f will be optimized to gradually align the feature distributions between the two domains. Here we note $\hat{G}_{ld}$ as G_{ld} equipped with GRL, as shown in Figure 4.

Since this work is built on MS-TCN, *integrating* $\hat{G}_{ld}$ *with proper stages* is critical for effective DA. From our investigation, the best performance happens when $\hat{G}_{ld}$s are integrated into middle stages. See Section 4.3 for details.

The overall loss function becomes a combination of the baseline prediction loss $\mathcal{L}_y$ and the local domain loss $\mathcal{L}_{ld}$ with reverse sign, which can be expressed as follows:

$$\mathcal{L} = \sum^{N_s} \mathcal{L}_y - \sum^{\widetilde{N_s}} \beta_l \mathcal{L}_{ld} \tag{1}$$

$$\mathcal{L}_{ld} = \frac{1}{T} \sum_{j=1}^{T} L_{ld}(G_{ld}(f_j), d_j) \tag{2}$$

where N_s is the total stage number in MS-TCN, $\widetilde{N_s}$ is the number of stages integrated with $\hat{G}_{ld}$, and T is the total frame number of a video. L_{ld} is a binary cross-entropy loss function, and β_l is the trade-off weight for local domain loss $\mathcal{L}_{ld}$, obtained by following the common strategy as [10, 11].

3.2.2 Global SSTDA

Although frame-level features $\boldsymbol{f}$ is learned using the context and dependencies from neighbor frames, the temporal receptive fields of $\boldsymbol{f}$ are still limited, unable to represent full videos. Solely integrating DA into $\boldsymbol{f}$ cannot fully address spatio-temporal variations for untrimmed long videos. Therefore, in addition to binary domain prediction for frame-level features, we propose the second self-supervised auxiliary task for video-level features: **sequential domain prediction**, which predicts a sequence of domains for video clips, as shown in the right part of Figure 3. This task is a temporal domain segmentation problem, aiming to predict the correct permutation of domains for long videos consisting of shuffled video clips from both source and target domains. Since this goal is related to both cross-domain and action segmentation problems, *sequential domain prediction* can effectively benefit our main task.

More specifically, we first divide $\boldsymbol{f^S}$ and $\boldsymbol{f^T}$ into two sets of segments $\boldsymbol{F^S} = \{\boldsymbol{f_a^S}, \boldsymbol{f_b^S}, ...\}$ and $\boldsymbol{F^T} = \{\boldsymbol{f_a^T}, \boldsymbol{f_b^T}, ...\}$, respectively, and then learn the corresponding two sets of segment-level feature representations $\boldsymbol{V^S} = \{v_a^S, v_b^S, ...\}$ and $\boldsymbol{V^T} = \{v_a^T, v_b^T, ...\}$ with *Domain Attentive Temporal Pooling (DATP)*. All features v are then shuffled and combined in random order and fed to a *sequential domain classifier* G_{gd} equipped with GRL (noted as $\hat{G}_{gd}$) to predict the permutation of domains, as shown in Figure 4.

Domain Attentive Temporal Pooling (DATP): The most straightforward method to obtain a video-level feature is to aggregate frame-level features using *temporal pooling*. However, not all the frame-level features contribute the same to the overall domain discrepancy, as mentioned in [4]. Hence, we assign larger attention weights w_j (calculated using $\hat{G}_{gd}$ in local SSTDA) to the features which have larger domain discrepancy so that we can focus more on aligning those features. Finally, the attended frame-level features are aggregated with temporal pooling to generate the video-level feature v, which can be expressed as:

$$v = \frac{1}{T'} \sum_{j=1}^{T'} w_j \cdot f_j \tag{3}$$

where T' is the number of frames in a video segment. For more details, please refer to the supplementary.

Sequential Domain Prediction: By separately applying DATP to both source and target segments, respectively, a set of segment-level feature representations $\boldsymbol{V} = \{v_a^S, v_b^S, ..., v_a^T, v_b^T, ...\}$ are obtained. We then shuffle all the features in $\boldsymbol{V}$ and concatenate them into a feature to represent a long and untrimmed video $\boldsymbol{V'}$, which contains video segments from both domains in random order. Finally, $\boldsymbol{V'}$ is fed into a *sequential domain classifier* G_{gd} to predict the permutation of domains for the video segments. For example, if $\boldsymbol{V'} = [v_a^S, v_a^T, v_b^T, v_b^S]$, the goal of G_{gd} is to predict

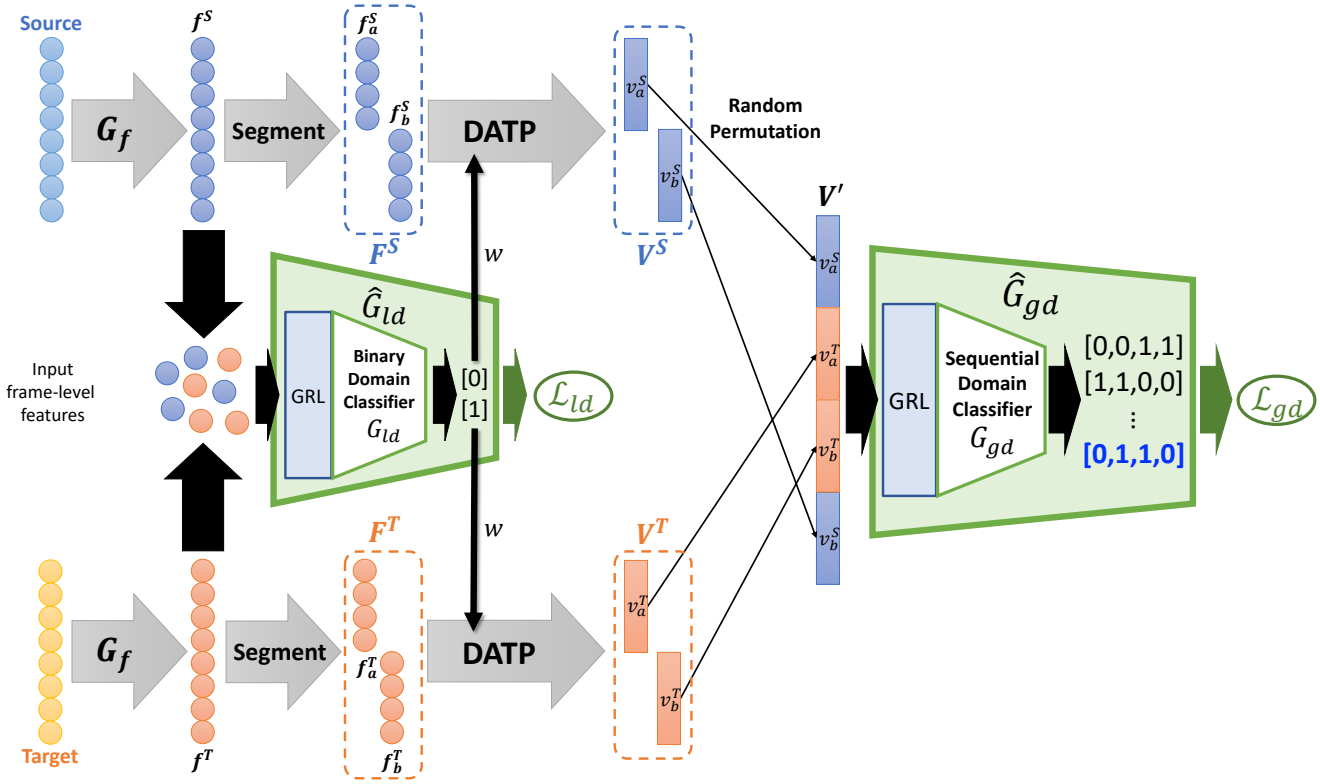

Figure 4: The overview of the proposed Self-Supervised Temporal Domain Adaptation (SSTDA). The inputs from the two domains are first encoded with local temporal dynamics using G_f to obtain the frame-level features $\boldsymbol{f^S}$ and $\boldsymbol{f^T}$, respectively. We apply local SSTDA on all $\boldsymbol{f}$ using binary domain prediction $\hat{G}_{ld}$. Besides, $\boldsymbol{f^S}$ and $\boldsymbol{f^T}$ are evenly divided into multiple segments to learn segment-level features $\boldsymbol{V^S}$ and $\boldsymbol{V^T}$ by DATP, respectively. Finally, the global SSTDA is applied on $\boldsymbol{V'}$, which is generated by concatenating shuffled $\boldsymbol{V^S}$ and $\boldsymbol{V^T}$, using sequential domain prediction $\hat{G}_{gd}$. $\mathcal{L}_{ld}$ and $\mathcal{L}_{gd}$ are the domain losses from $\hat{G}_{ld}$ and $\hat{G}_{gd}$, respectively. w corresponds to the attention weights for DATP, which are calculated form the outputs of $\hat{G}_{ld}$. Here we use 8-frame videos and 2 segments as an example for this figure. Best views in colors.

the permutation as $[0, 1, 1, 0]$. G_{gd} is a multi-class classifier where the class number corresponds to the total number of all possible permutations of domains, and the complexity of G_{gd} is determined by the segment number for each video (more analyses in Section 4.3). The outputs of G_{gd} are used to calculate the global domain loss $\mathcal{L}_{gd}$ as below:

$$\mathcal{L}_{gd} = L_{gd}(G_{gd}(\boldsymbol{V'})), y_d) \tag{4}$$

where L_{gd} is also a standard cross-entropy loss function where the class number is determined by the segment number. Through adversarial training with GRL, *sequential domain prediction* also contributes to optimizing G_f to align the feature distributions between the two domains.

There are some self-supervised learning works also proposing the concepts of *temporal shuffling* [22, 42]. However, they predict temporal orders within one domain, aiming to learn general temporal information for video features. Instead, our method predicts temporal permutation for cross-domain videos, which are shown with a dual-branch pipeline in Figure 4, and integrate with binary domain prediction to effectively address both *cross-domain* and *action segmentation* problems.

3.2.3 Local-Global Joint Training.

Finally, we also adopt a strategy from [39] to minimize the class entropy for the frames that are similar across domains by adding a domain attentive entropy (DAE) loss $\mathcal{L}_{ae}$. Please refer to the supplementary for more details.

By adding the global domain loss $\mathcal{L}_{gd}$ (Equation (4)) and the attentive entropy loss $\mathcal{L}_{ae}$ into Equation (1), the overall loss of our final proposed **Self-Supervised Temporal Domain Adaptation (SSTDA)** can be expressed as follows:

$$\mathcal{L} = \sum^{N_s} \mathcal{L}_y - \sum^{\widetilde{N_s}} (\beta_l \mathcal{L}_{ld} + \beta_g \mathcal{L}_{gd} - \mu \mathcal{L}_{ae}) \tag{5}$$

where β_g and μ are the weights for $\mathcal{L}_{gd}$ and $\mathcal{L}_{ae}$, respectively.

	GTEA	50Salads	Breakfast
subject #	4	25	52
class #	11	17	48
video #	28	50	1712
leave-#-subject-out	1	5	13

Table 1: The statistics of action segmentation datasets.

4. Experiments

To validate the effectiveness of the proposed methods in reducing spatial-temporal discrepancy for action segmentation, we choose three challenging datasets: GTEA [9], 50Salads [35], and Breakfast [17], which separate the training and validation sets by different people (noted as *subjects*) with leave-subjects-out cross-validation for evaluation, resulting in large domain shift problem due to spatio-temporal variations. Therefore, we regard the training set as *Source* domain, and the validation set as *Target* domain with the standard transductive unsupervised DA protocol [31, 5]. See the supplementary for more implementation details.

4.1. Datasets and Evaluation Metrics

The overall statistics of the three datasets are listed in Table 1. Three widely used evaluation metrics are chosen as follows [20]: frame-wise *accuracy (Acc)*, segmental *edit score*, and segmental F1 score at the IoU threshold $k\%$, denoted as *F1@k* ($k = \{10, 25, 50\}$). While *Acc* is the most common metric, *edit* and *F1 score* both consider the temporal relation between predictions and ground truths, better reflecting the performance for action segmentation.

4.2. Experimental Results

We first investigate the effectiveness of our approaches in utilizing unlabeled target videos for action segmentation. We choose MS-TCN [8] as the backbone model since it is the current state of the art for this task. "Source only" means the model is trained only with source labeled videos, i.e., the baseline model. And then our approach is compared to other methods with the same transductive protocol. Finally, we compare our method to the most recent action segmentation methods on all three datasets, and investigate how our method can reduce the reliance on source labeled data.

Self-Supervised Temporal Domain Adaptation: First we investigate the performance of local SSTDA by integrating the auxiliary task *binary domain prediction* with the baseline model. The results on all three datasets are improved significantly, as shown in Table 2. For example, on the GTEA dataset, our approach outperforms the baseline by 4.3% for F1@25, 3.2% for the edit score and 3.6% for the frame-wise accuracy. Although local SSTDA mainly works on the frame-level features, the temporal information is still encoded using the context from neighbor frames, helping address the variation problem for videos across domains.

GTEA	F1@{10, 25, 50}			Edit	Acc
Source only (MS-TCN)†	86.5	83.6	71.9	81.3	76.5
Local SSTDA	89.6	87.9	74.4	84.5	**80.1**
SSTDA‡	**90.0**	**89.1**	**78.0**	**86.2**	79.8
50Salads	F1@{10, 25, 50}			Edit	Acc
Source only (MS-TCN)†	75.4	73.4	65.2	68.9	82.1
Local SSTDA	79.2	77.8	70.3	72.0	82.8
SSTDA‡	**83.0**	**81.5**	**73.8**	**75.8**	**83.2**
Breakfast	F1@{10, 25, 50}			Edit	Acc
Source only (MS-TCN)†	65.3	59.6	47.2	65.7	64.7
Local SSTDA	72.8	67.8	55.1	71.7	**70.3**
SSTDA‡	**75.0**	**69.1**	**55.2**	**73.7**	70.2

Table 2: The experimental results for our approaches on three benchmark datasets. "SSTDA" refers to the full model while "Local SSTDA" only contains binary domain prediction. †We achieve higher performance than reported in [8] when using the released code, so use that as the baseline performance for the whole paper. ‡Global SSTDA requires outputs from local SSTDA, so it is not evaluated alone.

Despite the improvement from local SSTDA, integrating DA into frame-level features cannot fully address the problem of spatio-temporal variations for long videos. Therefore, we integrate our second proposed auxiliary task sequential domain prediction for untrimmed long videos. By jointly training with both auxiliary tasks, SSTDA can jointly align cross-domain feature spaces embedding with local and global temporal dynamics, and further improve over local SSTDA with significant margins. For example, on the 50Salads dataset, it outperforms local SSTDA by 3.8% for F1@10, 3.7% for F1@25, 3.5% for F1@50, and 3.8% for the edit score, as shown in Table 2.

One interesting finding is that local SSTDA contributes to most of the frame-wise accuracy improvement for SSTDA because it focuses on aligning frame-level feature spaces. On the other hand, sequential domain prediction benefits aligning video-level feature spaces, contributing to further improvement for the other two metrics, which consider temporal relation for evaluation.

Learning from Unlabeled Target Videos: We also compare SSTDA with other popular approaches [11, 26, 32, 41, 34, 21, 42] to validate the effectiveness of reducing spatio-temporal discrepancy with the same amount of unlabeled target videos. For the fair comparison, we integrate all these methods with the same baseline model, MS-TCN. For more implementation details, please refer to the supplementary.

Table 3 shows that our proposed SSTDA outperforms all the other investigated DA methods in terms of the two metrics that consider temporal relation. We conjecture the main reason is that all these DA approaches are designed for cross-domain image problems. Although they are in-

	F1@{10, 25, 50}			Edit
Source only (MS-TCN)	86.5	83.6	71.9	81.3
VCOP [42]	87.3	85.9	70.1	82.2
DANN [11]	89.6	87.9	74.4	84.5
JAN [26]	88.7	87.6	73.1	83.1
MADA [32]	88.6	86.7	75.8	83.5
MSTN [41]	89.9	88.2	75.9	84.7
MCD [34]	88.1	86.3	73.4	82.7
SWD [21]	89.0	87.3	73.8	84.4
SSTDA	**90.0**	**89.1**	**78.0**	**86.2**

Table 3: The comparison of different methods that can learn information from unlabeled target videos (on GTEA). All the methods are integrated with the same baseline model MS-TCN for fair comparison. Please refer to the supplementary for the results on other datasets.

tegrated with frame-level features which encode local temporal dynamics, the limited temporal receptive fields prevent them from fully addressing temporal domain discrepancy. Instead, the *sequential domain prediction* in SSTDA is directly applied to the whole untrimmed video, helping to globally align the cross-domain feature spaces that embed longer temporal dynamics, so that spatio-temporal variations can be reduced more effectively.

We also compare with the most recent video-based self-supervised learning method, [42], which can also learn temporal dynamics from unlabeled target videos. However, the performance is even worse than other DA methods, implying that temporal shuffling *within single domain* does not effectively benefit cross-domain action segmentation.

Comparison with Action Segmentation Methods: Here we compare the recent methods to SSTDA trained with two settings: 1) fully source labels, and 2) weakly source labels.

The first setting means we have labels for all the frames in source videos, and SSTDA outperforms all the previous methods on the three datasets with respect to all evaluation metrics. For example, SSTDA outperforms currently the state-of-the-art fully-supervised method, MS-TCN [8], by large margins (e.g. 8.1% for F1@25, 8.6% for F1@50, and 6.9% for the edit score on 50Salads; 9.5% for F1@25, 8.0% for F1@50, and 8.0% for the edit score on Breakfast), as demonstrated in Table 4. Since no additional labeled data is used, these results indicate how our proposed SSTDA address the spatio-temporal variation problem with unlabeled videos to improve the action segmentation performance.

Given the significant improvement by exploiting unlabeled target videos, it implies the potential to train with fewer number of labeled frames using SSTDA, which is our second setting. In this setting, we drop labeled frames from source domains with uniform sampling for training, and evaluate on the same length of validation data. Our experiment indicates that by integrating with SSTDA, only

GTEA	F1@{10, 25, 50}			Edit	Acc
LCDC [29]	75.4	-	-	72.8	65.3
TDRN [23]	79.2	74.4	62.7	74.1	70.1
MS-TCN [8]†	86.5	83.6	71.9	81.3	76.5
SSTDA (65%)	85.2	82.6	69.3	79.6	75.7
SSTDA	**90.0**	**89.1**	**78.0**	**86.2**	**79.8**
50Salads	F1@{10, 25, 50}			Edit	Acc
TDRN [23]	72.9	68.5	57.2	66.0	68.1
LCDC [29]	73.8	-	-	66.9	72.1
MS-TCN [8]†	75.4	73.4	65.2	68.9	82.1
SSTDA (65%)	77.7	75.0	66.2	69.3	80.7
SSTDA	**83.0**	**81.5**	**73.8**	**75.8**	**83.2**
Breakfast	F1@{10, 25, 50}			Edit	Acc
TCFPN [7]	-	-	-	-	52.0
GRU [33]	-	-	-	-	60.6
MS-TCN [8]†	65.3	59.6	47.2	65.7	64.7
SSTDA (65%)	69.3	62.9	49.4	69.0	65.8
SSTDA	**75.0**	**69.1**	**55.2**	**73.7**	**70.2**

Table 4: Comparison with the most recent action segmentation methods on all three datasets. SSTDA (65%) means training with 65% of total labeled training data. †Results from running the official code, as explained in Table 2.

	F1@{10, 25, 50}			Edit	Acc
Source only	86.5	83.6	71.9	81.3	76.5
$\{S1\}$	88.6	86.2	73.6	84.2	78.7
$\{S2\}$	89.1	87.2	**74.4**	84.3	79.1
$\{S3\}$	89.2	87.3	72.3	83.8	78.9
$\{S4\}$	88.1	86.4	73.0	83.0	78.8
$\{S1, S2\}$	89.0	85.8	73.5	**84.8**	79.5
$\{S2, S3\}$	**89.6**	**87.9**	**74.4**	84.5	**80.1**
$\{S3, S4\}$	88.3	86.8	73.9	83.6	78.6

Table 5: The experimental results of design choice for local SSTDA (on GTEA). $\{S_n\}$: add $\hat{G}_{ld}$ to the nth stage of MS-TCN, where smaller n implies closer to inputs.

65% of labeled training data are required to achieve comparable performance with MS-TCN, as shown in the "SSTDA (65%)" row in Table 4. For the full experiments about labeled data reduction, please refer to the supplementary.

4.3. Ablation Study and Analysis

Design Choice for Local SSTDA: Since we develop our approaches upon MS-TCN [8], it raises the question: *How to effectively integrate binary domain prediction to a multi-stage architecture?* To answer this, we first integrate $\hat{G}_{ld}$ into each stage and the results show that the best performance happens when the $\hat{G}_{ld}$ is integrated into middle stages, such as $S2$ or $S3$, as shown in Table 5. $S1$ is not a good choice for DA because it corresponds to low-level features with less discriminability where DA shows limited effects [24], and represents less temporal receptive fields for

Segment #	F1@{10, 25, 50}			Edit	Acc
1	89.4	87.7	75.4	85.3	79.2
2	**90.0**	**89.1**	**78.0**	**86.2**	**79.8**
3	89.7	87.6	75.4	85.2	79.2

Table 6: The experimental results for different segment numbers of sequential domain prediction (on GTEA).

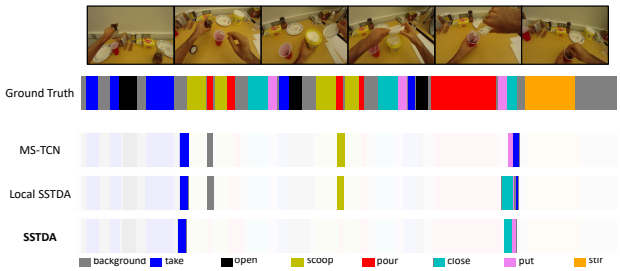

Figure 5: The visualization of temporal action segmentation for our methods with color-coding (input example: *make coffee*). "MS-TCN" is the baseline model without any DA methods. We only highlight the action segments that are different from the ground truth for clear comparison.

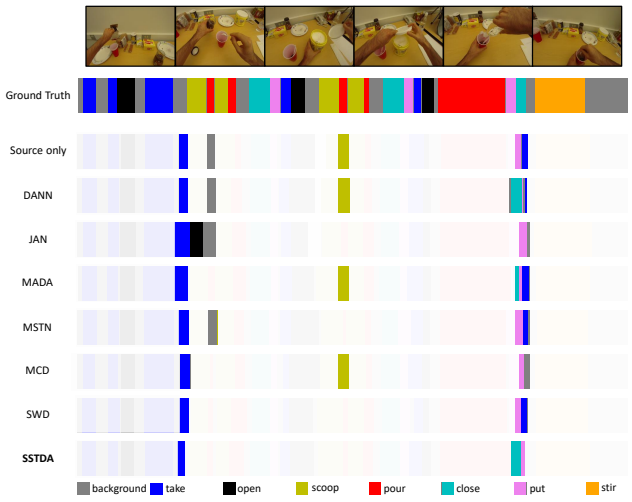

Figure 6: The visualization of temporal action segmentation for different DA methods (same input as Figure 5). "Source only" represents the baseline model, MS-TCN. Only the segments different from the ground truth are highlighted.

videos. However, higher stages (e.g. $S4$) are not always better. We conjecture that it is because the model fits more to the source data, causing difficulty for DA. In our case, integrating $\hat{G}_{ld}$ into $S2$ provides the best overall performance.

We also integrate binary domain prediction with multiple stages. However, multi-stage DA does not always guarantee improved performance. For example, $\{S1, S2\}$ has worse results than $\{S2\}$ in terms of F1@$\{10, 25, 50\}$. Since $\{S2\}$ and $\{S3\}$ provide the best single-stage DA performance, we use $\{S2, S3\}$, which performs the best, as the final model for all our approaches in all the experiments.

Design Choice for Global SSTDA: The most critical design decision for the sequential domain prediction is the segment number for each video. In our implementation, we divide one source video into m segments and do so for one target video, and then apply G_{gd} to predict the permutation of domains for these $2m$ video segments. Therefore, the category number of G_{gd} equals the number of all permutations $(2m)!/(m!)^2$. In other words, the segment number m determine the complexity of the self-supervised auxiliary task. For example, $m = 3$ leads to a 20-way classifier, and $m = 4$ results in a 70-way classifier. Since a good self-supervised task should be neither naive nor over complicated [30], we choose $m = 2$ as our final decision, which is supported by our experiments as shown in Table 6.

Segmentation Visualization: It is also common to evaluate the qualitative performance to ensure that the prediction results are aligned with human vision. First, we compare our approaches with the baseline model MS-TCN [8] and the ground truth, as shown in Figure 5. MS-TCN fails to detect some *pour* actions in the first half of the video, and falsely classify *close* as *take* in the latter part of the video. With local SSTDA, our approach can detect *close* in the latter part of the video. Finally, with full SSTDA, our proposed method also detects all *pour* action segments in the first half of video. We then compare SSTDA with other DA methods, and Figure 6 shows that our result is the closest to the ground truth. The others either incorrectly detect some actions or make incorrect classification. For more qualitative results, please refer to the supplementary.

5. Conclusions and Future Work

In this work, we propose a novel approach to effectively exploit unlabeled target videos to boost performance for action segmentation without target labels. To address the problem of spatio-temporal variations for videos across domains, we propose **Self-Supervised Temporal Domain Adaptation (SSTDA)** to jointly align cross-domain feature spaces embedded with local and global temporal dynamics by two self-supervised auxiliary tasks, *binary* and *sequential domain prediction*. Our experiments indicate that SSTDA outperforms other DA approaches by aligning temporal dynamics more effectively. We also validate the proposed SSTDA on three challenging datasets (GTEA, 50Salads, and Breakfast), and show that SSTDA outperforms the current state-of-the-art method by large margins and only requires 65% of the labeled training data to achieve the comparable performance, demonstrating the usefulness of adapting to unlabeled videos across variations. For the future work, we plan to apply SSTDA to more challenging video tasks (e.g. spatio-temporal action localization [13]).

References

[1] Unaiza Ahsan, Rishi Madhok, and Irfan Essa. Video jigsaw: Unsupervised learning of spatiotemporal context for video action recognition. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2019. 2

[2] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1

[3] Min-Hung Chen, Zsolt Kira, and Ghassan AlRegib. Temporal attentive alignment for video domain adaptation. *CVPR Workshop on Learning from Unlabeled Videos*, 2019. 2

[4] Min-Hung Chen, Zsolt Kira, Ghassan AlRegib, Jaekwon Woo, Ruxin Chen, and Jian Zheng. Temporal attentive alignment for large-scale video domain adaptation. In *IEEE International Conference on Computer Vision (ICCV)*, 2019. 2, 3, 4

[5] Gabriela Csurka. A comprehensive survey on domain adaptation for visual applications. In *Domain Adaptation in Computer Vision Applications*, pages 1–35. Springer, 2017. 2, 3, 6

[6] Li Ding and Chenliang Xu. Tricornet: A hybrid temporal convolutional and recurrent network for video action segmentation. *arXiv preprint arXiv:1705.07818*, 2017. 1, 2

[7] Li Ding and Chenliang Xu. Weakly-supervised action segmentation with iterative soft boundary assignment. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 7

[8] Yazan Abu Farha and Jurgen Gall. Ms-tcn: Multi-stage temporal convolutional network for action segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 1, 2, 3, 6, 7, 8

[9] Alireza Fathi, Xiaofeng Ren, and James M Rehg. Learning to recognize objects in egocentric activities. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011. 2, 6

[10] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *International Conference on Machine Learning (ICML)*, 2015. 2, 3, 4

[11] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research (JMLR)*, 17(1):2096–2030, 2016. 2, 3, 4, 6, 7

[12] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2014. 2

[13] Chunhui Gu, Chen Sun, David A Ross, Carl Vondrick, Caroline Pantofaru, Yeqing Li, Sudheendra Vijayanarasimhan, George Toderici, Susanna Ricco, Rahul Sukthankar, et al. Ava: A video dataset of spatio-temporally localized atomic visual actions. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 8

[14] Arshad Jamal, Vinay P Namboodiri, Dipti Deodhare, and KS Venkatesh. Deep domain adaptation in action space. In *British Machine Vision Conference (BMVC)*, 2018. 2

[15] Dahun Kim, Donghyeon Cho, and In So Kweon. Self-supervised video representation learning with space-time cubic puzzles. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2019. 2

[16] Yu Kong and Yun Fu. Human action recognition and prediction: A survey. *arXiv preprint arXiv:1806.11230*, 2018. 1, 3

[17] Hilde Kuehne, Ali Arslan, and Thomas Serre. The language of actions: Recovering the syntax and semantics of goal-directed human activities. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014. 2, 6

[18] Vinod Kumar Kurmi, Shanu Kumar, and Vinay P Namboodiri. Attending to discriminative certainty for domain adaptation. In *IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2

[19] Avisek Lahiri, Sri Charan Ragireddy, Prabir Biswas, and Pabitra Mitra. Unsupervised adversarial visual level domain adaptation for learning video object detectors from images. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2019. 2

[20] Colin Lea, Michael D Flynn, Rene Vidal, Austin Reiter, and Gregory D Hager. Temporal convolutional networks for action segmentation and detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1, 2, 6

[21] Chen-Yu Lee, Tanmay Batra, Mohammad Haris Baig, and Daniel Ulbricht. Sliced wasserstein discrepancy for unsupervised domain adaptation. In *IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2, 6, 7

[22] Hsin-Ying Lee, Jia-Bin Huang, Maneesh Singh, and Ming-Hsuan Yang. Unsupervised representation learning by sorting sequences. In *IEEE International Conference on Computer Vision (ICCV)*, 2017. 2, 5

[23] Peng Lei and Sinisa Todorovic. Temporal deformable residual networks for action segmentation in videos. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 1, 2, 7

[24] Mingsheng Long, Yue Cao, Jianmin Wang, and Michael Jordan. Learning transferable features with deep adaptation networks. In *International Conference on Machine Learning (ICML)*, 2015. 2, 7

[25] Mingsheng Long, Han Zhu, Jianmin Wang, and Michael I Jordan. Unsupervised domain adaptation with residual transfer networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2016. 2

[26] Mingsheng Long, Han Zhu, Jianmin Wang, and Michael I Jordan. Deep transfer learning with joint adaptation networks. In *International Conference on Machine Learning (ICML)*, 2017. 2, 6, 7

[27] Chih-Yao Ma, Min-Hung Chen, Zsolt Kira, and Ghassan AlRegib. Ts-lstm and temporal-inception: Exploiting spatiotemporal dynamics for activity recognition. *Signal Processing: Image Communication*, 71:76–87, 2019. 1

[28] Chih-Yao Ma, Asim Kadav, Iain Melvin, Zsolt Kira, Ghassan AlRegib, and Hans Peter Graf. Attend and interact: higher-order object interactions for video understanding. In *IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 1

[29] Khoi-Nguyen C Mac, Dhiraj Joshi, Raymond A Yeh, Jinjun Xiong, Rogerio S Feris, and Minh N Do. Learning motion in feature space: Locally-consistent deformable convolution networks for fine-grained action detection. In *IEEE International Conference on Computer Vision (ICCV)*, 2019. 7

[30] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *European Conference on Computer Vision (ECCV)*, 2016. 8

[31] Sinno Jialin Pan, Qiang Yang, et al. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, 22(10):1345–1359, 2010. 2, 6

[32] Zhongyi Pei, Zhangjie Cao, Mingsheng Long, and Jianmin Wang. Multi-adversarial domain adaptation. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2018. 2, 6, 7

[33] Alexander Richard, Hilde Kuehne, and Juergen Gall. Weakly supervised action learning with rnn based fine-to-coarse modeling. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 7

[34] Kuniaki Saito, Kohei Watanabe, Yoshitaka Ushiku, and Tatsuya Harada. Maximum classifier discrepancy for unsupervised domain adaptation. In *IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 2, 6, 7

[35] Sebastian Stein and Stephen J McKenna. Combining embedded accelerometers with computer vision for recognizing food preparation activities. In *ACM international joint conference on Pervasive and ubiquitous computing (UbiComp)*, 2013. 2, 6

[36] Waqas Sultani and Imran Saleemi. Human action recognition across datasets by foreground-weighted histogram decomposition. In *IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, 2014. 2

[37] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 2

[38] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 1

[39] Ximei Wang, Liang Li, Weirui Ye, Mingsheng Long, and Jianmin Wang. Transferable attention for domain adaptation. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2019. 2, 5

[40] Donglai Wei, Joseph J Lim, Andrew Zisserman, and William T Freeman. Learning and using the arrow of time. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 2

[41] Shaoan Xie, Zibin Zheng, Liang Chen, and Chuan Chen. Learning semantic representations for unsupervised domain adaptation. In *International Conference on Machine Learning (ICML)*, 2018. 2, 6, 7

[42] Dejing Xu, Jun Xiao, Zhou Zhao, Jian Shao, Di Xie, and Yueting Zhuang. Self-supervised spatiotemporal learning via video clip order prediction. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2, 5, 6, 7

[43] Tiantian Xu, Fan Zhu, Edward K Wong, and Yi Fang. Dual many-to-one-encoder-based transfer learning for cross-dataset human action recognition. *Image and Vision Computing*, 55:127–137, 2016. 2

[44] Weichen Zhang, Wanli Ouyang, Wen Li, and Dong Xu. Collaborative and adversarial network for unsupervised domain adaptation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 2

[45] Xiao-Yu Zhang, Haichao Shi, Changsheng Li, Kai Zheng, Xiaobin Zhu, and Lixin Duan. Learning transferable self-attentive representations for action recognition in untrimmed videos with weak supervision. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2019. 2