Contrast and Mix: Temporal Contrastive Video Domain Adaptation with Background Mixing

Aadarsh Sahoo[1] **Rutav Shah**[1] **Rameswar Panda**[2] **Kate Saenko**[2,3] **Abir Das**[1]
[1] IIT Kharagpur, [2] MIT-IBM Watson AI Lab, [3] Boston University
{sahoo_aadarsh@, rutavms@, abir@cse.}iitkgp.ac.in, rpanda@ibm.com, saenko@bu.edu

Abstract

Unsupervised domain adaptation which aims to adapt models trained on a labeled source domain to a completely unlabeled target domain has attracted much attention in recent years. While many domain adaptation techniques have been proposed for images, the problem of unsupervised domain adaptation in videos remains largely underexplored. In this paper, we introduce Contrast and Mix (CoMix), a new contrastive learning framework that aims to learn discriminative invariant feature representations for unsupervised video domain adaptation. First, unlike existing methods that rely on adversarial learning for feature alignment, we utilize temporal contrastive learning to bridge the domain gap by maximizing the similarity between encoded representations of an unlabeled video at two different speeds as well as minimizing the similarity between different videos played at different speeds. Second, we propose a novel extension to the temporal contrastive loss by using background mixing that allows additional positives per anchor, thus adapting contrastive learning to leverage action semantics shared across both domains. Moreover, we also integrate a supervised contrastive learning objective using target pseudo-labels to enhance discriminability of the latent space for video domain adaptation. Extensive experiments on several benchmark datasets demonstrate the superiority of our proposed approach over state-of-the-art methods. Project page: https://cvir.github.io/projects/comix.

1 Introduction

Unsupervised domain adaptation (UDA), which alleviates the requirement of large amounts of annotated data by adapting a model learned on a labelled source domain to an unlabelled target domain, has drawn a great deal of attention in the last few years [13, 85]. Much progress has been made in developing deep UDA methods by minimizing the cross-domain divergence [42, 74], adding adversarial domain discriminators [21, 79], and image-to-image translation techniques [27, 55]. However, despite impressive results on commonly used benchmark datasets (*e.g.*, [65, 80, 61]), most of the methods have been developed only for images and not for videos, where the annotation task is often more complicated requiring tedious human labor in comparison to images.

More recently, very few works have attempted deep UDA for video action recognition by directly matching segment-level features [9, 28, 54, 45] or with attention weights [12, 57]. However, (1) trivially matching segment-level feature distributions by extending the image-specific approaches, without considering the rich temporal information may not alone be sufficient for video domain adaptation; (2) prior methods often focus on aligning target features with source, rather than exploiting any action semantics shared across both domains (*e.g.*, difference in background with the same action: videos in the top row of Figure 1 are from the source and target domain respectively, but both capture the same action *walking*); (3) existing methods often rely on complex adversarial learning which is unwieldy to train, resulting in very fragile convergence.

35th Conference on Neural Information Processing Systems (NeurIPS 2021).

Meanwhile, self-supervised pretext tasks like predicting rotation and translation have recently emerged as an alternative to adversarial learning for unsupervised domain adaptation in images [41, 75]. While these works show the promising potential of self-supervised learning in aligning source and target domains, the more recent very successful contrastive representation learning [10, 24, 56] has never been used to adapt video action recognition models to target domains. Motivated by this, in this paper, we explore the following natural, yet important question: *whether and how contrastive learning could be exploited for the challenging and practically important task of unsupervised video domain adaptation for human action recognition?*

To this end, we introduce Contrast and Mix (`CoMix`), a simple yet effective approach based on contrastive learning to adapt video action recognition models trained on a labeled source domain to unlabelled target domains. First, we propose to represent video as a graph and then utilize temporal contrastive self-supervised learning over the graph representations as a nexus between source and target domains to align features, without requiring any additional adversarial learning, as most prior works do in video domain adaptation [9, 12, 57]. Specifically, we maximize the similarity between encoded representations of the same video at two different speeds as well as minimize the similarity between different videos played at different speeds, leveraging the fact that changing video speed does not change an action on both domains. While minimization of contrastive self-supervised losses in both domains simultaneously helps in domain alignment, it ignores action semantics shared across them as the loss treats each domain individually. To alleviate this, we incorporate new synthetic videos into the temporal contrastive objective, which are obtained by mixing background of a video from one domain to a video from another domain, as shown in Figure 1 (bottom). Importantly, since mixing background doesn't change the temporal dynamics, we introduce pseudo-labels for the mixed videos to be same as the label of the original videos and consider additional positives per anchor (see Figure 2), which encourages the model to generalize to new samples that may not be covered by temporal contrastive learning in hand. In other words, mixed background video of an input sample in the embedding space act as small semantic perturbations that are not imaginary, i.e., they are representative of the action semantics shared across source and target domains. Finally, rather than relying only on the supervision of source categories to learn a discriminative representation, we generate pseudo-labels for the target samples in every batch and then harness the label information using a temporal supervised contrastive term, that pushes the examples from the same class close and the examples from different classes further apart (Figure 2: right). While our modified contrastive losses are motivated by the supervised contrastive learning [31], we use pseudo labels for exploiting shared action semantics and discriminative information from target domain, instead of using true labels as an alternative to supervised cross-entropy loss (which is not present for target samples). To the best of our knowledge, ours is the first work that successfully leverages contrastive learning in an unified framework to align cross-domain features while enhancing discriminabilty of the latent space for unsupervised video domain adaptation.

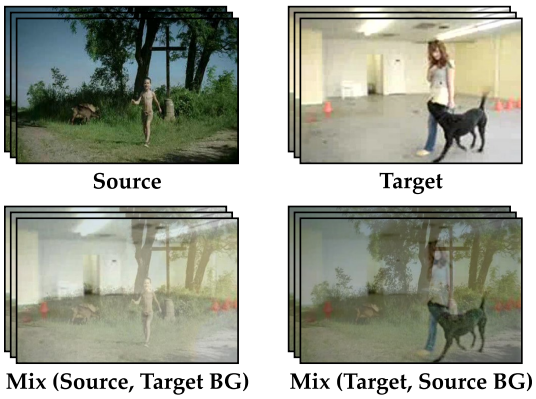

Figure 1: **Background Mixing.** Top row shows two representative videos from the source and target domain respectively. Both videos capture the same action "*walking*" with different backgrounds. Bottom row shows videos obtained after mixing target background with source video and vice versa.

To summarize, the main **contributions** of our work are as follows:

- We introduce Contrast and Mix (`CoMix`), a new contrastive learning framework to learn discriminative invariant feature representations for unsupervised video domain adaptation. Overall, `CoMix` is simple and easy to implement which perfectly fits into modern mini-batch end-to-end training.

- We propose a novel extension to temporal contrastive loss by using background mixing that allows additional positives per anchor, thus adapting contrastive learning to leverage action semantics shared across both domains. We also integrate a supervised contrastive learning objective using pseudo label information from the target domain to enhance discriminabilty of the latent space.

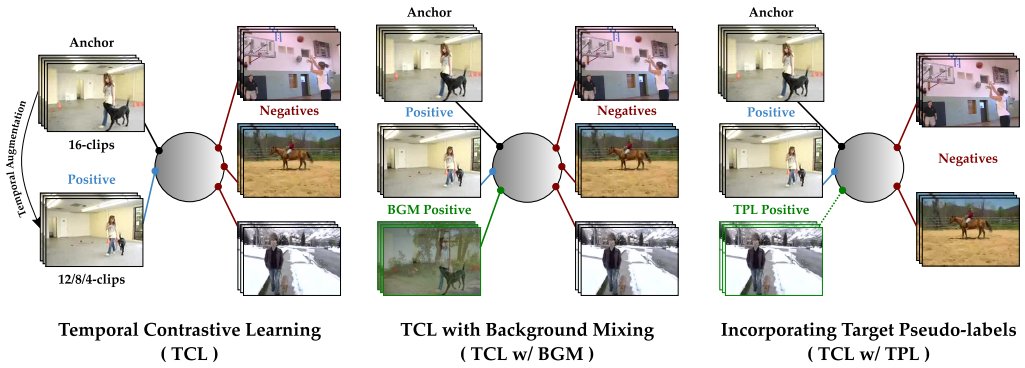

Figure 2: **Temporal Contrastive Learning with Background Mixing and Target Pseudo-labels.** Temporal contrastive loss (left) contrasts a single temporally augmented positive (same video, different speed) per anchor against rest of the videos in a mini-batch as negatives. Incorporating background mixing (middle) provides additional positives per anchor possessing same action semantics with a different background alleviating background shift across domains. Incorporating target pseudo-labels (right) additionally enhances the discriminabilty by contrasting the target videos with the same pseudo-label as positives against rest of the videos as negatives.

- We conduct extensive experiments on several challenging benchmarks (UCF-HMDB [9], Jester [57], and Epic-Kitchens [54]) for video domain adaptation to demonstrate the superiority of our approach over state-of-the-art methods. Our experiments show that `CoMix` delivers a significant performance increase over the compared methods, *e.g.*, `CoMix` outperforms SAVA [12] (ECCV'20) by 3.6% on UCF-HMDB [9] and TA^{3}N [9] (ICCV'19) by 9.2% on Jester [49] benchmark respectively).

2 Related Work

Action Recognition. Much progress has been made in developing a variety of ways to recognize video actions, by either applying 2D-CNNs [6, 39, 51, 84] or 3D-CNNs [4, 18, 23, 78]. Many successful architectures are usually based on the two-stream model [71], processing RGB frames and optical-flow in two separate CNNs with a late fusion in the upper layers [30]. SlowFast network [19] employs two pathways for recognizing actions by processing a video at different frame rates. Mitigating background bias in action recognition has also been presented in [11, 38]. Despite remarkable progress, these models critically depend on large labeled datasets which impose challenges for cross-domain action recognition. In contrast, our work focuses on unsupervised domain adaptation for action recognition, with labeled data in source domain, but only unlabeled data in target domain.

Unsupervised Domain Adaptation. Unsupervised domain adaptation has been studied from multiple perspectives (see reviews [13, 85]). Representative works minimize some measurement of distributional discrepancy [22, 42, 69, 74] or adopt adversarial learning [5, 21, 43, 60, 79] to generate domain-invariant features. Leveraging image translation [26, 27, 55] or style transfer [16, 97] is also another popular trend in domain adaptation. Deep self-training that focus on iteratively training the model using both labeled source data and generated target pseudo labels have been proposed in [50, 96]. Semi-supervised domain adaptation leveraging a few labeled samples from the target domain has also been proposed for many applications [15, 35, 67]. A very few methods have recently attempted video domain adaptation, using adversarial learning combined with temporal attention [9, 45, 57], multi-modal cues [54], and clip order prediction [12]. While existing video DA methods mainly rely on adversarial learning (which is often complicated and hard to train) in some form or other, they do not take any action semantics shared across domains into consideration. Our approach on the other hand, successfully leverages temporal contrastive learning to learn domain-invariant features while exploiting shared action semantics through background mixing for video domain adaptation. Recently, self-supervised tasks like predicting rotation and translation have been used for unsupervised domain adaptation and generalization, mainly for images [3, 41, 75]. By contrast, we focus on the more challenging problem of domain adaptation for human action recognition, where our goal is to align domains by learning consistent features representing different speeds of unlabeled videos. We further propose a temporal supervised contrastive loss to ensure discriminabilty by considering pseudo-labeling in an unified framework for video domain adaptation.

Contrastive Learning. Contrastive representation learning is becoming increasingly attractive due to its great potential to leverage large amount of unlabeled images [10, 17, 24, 52, 25, 56] and videos [20, 33, 58, 64, 63, 83]. Speed of a video is investigated for self-supervised [1, 29, 82, 92]

and semi-supervised learning [72, 100] unlike the problem we consider in this paper. Recent works [90, 93] utilize contrastive learning with different augmentations for learning unsupervised representations of graph data. Contrastive learning has also been recently used in supervised settings, where labels are used to guide the choice of positive and negative pairs [31]. While our approach is inspired by these, we propose a novel temporal contrastive learning framework with background mixing for video domain adaptation, which to our best knowledge has not been explored in the literature.

Image Mixtures. Mixup regularization [95] and its variants [2, 81, 94] that train models on virtual examples constructed as convex combinations of pairs of images and labels have been used to improve the generalization of neural networks. Very few methods apply Mixup in domain adaptation, but mainly to stabilize the domain discriminator [66, 89, 91] or to smoothen the predictions [48]. Several works have recently leveraged the idea of different image mixtures [36, 70] for improving contrastive representation learning. Our proposed background mixing can be regarded as an extension of this line of research by adding background of a video from one domain to a video from another domain, to explore shared semantics while learning domain-invariant features for action recognition.

3 Proposed Method

Unsupervised video domain adaptation aims to improve the model generalization performance by transferring knowledge from a labeled source domain to an unlabeled target domain. Formally, we have a set of labelled source videos $\mathcal{D}_{source} = \{(\mathbf{V}^{i\{s\}}, y^i)\}_{i=1}^{N_S}$ and a set of unlabelled target videos $\mathcal{D}_{target} = \{\mathbf{V}^{i\{t\}}\}_{i=1}^{N_T}$, with a common label space $\mathcal{L}$. Given these data sets, our goal is to learn a single model for action recognition that performs well on previously unseen target domain videos.

Approach Overview. Figure 3 illustrates an overview of `CoMix`. Our action recognition model consists of a feature encoder $\mathcal{F}$ with a temporal graph encoder $\mathcal{G}$. Given a video, the feature encoder $\mathcal{F}$ first extracts clip-level features, and then a graph encoder $\mathcal{G}$ utilizes those features to model intrinsic temporal relations for providing a robust encoded representation for action recognition. `CoMix` adopts supervised learning on the source videos, as the labels are available, jointly with two novel temporal contrastive learning loss terms to align the features for domain adaptation. Specifically, we maximize the similarity of the encoded representation of the fast version of a video (represented by f clips) with that of the slow version of the same video (represented by s clips, where $s < f$) as well as minimize the similarity of the representations of different videos within each of the two domains. However, as temporal contrastive loss treats each domain individually, we further add two new sets of synthetic videos that contain source videos mixed with target background and vice versa, respectively for introducing the background variations among the videos while keeping the action semantics intact. Finally, we generate pseudo-labels for the target videos in every mini-batch and utilize them using another temporal supervised contrastive term. This term contrasts target videos with the same pseudo-label as positives to learn features discriminative for the target domain. We now describe each of our proposed components individually in detail in the following subsections.

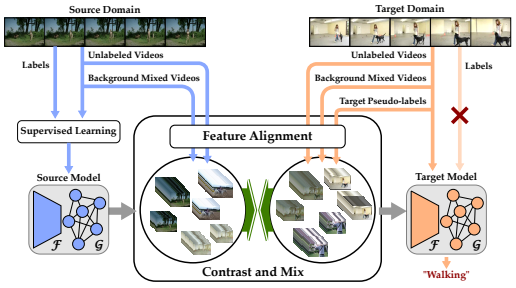

Figure 3: **An Overview of our Approach.** Given labeled videos in source domain and only unlabeled videos in target domain, `CoMix` adopts supervised learning on source videos, jointly with temporal contrastive learning on both domains to align features. Additional cross-domain contrastive supervision is obtained using background mixing across domains and using target pseudo-labels for enhancing discriminability of the latent space. `CoMix` provides a more simpler yet effective approach than adversarial learning for aligning both domains.

Video Representation. Capturing long-range temporal structure in videos is crucial for action recognition, which in turn affects the overall generalization performance of a model when adapting across domains. Thus, we adopt a graph convolutional neural network $(\mathcal{G})$ on top of a 3D convolutional neural network $(\mathcal{F})$ as our video feature encoder. Specifically, for a video $\mathbf{V}$ with n clips, the feature extractor $\mathcal{F}$ maps the clips into the corresponding sequence of features, which alone do not incorporate the rich temporal structure of the video. Therefore, we use the temporal graph encoder which constructs a fully connected graph on top of the clip-level features, with learnable edge weights through a parameterized adjacency matrix, as in [86]. With these graph representations, we apply a

graph convolutional neural network with three layers and finally perform average pooling over all the node features to output the encoded representation of the video $\mathbf{V}$. In summary, the end-to-end network $\mathcal{G}(\mathcal{F}(.)) : \mathbf{V} \rightarrow \mathbb{R}^c$ takes a sequence of clips from a video as input and outputs confidence scores (logits) over the number of classes c for recognizing actions.

Temporal Contrastive Learning. Given video representations, our goal is to leverage contrastive self-supervised learning in both domains for unsupervised domain adaptation. To this end, we use temporal speed invariance in videos as a proxy task and enforce this with a pairwise contrastive loss. Specifically, our key idea is to represent videos in two different temporal speeds (fast and slow) to obtain their encoded representations and then consider the fast and slow version representations of the same video to constitute positive pairs, while versions from different videos constitute negative pairs. Formally, let us consider a mini-batch of B videos $\{\mathbf{V}_n^1, \mathbf{V}_n^2, ..., \mathbf{V}_n^B\}$ with corresponding feature representations $\{\mathbf{z}_n^1, \mathbf{z}_n^2, ..., \mathbf{z}_n^B\}$, where each of the videos $\mathbf{V}_n^i$ is represented using n number of sampled clips. Let f be the number of clips used to represent the fast version of the videos (forwarded through the base branch), and s be that used for the slow version (forwarded through the auxiliary branch), with $s < f$, as shown in Figure 4. Given positive and negative pairs, the model is trained such that it learns to maximize agreement between positive pairs, while minimizing agreement between negative pairs. This is achieved by employing a temporal contrastive loss ($\mathcal{L}_{tcl}$) as

$$\mathcal{L}_{tcl}(\mathbf{V}_f^i, \mathbf{V}_s^i) = -\log \frac{h(\mathbf{z}_f^i, \mathbf{z}_s^i)}{h(\mathbf{z}_f^i, \mathbf{z}_s^i) + \sum\limits_{\substack{j=1, j\neq i \\ v\in\{s,f\}}}^{B} h(\mathbf{z}_f^i, \mathbf{z}_v^j)} \tag{1}$$

where, $h(\mathbf{u}, \mathbf{v}) = \exp(\frac{\mathbf{u}^\intercal \mathbf{v}}{\|\mathbf{u}\|_2 \|\mathbf{v}\|_2}/\tau)$ is the exponential of cosine similarity measure and τ is the temperature hyperparameter [10]. We use $f = 16$, and choose s from $\{12, 8, 4\}$ following a random uniform distribution in every training iteration where randomness encourages the model to learn from a variety of temporal speed variations to learn robust representations.

Background Mixing. As temporal contrastive loss treats each domain individually, it ignores shared action semantics which is vital for domain alignment. Thus, we propose a new perspective of temporal contrastive loss through background mixing, specifically to alleviate the cross-domain background shift, as seen in Figure 1. The basic idea is to obtain the background frames for the videos in one domain and mix it with the frames of the videos from the other domain. More details on how we extract the backgrounds are provided in the appendix. This introduces variation in each of the domains by adding new synthetic videos with the same action semantics as earlier, but possessing background from the other domain. Given two videos $\mathbf{V}^{i\{s\}} \in \mathcal{D}_{source}$ and $\mathbf{V}^{i\{t\}} \in \mathcal{D}_{target}$ with corresponding background frames (single image per video) as $\mathbf{BG}^{i\{s\}}$ and $\mathbf{BG}^{i\{t\}}$, we obtain the synthetic videos in both domains by a convex combination of the background with each of the frames in the videos as follows.

$$\begin{aligned} \hat{\mathbf{V}}^{i\{s\}} &= (1-\lambda) \cdot \mathbf{V}^{i\{s\}} + \lambda \cdot \mathbf{BG}^{i\{t\}} \\ \hat{\mathbf{V}}^{i\{t\}} &= (1-\lambda) \cdot \mathbf{V}^{i\{t\}} + \lambda \cdot \mathbf{BG}^{i\{s\}} \end{aligned} \tag{2}$$

where, λ is sampled from the uniform distribution $[0, \gamma]$, $\hat{\mathbf{V}}^{i\{s\}}$ and $\hat{\mathbf{V}}^{i\{t\}}$ correspond to the video from source domain with target background and vice versa, respectively. The main operation in our proposed background mixing is to generate a synthetic video with background from the other domain while retaining the temporal action semantics intact. Since mixing background doesn't change the motion pattern of a video which actually defines an action, we assume both the original and mixed video to be of the same action class and go beyond single instance positives in Eq. 1 by adding additional positives per anchor, as in supervised

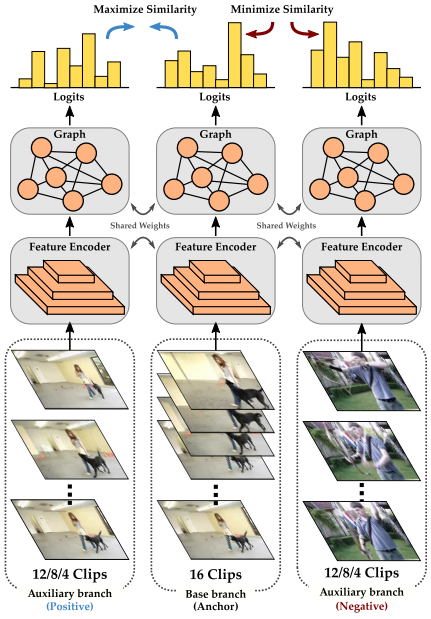

Figure 4: **Temporal Contrastive Loss.** Given unlabeled videos, we maximize similarity between encoded representations of the same video at two different speeds (fast and slow) as well as minimize similarity between different videos played at different speeds.

contrastive learning [31] (see Figure 2 for an illustrative example). The modified temporal contrastive loss with background mixing ($\mathcal{L}_{bgm}$) is defined as below:

$$\mathcal{L}_{bgm}(\mathbf{V}_f^i, \mathbf{V}_s^i) = -\frac{1}{|\mathbf{P}(\mathbf{z}_f^i)|} \sum_{\mathbf{p}\in\mathbf{P}(\mathbf{z}_f^i)} \log \frac{h(\mathbf{z}_f^i, \mathbf{p})}{\sum\limits_{\mathbf{p}\in\mathbf{P}(\mathbf{z}_f^i)} h(\mathbf{z}_f^i, \mathbf{p}) + \sum\limits_{\substack{j=1, j\neq i \\ v\in\{s,f\}}}^{B} \left\{h(\mathbf{z}_f^i, \mathbf{z}_v^j) + h(\mathbf{z}_f^i, \hat{\mathbf{z}}_v^j)\right\}} \tag{3}$$

where, $\mathbf{P}(\mathbf{z}_f^i) \equiv \{\mathbf{z}_s^i, \hat{\mathbf{z}}_s^i, \hat{\mathbf{z}}_f^i\}$ is the set of positives for the anchor $\mathbf{z}_f^i$, and $\hat{\mathbf{z}}_{s/f}^i$ represent the feature representation of the corresponding background-mixed video depending on the domain to which $\mathbf{V}^i$ belongs. Note that for anchor $\mathbf{z}_f^i$, there are 3 positive pairs: (a) slow version of the mixed video ($\hat{\mathbf{z}}_s^i$), (b) fast version of the mixed video ($\hat{\mathbf{z}}_f^i$), and (c) slow version of the original video ($\mathbf{z}_s^i$). Also, the loss is computed for all positive pairs in the mini-batch, i.e., $(\mathbf{V}_f^i, \mathbf{V}_s^i)$, $(\mathbf{V}_s^i, \mathbf{V}_f^i)$, $(\hat{\mathbf{V}}_f^i, \hat{\mathbf{V}}_s^i)$, and $(\hat{\mathbf{V}}_s^i, \hat{\mathbf{V}}_f^i)$. Simultaneous minimization of $\mathcal{L}_{bgm}$ in both source and target domains not only learns temporal dynamics but also helps to better align the features for video domain adaptation by leveraging action semantics shared across both domains. Our background mixing is especially effective in video domain adaptation as it enforces the model to be robust to domain changes (i.e., difference in background as shown in Figure 1) while leaving the action semantics intact. Further, it can also be adopted as a data augmentation strategy for improved generalization in standard video action recognition: we leave this as an interesting future work.

Incorporating Target Pseudo Labels. While temporal contrastive loss with background mixing helps in aligning the learned representations across the two domains, we cannot fully rely on source categories to learn features discriminative for target domain. Therefore, we propose to use a supervised contrastive loss [31] over pseudo-labeled target samples, an extended version of temporal contrastive loss in Eqn. 1 to enhance discriminabilty by allowing many samples per anchor to be positive, so that videos of the same pseudo-label can be attracted to each other in the embedding space. Let A be the subset of videos assigned pseudo-labels using a confidence threshold, from a mini-batch of B videos, the supervised temporal contrastive loss for incorporating target pseudo-labels ($\mathcal{L}_{tpl}$) is defined as

$$\mathcal{L}_{tpl}(\mathbf{V}_f^i, \mathbf{V}_s^i) = -\frac{1}{|\mathbf{P}(\mathbf{z}_f^i)|} \sum_{\mathbf{p}\in\mathbf{P}(\mathbf{z}_f^i)} \log \frac{h(\mathbf{z}_f^i, \mathbf{p})}{\sum\limits_{\mathbf{p}\in\mathbf{P}(\mathbf{z}_f^i)} h(\mathbf{z}_f^i, \mathbf{p}) + \sum\limits_{\substack{a\in A, a\neq i \\ v\in\{s,f\}}} h(\mathbf{z}_f^i, \mathbf{z}_v^a)} \tag{4}$$

where, $\mathbf{P}(\mathbf{z}_f^i) \equiv \{\mathbf{z}_s^p, \mathbf{z}_f^p : p \in A \;\&\; \tilde{y}^p = \tilde{y}^i\} \setminus \{\mathbf{z}_f^i\}$ is the set of all positives for video $\mathbf{V}_f^i$ and $\tilde{y}^i$ represent the pseudo-label for target video $\mathbf{V}^i$. Note that the set of positives ($\mathbf{P}(.)$) includes all the target domain samples (fast and slow) classified as the same action class as that of the anchor ($\mathbf{z}_f^i$) through the pseudo labels. Following [101], we leverage a temporal ensemble prediction for a given video $\mathbf{V}^i$ from the target domain to produce robust and better-calibrated version of pseudo-labels. Specifically, we obtain the encoded (logits) representations $\mathbf{z}_f^i$ and $\mathbf{z}_s^i$ from the base and auxiliary branch respectively and then compute the pseudo-label as $\tilde{y}^i = \arg\max_k \ \text{softmax}(\mathbf{z}_{fused}^i)$, where $\mathbf{z}_{fused}^i$ represents the mean of both logits. We consider the class index k on which the model is most confident among c classes, provided it is higher than a confidence threshold.

Optimization. Besides the losses $\mathcal{L}_{bgm}$ and $\mathcal{L}_{tpl}$, we minimize the standard supervised cross-entropy loss ($\mathcal{L}_{ce}$) on the labelled source videos as follows.

$$\mathcal{L}_{ce}(\mathbf{V}^{i\{s\}}, y^i) = -\sum_{k=1}^{c} (y^i)_k \log(\mathcal{G}(\mathcal{F}(\mathbf{V}^{i\{s\}})))_k \tag{5}$$

Overall, the loss function for training our model involving both source and target domain data is,

$$\mathcal{L}_{CoMix} = \mathcal{L}_{ce}^{\{s\}} + \lambda_{bgm}(\mathcal{L}_{bgm}^{\{s\}} + \mathcal{L}_{bgm}^{\{t\}}) + \lambda_{tpl}\mathcal{L}_{tpl}^{\{t\}} \tag{6}$$

where λ_{bgm} and λ_{tpl} are weights to balance the impact of individual loss terms. To reduce the number of hyper-parameters, we use the same weight λ_{bgm} for both $\mathcal{L}_{bgm}^{\{s\}}$ and $\mathcal{L}_{bgm}^{\{t\}}$. Notably, for the semi-supervised domain adaptation setting, we also use supervised cross-entropy loss for the few labeled target domain videos in addition to the source domain videos.

4 Experiments

Datasets. We evaluate the performance of our approach using several publicly available benchmark datasets for video domain adaptation, namely UCF-HMDB [8], Jester [57], and Epic-Kitchens [54]. UCF-HMDB (assembled by authors in [8]) is an overlapped subset of the original UCF [73] and HMDB datasets [34], containing $3,209$ videos across 12 classes. Jester (assembled by authors in [57]) is a large-scale cross-domain dataset that contains videos of humans performing hand gestures [49] from two domains, namely Source and Target that contain $51,498$ and $51,415$ video clips respectively across 7 classes. Epic-Kitchens (assembled by authors in [54]) is a challenging egocentric dataset that consists of videos across 8 largest action classes from three domains, namely D1, D2 and D3, corresponding to P08, P01 and P22 kitchens on the full Epic-Kitchens dataset [14]. We use the standard training and testing splits provided by the authors in [8, 57, 54] to conduct our experiments on each dataset. More details about the datasets can be found in the appendix.

Baselines. We compare our approach with the following baselines. (1) source only (a lower bound) and supervised target only (an upper bound) baselines that trains the network using labeled source data and labeled target data respectively, (2) popular UDA methods based on adversarial learning (e.g., DANN [21], and ADDA [79]), (3) existing video domain adaptation methods, including SAVA [12], TA^{3}N [9], ABG [45] and TCoN [57]. We also compare with Source + Target (which simply uses all labelled data available to it to train the network) and ENT [67] in semi-supervised domain adaptation experiments. We directly quote the numbers reported in published papers when possible and use source code made publicly available by the authors of TA^{3}N [9] on both Jester and Epic-Kitchens.

Implementation Details. Following [12], we use I3D [4] as the backbone feature encoder network, initialized with Kinetics pre-trained weights. For the temporal graph encoder, we use a 3-layer GCN similar to [86]. We follow the standard 'pre-train then adapt' procedure used in prior works [79, 12] and train the model with only source data to provide a warmstart before the proposed approach is employed. The dimension of the features extracted from the I3D encoder is 1024 which is the same as the node-feature dimension of the initial layer of the GCN. The final layer of the GCN has its node-feature dimension same as the number of action classes in a dataset and uses a mean aggregation strategy to output the logits. We use a clip-length of 8-frames and train all the models end-to-end using SGD with a momentum of 0.9 and a weight decay of 1e-7. We use an initial learning rate of 0.001 for the I3D and 0.01 for the GCN in all our experiments. We use a batch size of 40 equally split over the two domains, where each batch consists of n clips from the same video, where n is 16 for the fast version (f) and 12, 8, or 4 for the slow version (s). For inference, we use 16 uniformly sampled clips per video and use the base branch of the model to recognize the action. The temperature parameter is set to $\tau = 0.5$. We extract backgrounds from videos using temporal median filtering [62] and empirically set $\gamma = 0.5$ for background mixing. We use a pseudo-label threshold of 0.7 in all our experiments and smooth the cross-entropy loss with $\epsilon = 0.1$, following [76, 53]. We set λ_{bgm} and λ_{tpl} from $\{0.01, 0.1\}$ depending on the dataset. We report the average action recognition accuracy over 3 random trials. We use 6 NVIDIA Tesla V100 GPUs for training all our models.

Results on UCF-HMDB. Table 1 shows results of our method and other competing approaches on UCF-HMDB dataset. Our `CoMix` framework achieves the best average performance of $\mathbf{90.3\%}$, which is about $\mathbf{2.2\%}$ more than the previous state-of-the-art performance on this dataset. While comparing with the recent method, SAVA [12] using the same I3D backbone, `CoMix` obtains $\mathbf{4.5\%}$ and $\mathbf{2.7\%}$ improvement on UCF→HMDB and HMDB→UCF task respectively, without relying on frame attention or adversarial learning. These improvements clearly show that our temporal graph contrastive learning with background mixing is not only able to better leverage the temporal information but also shared action semantics, essential for effective video domain adaptation. In summary, `CoMix` outperforms all the existing video

Table 1: **Results on UCF-HMDB Dataset.** `CoMix` establishes new state-of-the-art for unsupervised video domain adaptation on UCF-HMDB, by significantly outperforming existing methods.

Method	Backbone	UCF→HMDB	HMDB→UCF	Average
DANN [21]	ResNet-101	75.3	76.4	75.8
JAN [44]	ResNet-101	74.7	79.3	77.0
AdaBN [37]	ResNet-101	75.5	77.4	76.4
MCD [68]	ResNet-101	74.4	79.3	76.8
TA^{3}N [9]	ResNet-101	78.3	81.8	80.1
ABG [45]	ResNet-101	79.1	85.1	82.1
TCoN [57]	ResNet-101	87.2	89.1	88.1
Source Only	I3D	80.3	88.8	84.5
DANN [21]	I3D	80.7	88.0	84.3
ADDA [79]	I3D	79.1	88.4	83.7
TA^{3}N [9]	I3D	81.4	90.5	85.9
SAVA [12]	I3D	82.2	91.2	86.7
`CoMix`	I3D	**86.7**	**93.9**	**90.3**
Supervised Target	I3D	95.0	96.8	95.9

Table 2: **Results on Jester and Epic-Kitchens Datasets.** CoMix outperforms TA^{3}N [9] by 9.2% on the challenging Jester dataset. On Epic-Kitchens, CoMix achieves the best performance on 5 out of 6 transfer tasks including the best average performance among all compared methods.

Method	Backbone	Jester	Epic-Kitchens						Average
		Source→Target	D2→D1	D3→D1	D1→D2	D3→D2	D1→D3	D2→D3	
Source Only	I3D	51.5	35.4	34.6	32.8	35.8	34.1	39.1	35.3
DANN [21]	I3D	55.4	38.3	38.8	37.7	42.1	36.6	41.9	39.2
ADDA [79]	I3D	52.3	36.3	36.1	35.4	41.4	34.9	40.8	37.4
TA^{3}N [9]	I3D	55.5	**40.9**	39.9	34.2	44.2	37.4	42.8	39.9
CoMix	I3D	**64.7**	38.6	**42.3**	**42.9**	**49.2**	**40.9**	**45.2**	**43.2**
Supervised Target	I3D	95.6	57.0	57.0	64.0	64.0	63.7	63.7	61.5

DA methods on UCF-HMDB, showing the efficacy of our approach in learning more transferable features for cross-domain action recognition without using any target labels.

Results on Jester and Epic-Kitchens. On the large-scale Jester dataset, our proposed approach, CoMix also outperforms other DA approaches by increasing the Source Only (no adaptation) accuracy from **51.5%** to **64.7%**, as shown in Table 2 (left). In particular, our approach achieves an absolute improvement of **9.2%** over TA^{3}N [9], which corroborates the fact that CoMix can well handle not only the appearance gap but also the action gap present on this dataset (e.g., for the action class "rolling hand", source domain contains videos of "rolling hand forward", while the target domain only consists of videos of "rolling hand backward"). Table 2 (right) summarizes the results on Epic-Kitchens, which is another challenging dataset consisting of total 6 transfer tasks with a large imbalance across different action classes. Overall, CoMix obtains the best on 5 tasks including the best average performance of **43.2%**, compared to only **35.3%** and **39.9%** achieved by the source only and TA^{3}N [9] respectively. While the improvements achieved by our approach are encouraging on both Jester and Epic-Kitchens, the accuracy gap between CoMix and supervised target is still significant (**30.9%** on Jester and **18.3%** on Epic-Kitchens), which highlights the great potential for improvement in future for unsupervised video domain adaptation.

Comparison with MM-SADA [54]. MM-SADA[54] is another state-of-the-art approach for video domain adaptation that leverages the idea of using multi-modal (RGB and Optical flow) data to learn better domain invariant representations. The approach has two main components: adversarial learning and multi-modal supervision. While CoMix does not use optical flow features anywhere, the RGB-only version of MM-SADA still uses optical flow features for the multi-modal self-supervision. Interestingly, CoMix (43.2%) shows very competitive performance using only RGB features when compared to the above (43.9%) on the Epic-Kitchens dataset. Additionally, we train MM-SADA (RGB-only) (but perform multimodal supervision using both RGB and flow following the original paper [54]) on UCF-HMDB dataset and notice that CoMix outperforms it by a margin of 3% on an average (UCF → HMDB: 82.2% vs 86.7%, HMDB → UCF: 91.2% vs 93.9%, Avg: 86.7% vs 90.3%), showing its effectiveness in unsupervised video domain adaptation.

Semi-supervised Domain Adaptation. To further study the robustness of our proposed approach, we extend the unsupervised domain adaptation to a semi-supervised setting, where one (1-shot) and three target labels (3-shot) per class are available for training. Table 3 shows that our simple approach consistently outperforms the adversarial DA methods (DANN [21], and ADDA [79]) including the semi-supervised method, ENT [67], on both UCF-HMDB and Jester datasets. Remarkably, CoMix with three target labels per class improves the performance of Source + Target baseline from **93.7%** to **96.6%**, which is only **0.2%** lower than the supervised target upper bound (in Table 1) on HMDB→UCF task (**96.6%** vs **96.8%**). These results well demonstrate the utility of our proposed approach in many practical applications where annotating *a few* videos per class is typically possible and therefore worth doing given the boost it provides.

Table 3: **Semi-Supervised Domain Adaptation on UCF-HMDB and Jester Datasets.** CoMix significantly outperforms all the compared methods in both one-shot and three-shot settings.

Method	UCF→HMDB		HMDB→UCF		Jester(S) → Jester(T)	
	1-shot	3-shot	1-shot	3-shot	1-shot	3-shot
Source + Target	83.2	85.8	90.3	93.7	53.8	55.0
DANN [21]	85.4	86.9	92.1	93.1	55.1	59.9
ADDA [79]	83.6	86.3	91.2	93.0	59.5	61.3
ENT [67]	85.6	88.6	92.8	95.8	58.6	61.5
CoMix	**88.4**	**93.1**	**95.4**	**96.6**	**65.3**	**69.6**

Effectiveness of Individual Components. As seen from Table 4, the vanilla temporal contrastive learning (TCL) achieves an average accuracy of 85.8% on UCF-HMDB while 57.5% on Jester (1st row), which is already better than DANN [21], and ADDA [79] (ref. Table 1,2), showing its

effectiveness over adversarial learning in aligning features. While both background mixing (BGM) and incorporation of target pseudo-labels (TPL) individually improves the performance over TCL ($+\mathbf{2.9\%}$, $+\mathbf{5.6\%}$ using BGM and $+\mathbf{1.9\%}$, $+\mathbf{5.4\%}$ using TPL, respectively), addition of both of them leads to the best average performance of 90.3% on UCF-HMDB dataset and 64.7% on the Jester dataset. This corroborates the fact that both cross-domain action semantics (through BGM) and discriminabilty (through TPL) of the latent space play crucial roles in video domain adaptation in addition to the vanilla contrastive learning for aligning features.

Table 4: **Ablation Study on UCF-HMDB and Jester.** TCL: Temporal Contrastive Learning, BGM: Background Mixing, TPL: Target Pseudo-Labels.

TCL	BGM	TPL	U→H	H→U	Average	Jester(S)→Jester(R)
✓	✗	✗	83.3	88.4	85.8	57.5
✓	✓	✗	86.2	91.2	88.7	63.1
✓	✗	✓	83.5	91.9	87.7	62.9
✓	✓	✓	**86.7**	**93.9**	**90.3**	**64.7**

Table 5: **Comparison with MixUp Strategies**. Background mixing outperforms other alternatives in leveraging shared action semantics on UCF-HMDB.

Method	U→H	H→U	Average	Jester(S)→Jester(R)
Gaussian Noise	84.7	90.6	87.6	54.3
Video MixUp	85.1	91.7	88.4	62.2
Video CutMix	84.6	92.1	88.3	58.6
Background Mixing	**86.7**	**93.9**	**90.3**	**64.7**

Comparison with Different MixUp Strategies. We explore the effectiveness of background mixing by comparing with different MixUp strategies (Table 5): (a) Gaussian Noise: adding White Gaussian Noise to videos in both domains; (b) Video MixUp [95]: directly mixing one video with another from a different domain, as in images; (c) Video CutMix [94]: randomly replacing a region of a video with another region from the other domain. The proposed way of generating synthetic videos by mixing background of a video from one domain to a video from another domain, outperforms all three alternatives on UCF-HMDB as well as on the more challenging Jester dataset. Note that while both MixUp and CutMix destroy motion pattern of original video, background mixing keeps semantic consistency without changing the temporal dynamics.

Effect of Background Pseudo-labels. We investigate the effect of pseudo-labels on background mixed videos (i.e., both videos considered to be of same action class while creating positives) by simply adding them as unlabeled videos without any modification to the contrastive objective in Eq. 1. `CoMix` without background pseudo-labels decreases the performance from 90.3% to 89.0% ($-\mathbf{1.3\%}$: Table 6), showing its effectiveness in leveraging action semantics shared across both domains.

Effect of Source Contrastive Learning. `CoMix` adopts contrastive learning on both source and target domains, although we already have supervised cross-entropy loss on source videos. We observe that applying contrastive learning on target domain only, by removing source contrastive objective $\mathcal{L}_{bgm}^{\{s\}}$ from Eq. 6, lowers down the performance from 90.3% to 88.4% ($-\mathbf{1.9\%}$) on UCF-HMDB (Table 6). This shows the importance of training the model using the same temporal invariance objective on both domains simultaneously to achieve effective alignment across domains.

Effect of Random Speed Invariance. We remove randomness in video speed from the auxiliary branch of our temporal contrastive learning framework and observe that `CoMix` (with 16 clips in the base branch and only 8 clips in the auxiliary branch) leads to an average top-1 accuracy of 89.6% compared to 90.3% ($-\mathbf{0.7\%}$: Table 6), showing the importance of random speed invariance in learning robust features.

Table 6: **Ablation Study on Contrastive Learning.**

Method	U→H	H→U	Average
`CoMix`	**86.7**	**93.7**	**90.3**
– w/o Background Pseudo-labels	85.8	92.2	89.0
– w/o Source Contrastive Learning	85.1	91.8	88.4
– w/o Random Speed Invariance	86.4	92.8	89.6

Self-Training vs Supervised Contrastive Learning. We directly use self-training that uses cross-entropy loss on target pseudo labels instead of $\mathcal{L}_{tpl}^{\{t\}}$ and find that the average performance drops to 88.7% on UCF-HMDB, indicating the advantage of supervised contrastive objective in enhancing discriminability of the latent space by successfully leveraging label information from target domain.

Effect of Graph Representation. (a) *Removal of Graph Representation from* `CoMix`: We examine the effect of graph representation for videos and find that by removing GCN from our framework lowers down the performance from 90.3% to 88.1% on UCF-HMDB dataset, which shows that graph contrastive learning is more useful in capturing the temporal dependencies, essential for video domain adaptation. (b) *Effect of Graph*

Table 7: **Baseline Comparisons w/ GCN Representations on UCF-HMDB and Jester Datasets.**

Method (w/ GCN)	U→H	H→U	Average	Jester(S)→Jester(R)
Source Only	82.5	87.7	85.1	54.0
DANN [21]	80.0	86.3	83.2	62.9
TA^{3}N [9]	52.5	72.4	62.3	51.7
`CoMix`	**86.7**	**93.9**	**90.3**	**64.7**

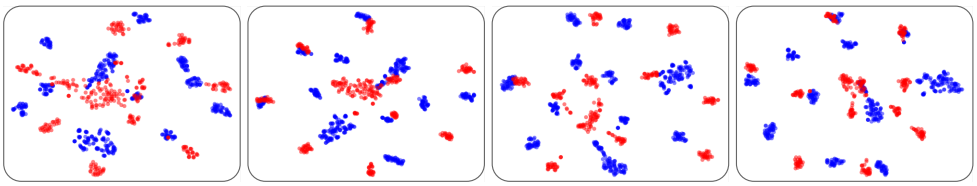

Figure 5: **Feature Visualizations using t-SNE.** Plots show visualization of our approach with different components on UCF→HMDB task. Blue and red dots represent source and target data respectively. Features for both target and source domain become progressively discriminative and improve from left to right by adoption of our novel components within a temporal contrastive learning framework. Best viewed in color.

Representation on Baseline Methods: Additionally, in Table 7 we compare with domain adversarial adaptation methods DANN [21] and TA3N [9] including the Source only baseline with GCN feature representation on both UCF-HMDB and Jester datasets. `CoMix` improves the Source only accuracy by 5.2% and 10.7% respectively on UCF-HMDB and Jester datasets. Furthermore, `CoMix` outperforms DANN [21] with the same GCN equipped as ours, on both datasets (+7.1%, +1.8%, respectively) showing its effectiveness over adversarial learning in aligning features for video domain adaptation. TA3N [9] performs very poorly (62.3% and 51.7%) when equipped additionally with graph representations. We believe this is because TA3N already utilizes Temporal Relational Network [98] for modeling temporal relations, which probably hinders in learning GCN features for successful domain adaptation in videos. (c) *Alternatives for Graph Representation*: We replace GCN using MLP/LSTM of similar complexity and notice that both alternatives are inferior to GCN on UCF-HMDB (MLP: 88.1%, LSTM: 84.3%, GCN: 90.3%), which shows the effectiveness of GCN in our contrastive learning framework for capturing the temporal dependencies, essential for video domain adaptation.

Effect of Background Extraction Method. We experiment with a different background extraction strategy [99] that uses Gaussian Mixture Models (GMM) to extract the backgrounds and observe that the very simple and fast strategy based on temporal median filtering [62] outperforms GMM by 2.3% on average on UCF-HMDB (UCF→HMDB: 85.3% vs 86.7%, HMDB→UCF: 90.7% vs 93.9%, Avg: 88.0% vs 90.3%). Note that our `CoMix` framework is agnostic to the method used for background extraction and can be incorporated with any other background extraction techniques for videos, *e.g.*, learnable background segmentation strategies such as [87, 59].

Feature Visualizations. We use t-SNE [46] to visualize the features learned using different components of our `CoMix` framework. As seen from Figure 5, alignment of domains including discriminability improves as we adopt "TCL" and "BGM" to the vanilla Source only model. The best results are obtained when all the three components "TCL", "BGM" and "TPL" i.e., `CoMix` are added and trained using an unified framework (Eq. 6) for unsupervised video domain adaptation. Additional results and analysis including more qualitative examples are included in the appendix.

5 Conclusions

In this paper, we introduce a new end-to-end temporal contrastive learning framework to bridge the domain gap by learning consistent features representing two different speeds of the unlabeled videos. We also propose two novel extension to temporal contrastive loss by using background mixing and target pseudo-labels, that allows additional positive(s) per anchor, thus adapting contrastive learning to leverage cross-domain action semantics and label information from the target domain respectively in an unified framework, for learning discriminative invariant features. We demonstrate the effectiveness of our approach on three standard datasets, outperforming several competing methods.

Broader Impact. Our research can help reduce burden of collecting large-scale supervised data in many real-world applications of human action recognition by transferring knowledge from auxiliary datasets. The positive impact that our work could have on society is in making technology more accessible for institutions and individuals that do not have rich resources for collecting and annotating large-scale video datasets. Negative impacts of our research are difficult to predict, however, it shares many of the pitfalls associated with standard deep learning models such as susceptibility to adversarial attacks and lack of interpretablity. Other adverse effects could be potential attrition in jobs in certain sectors of economy where fewer employees (security guards, nurses, etc.) are needed to monitor human activities as a result of wider adoption of automated video recognition systems.

Acknowledgements. This work was partially supported by the ISIRD Grant EEE.

References

[1] Sagie Benaim, Ariel Ephrat, Oran Lang, Inbar Mosseri, William T Freeman, Michael Rubinstein, Michal Irani, and Tali Dekel. Speednet: Learning the Speediness in Videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9922–9931, 2020.

[2] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A Raffel. Mixmatch: A Holistic Approach to Semi-Supervised Learning. In *Advances in Neural Information Processing Systems*, pages 5049–5059, 2019.

[3] Fabio M Carlucci, Antonio D'Innocente, Silvia Bucci, Barbara Caputo, and Tatiana Tommasi. Domain Generalization by Solving Jigsaw Puzzles. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2229–2238, 2019.

[4] Joao Carreira and Andrew Zisserman. Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308, 2017.

[5] Chao Chen, Zhihong Chen, Boyuan Jiang, and Xinyu Jin. Joint Domain Alignment and Discriminative Feature Learning for Unsupervised Deep Domain Adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 3296–3303, 2019.

[6] Chun-Fu Richard Chen, Rameswar Panda, Kandan Ramakrishnan, Rogerio Feris, John Cohn, Aude Oliva, and Quanfu Fan. Deep Analysis of CNN-Based Spatio-Temporal Representations for Action Recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6165–6175, June 2021.

[7] Chun-Fu Richard Chen, Rameswar Panda, Kandan Ramakrishnan, Rogerio Feris, John Cohn, Aude Oliva, and Quanfu Fan. Deep analysis of cnn-based spatio-temporal representations for action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6165–6175, 2021.

[8] Jin Chen, Xinxiao Wu, Lixin Duan, and Shenghua Gao. Domain Adversarial Reinforcement Learning for Partial Domain Adaptation. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–15, 2020.

[9] Min-Hung Chen, Zsolt Kira, Ghassan AlRegib, Jaekwon Yoo, Ruxin Chen, and Jian Zheng. Temporal Attentive Alignment for Large-Scale Video Domain Adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6321–6330, 2019.

[10] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A Simple Framework for Contrastive Learning of Visual Representations. In *International conference on machine learning*, pages 1597–1607, 2020.

[11] Jinwoo Choi, Chen Gao, Joseph CE Messou, and Jia-Bin Huang. Why Can't I Dance in the Mall? Learning to Mitigate Scene Bias in Action Recognition. *Advances in Neural Information Processing Systems*, 32:853–865, 2019.

[12] Jinwoo Choi, Gaurav Sharma, Samuel Schulter, and Jia-Bin Huang. Shuffle and Attend: Video Domain Adaptation. In *European Conference on Computer Vision*, pages 678–695, 2020.

[13] Gabriela Csurka. *A Comprehensive Survey on Domain Adaptation for Visual Applications*, pages 1–35. Springer International Publishing, Cham, 2017.

[14] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, et al. Scaling Egocentric Vision: The Epic-Kitchens Dataset. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 720–736, 2018.

[15] Hal Daumé III, Abhishek Kumar, and Avishek Saha. Frustratingly Easy Semi-Supervised Domain Adaptation. In *Proceedings of the 2010 Workshop on Domain Adaptation for Natural Language Processing*, pages 53–59, 2010.

[16] Aysegul Dundar, Ming-Yu Liu, Ting-Chun Wang, John Zedlewski, and Jan Kautz. Domain Stylization: A Strong, Simple Baseline for Synthetic to Real Image Domain Adaptation. *arXiv preprint arXiv:1807.09384*, 2018.

[17] Debidatta Dwibedi, Yusuf Aytar, Jonathan Tompson, Pierre Sermanet, and Andrew Zisserman. With a Little Help From My Friends: Nearest-Neighbor Contrastive Learning of Visual Representations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9588–9597, October 2021.

[18] Christoph Feichtenhofer. X3d: Expanding Architectures for Efficient Video Recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 203–213, 2020.

[19] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast Networks for Video Recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6202–6211, 2019.

[20] Christoph Feichtenhofer, Haoqi Fan, Bo Xiong, Ross Girshick, and Kaiming He. A Large-Scale Study on Unsupervised Spatiotemporal Representation Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3299–3309, 2021.

[21] 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*, 17(1):2096–2030, 2016.

[22] Arthur Gretton, Karsten M Borgwardt, Malte J Rasch, Bernhard Schölkopf, and Alexander Smola. A Kernel Two-Sample Test. *The Journal of Machine Learning Research*, 13(1):723–773, 2012.

[23] Kensho Hara, Hirokatsu Kataoka, and Yutaka Satoh. Learning Spatio-Temporal Features with 3D Residual Networks for Action Recognition. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 3154–3160, 2017.

[24] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum Contrast for Unsupervised Visual Representation Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9729–9738, 2020.

[25] R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. Learning Deep Representations by Mutual Information Estimation and Maximization. In *International Conference on Learning Representations*, 2018.

[26] Judy Hoffman, Eric Tzeng, Taesung Park, Jun-Yan Zhu, Phillip Isola, Kate Saenko, Alexei Efros, and Trevor Darrell. Cycada: Cycle-Consistent Adversarial Domain Adaptation. In *International conference on machine learning*, pages 1989–1998. PMLR, 2018.

[27] Lanqing Hu, Meina Kan, Shiguang Shan, and Xilin Chen. Duplex Generative Adversarial Network for Unsupervised Domain Adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1498–1507, 2018.

[28] Arshad Jamal, Vinay P Namboodiri, Dipti Deodhare, and KS Venkatesh. Deep Domain Adaptation in Action Space. In *BMVC*, volume 2, page 4, 2018.

[29] Simon Jenni, Givi Meishvili, and Paolo Favaro. Video Representation Learning by Recognizing Temporal Transformations. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 425–442. Springer, 2020.

[30] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. Large-Scale Video Classification with Convolutional Neural Networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 1725–1732, 2014.

[31] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised Contrastive Learning. *Advances in Neural Information Processing Systems*, 33, 2020.

[32] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

[33] Bruno Korbar, Du Tran, and Lorenzo Torresani. Cooperative Learning of Audio and Video Models From Self-Supervised Synchronization. *arXiv preprint arXiv:1807.00230*, 2018.

[34] Hildegard Kuehne, Hueihan Jhuang, Estíbaliz Garrote, Tomaso Poggio, and Thomas Serre. HMDB: A Large Video Database for Human Motion Recognition. In *2011 International conference on computer vision*, pages 2556–2563. IEEE, 2011.

[35] Abhishek Kumar, Avishek Saha, and Hal Daume. Co-Regularization Based Semi-Supervised Domain Adaptation. *Advances in neural information processing systems*, 23:478–486, 2010.

[36] Kibok Lee, Yian Zhu, Kihyuk Sohn, Chun-Liang Li, Jinwoo Shin, and Honglak Lee. I-Mix: A Domain-Agnostic Strategy for Contrastive Representation Learning. *arXiv preprint arXiv:2010.08887*, 2021.

[37] Yanghao Li, Naiyan Wang, Jianping Shi, Xiaodi Hou, and Jiaying Liu. Adaptive Batch Normalization for Practical Domain Adaptation. *Pattern Recognition*, 80:109–117, 2018.

[38] Yingwei Li, Yi Li, and Nuno Vasconcelos. Resound: Towards Action Recognition Without Representation Bias. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 513–528, 2018.

[39] Ji Lin, Chuang Gan, and Song Han. Temporal Shift Module for Efficient Video Understanding. In *CVF International Conference on Computer Vision (ICCV)*, pages 7082–7092, 2019.

[40] Wei Liu, Yuanzheng Cai, Miaohui Zhang, Hui Li, and Hejin Gu. Scene background estimation based on temporal median filter with gaussian filtering. In *2016 23rd International Conference on Pattern Recognition (ICPR)*, pages 132–136. IEEE, 2016.

[41] Weizhe Liu, David Ferstl, Samuel Schulter, Lukas Zebedin, Pascal Fua, and Christian Leistner. Domain Adaptation for Semantic Segmentation via Patch-Wise Contrastive Learning. *arXiv preprint arXiv:2104.11056*, 2021.

[42] Mingsheng Long, Yue Cao, Jianmin Wang, and Michael Jordan. Learning Transferable Features with Deep Adaptation Networks. In *International conference on machine learning*, pages 97–105, 2015.

[43] Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. Conditional Adversarial Domain Adaptation. In *Advances in Neural Information Processing Systems*, pages 1640–1650, 2018.

[44] Mingsheng Long, Han Zhu, Jianmin Wang, and Michael I Jordan. Deep Transfer Learning With Joint Adaptation Networks. In *International conference on machine learning*, pages 2208–2217. PMLR, 2017.

[45] Yadan Luo, Zi Huang, Zijian Wang, Zheng Zhang, and Mahsa Baktashmotlagh. Adversarial Bipartite Graph Learning for Video Domain Adaptation. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 19–27, 2020.

[46] Laurens van der Maaten and Geoffrey Hinton. Visualizing Data Using t-SNE. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.

[47] Murari Mandal, Lav Kush Kumar, Mahipal Singh Saran, et al. Motionrec: A unified deep framework for moving object recognition. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2734–2743, 2020.

[48] Xudong Mao, Yun Ma, Zhenguo Yang, Yangbin Chen, and Qing Li. Virtual Mixup Training for Unsupervised Domain Adaptation. *arXiv preprint arXiv:1905.04215*, 2019.

[49] Joanna Materzynska, Guillaume Berger, Ingo Bax, and Roland Memisevic. The Jester Dataset: A Large-Scale Video Dataset of Human Gestures. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 0–0, 2019.

[50] Ke Mei, Chuang Zhu, Jiaqi Zou, and Shanghang Zhang. Instance Adaptive Self-Training for Unsupervised Domain Adaptation. In *European Conference on Computer Vision*. Springer, 2020.

[51] Yue Meng, Chung-Ching Lin, Rameswar Panda, Prasanna Sattigeri, Leonid Karlinsky, Aude Oliva, Kate Saenko, and Rogerio Feris. AR-Net: Adaptive Frame Resolution for Efficient Action Recognition. In *European Conference on Computer Vision*, pages 86–104, 2020.

[52] Ishan Misra and Laurens van der Maaten. Self-Supervised Learning of Pretext-Invariant Representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6707–6717, 2020.

[53] Rafael Müller, Simon Kornblith, and Geoffrey E Hinton. When Does Label Smoothing Help? In *Advances in Neural Information Processing Systems*, pages 4694–4703, 2019.

[54] Jonathan Munro and Dima Damen. Multi-Modal Domain Adaptation for Fine-Grained Action Recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 122–132, 2020.

[55] Zak Murez, Soheil Kolouri, David Kriegman, Ravi Ramamoorthi, and Kyungnam Kim. Image to Image Translation for Domain Adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4500–4509, 2018.

[56] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation Learning With Contrastive Predictive Coding. *arXiv preprint arXiv:1807.03748*, 2018.

[57] Boxiao Pan, Zhangjie Cao, Ehsan Adeli, and Juan Carlos Niebles. Adversarial Cross-Domain Action Recognition With Co-Attention. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 11815–11822, 2020.

[58] Tian Pan, Yibing Song, Tianyu Yang, Wenhao Jiang, and Wei Liu. Videomoco: Contrastive Video Representation Learning With Temporally Adversarial Examples. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11205–11214, 2021.

[59] Prashant W Patil, Akshay Dudhane, and Subrahmanyam Murala. Multi-frame Recurrent Adversarial Network for Moving Object Segmentation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2302–2311, 2021.

[60] Zhongyi Pei, Zhangjie Cao, Mingsheng Long, and Jianmin Wang. Multi-Adversarial Domain Adaptation. *Thirty-second AAAI conference on artificial intelligence*, 2018.

[61] Xingchao Peng, Ben Usman, Neela Kaushik, Judy Hoffman, Dequan Wang, and Kate Saenko. VisDA: The Visual Domain Adaptation Challenge. *arXiv preprint arXiv:1710.06924*, 2017.

[62] Massimo Piccardi. Background Subtraction Techniques: A Review. In *2004 IEEE International Conference on Systems, Man and Cybernetics*, volume 4, pages 3099–3104. IEEE, 2004.

[63] Rui Qian, Tianjian Meng, Boqing Gong, Ming-Hsuan Yang, Huisheng Wang, Serge Belongie, and Yin Cui. Spatiotemporal Contrastive Video Representation Learning. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6964–6974, 2021.

[64] Adrià Recasens, Pauline Luc, Jean-Baptiste Alayrac, Luyu Wang, Florian Strub, Corentin Tallec, Mateusz Malinowski, Viorica Patraucean, Florent Altché, Michal Valko, et al. Broaden Your Views for Self-Supervised Video Learning. *arXiv preprint arXiv:2103.16559*, 2021.

[65] Kate Saenko, Brian Kulis, Mario Fritz, and Trevor Darrell. Adapting Visual Category Models to New Domains. In *European conference on computer vision*, pages 213–226. Springer, 2010.

[66] Aadarsh Sahoo, Rameswar Panda, Rogerio Feris, Kate Saenko, and Abir Das. Select, Label, and Mix: Learning Discriminative Invariant Feature Representations for Partial Domain Adaptation. *arXiv preprint arXiv:2012.03358*, 2020.

[67] Kuniaki Saito, Donghyun Kim, Stan Sclaroff, Trevor Darrell, and Kate Saenko. Semi-Supervised Domain Adaptation via Minimax Entropy. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8050–8058, 2019.

[68] Kuniaki Saito, Kohei Watanabe, Yoshitaka Ushiku, and Tatsuya Harada. Maximum Classifier Discrepancy for Unsupervised Domain Adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3723–3732, 2018.

[69] Jian Shen, Yanru Qu, Weinan Zhang, and Yong Yu. Wasserstein Distance Guided Representation Learning for Domain Adaptation. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[70] Zhiqiang Shen, Zechun Liu, Zhuang Liu, Marios Savvides, Trevor Darrell, and Eric Xing. Un-Mix: Rethinking Image Mixtures for Unsupervised Visual Representation Learning. *arXiv preprint arXiv:2003.05438*, 3(7), 2020.

[71] Karen Simonyan and Andrew Zisserman. Two-Stream Convolutional Networks for Action Recognition in Videos. In *Advances in Neural Information Processing Systems*, pages 568–576, 2014.

[72] Ankit Singh, Omprakash Chakraborty, Ashutosh Varshney, Rameswar Panda, Rogerio Feris, Kate Saenko, and Abir Das. Semi-Supervised Action Recognition with Temporal Contrastive Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10389–10399, 2021.

[73] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. UCF101: A Dataset of 101 Human Actions Classes From Videos in the Wild. *arXiv preprint arXiv:1212.0402*, 2012.

[74] Baochen Sun and Kate Saenko. Deep Coral: Correlation Alignment for Deep Domain Adaptation. In *European conference on computer vision*, pages 443–450. Springer, 2016.

[75] Yu Sun, Eric Tzeng, Trevor Darrell, and Alexei A Efros. Unsupervised Domain Adaptation Through Self-Supervision. *arXiv preprint arXiv:1909.11825*, 2019.

[76] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the Inception Architecture for Computer Vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016.

[77] Ozan Tezcan, Prakash Ishwar, and Janusz Konrad. Bsuv-net: A fully-convolutional neural network for background subtraction of unseen videos. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2774–2783, 2020.

[78] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning Spatiotemporal Features With 3D Convolutional Networks. In *Proceedings of the IEEE international conference on computer vision*, pages 4489–4497, 2015.

[79] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial Discriminative Domain Adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7167–7176, 2017.

[80] Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep Hashing Network for Unsupervised Domain Adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5018–5027, 2017.

[81] Vikas Verma, Alex Lamb, Christopher Beckham, Amir Najafi, Ioannis Mitliagkas, David Lopez-Paz, and Yoshua Bengio. Manifold Mixup: Better Representations by Interpolating Hidden States. In *International Conference on Machine Learning*, pages 6438–6447. PMLR, 2019.

[82] Jiangliu Wang, Jianbo Jiao, and Yun-Hui Liu. Self-Supervised Video Representation Learning by Pace Prediction. In *European Conference on Computer Vision*, pages 504–521, 2020.

[83] Jinpeng Wang, Yuting Gao, Ke Li, Yiqi Lin, Andy J Ma, Hao Cheng, Pai Peng, Feiyue Huang, Rongrong Ji, and Xing Sun. Removing the Background by Adding the Background: Towards Background Robust Self-supervised Video Representation Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11804–11813, 2021.

[84] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal Segment Networks: Towards Good Practices for Deep Action Recognition. In *European conference on computer vision*, pages 20–36, 2016.

[85] Mei Wang and Weihong Deng. Deep Visual Domain Adaptation: A Survey. *Neurocomputing*, 312:135–153, 2018.

[86] Xiaolong Wang and Abhinav Gupta. Videos as Space-Time Region Graphs. In *Proceedings of the European conference on computer vision (ECCV)*, pages 399–417, 2018.

[87] Xueying Wang, Lei Liu, Guangli Li, Xiao Dong, Peng Zhao, and Xiaobing Feng. Background Subtraction on Depth Videos With Convolutional Neural Networks. In *2018 International Joint Conference on Neural Networks (IJCNN)*, pages 1–7. IEEE, 2018.

[88] Felix Wu, Amauri Souza, Tianyi Zhang, Christopher Fifty, Tao Yu, and Kilian Weinberger. Simplifying graph convolutional networks. In *International conference on machine learning*, pages 6861–6871. PMLR, 2019.

[89] Yuan Wu, Diana Inkpen, and Ahmed El-Roby. Dual Mixup Regularized Learning for Adversarial Domain Adaptation. In *European Conference on Computer Vision*, pages 540–555. Springer, 2020.

[90] Yaochen Xie, Zhao Xu, Zhengyang Wang, and Shuiwang Ji. Self-Supervised Learning of Graph Neural Networks: A Unified Review. *arXiv preprint arXiv:2102.10757*, 2021.

[91] Minghao Xu, Jian Zhang, Bingbing Ni, Teng Li, Chengjie Wang, Qi Tian, and Wenjun Zhang. Adversarial Domain Adaptation with Domain Mixup. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 6502–6509, 2020.

[92] Yuan Yao, Chang Liu, Dezhao Luo, Yu Zhou, and Qixiang Ye. Video Playback Rate Perception for Self-Supervised Spatio-Temporal Representation Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6548–6557, 2020.

[93] Yuning You, Tianlong Chen, Yongduo Sui, Ting Chen, Zhangyang Wang, and Yang Shen. Graph Contrastive Learning With Augmentations. *Advances in Neural Information Processing Systems*, 33, 2020.

[94] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization Strategy To Train Strong Classifiers With Localizable Features. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6023–6032, 2019.

[95] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. Mixup: Beyond Empirical Risk Minimization. *arXiv preprint arXiv:1710.09412*, 2017.

[96] Yabin Zhang, Bin Deng, Kui Jia, and Lei Zhang. Label Propagation with Augmented Anchors: A Simple Semi-Supervised Learning baseline for Unsupervised Domain Adaptation. In *European Conference on Computer Vision*, pages 781–797. Springer, 2020.

[97] Yiheng Zhang, Zhaofan Qiu, Ting Yao, Dong Liu, and Tao Mei. Fully Convolutional Adaptation Networks for Semantic Segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6810–6818, 2018.

[98] Bolei Zhou, Alex Andonian, Aude Oliva, and Antonio Torralba. Temporal Relational Reasoning in Videos. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 803–818, 2018.

[99] Zoran Zivkovic. Improved Adaptive Gaussian Mixture Model for Background Subtraction. In *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.*, volume 2, pages 28–31. IEEE, 2004.

[100] Yuliang Zou, Jinwoo Choi, Qitong Wang, and Jia-Bin Huang. Learning Representational Invariances for Data-Efficient Action Recognition. *arXiv preprint arXiv:2103.16565*, 2021.

[101] Yuliang Zou, Zizhao Zhang, Han Zhang, Chun-Liang Li, Xiao Bian, Jia-Bin Huang, and Tomas Pfister. PseudoSeg: Designing Pseudo Labels for Semantic Segmentation. *arXiv preprint arXiv:2010.09713*, 2020.

Appendix

The appendix consists of the following sections:

- Section A: Details of all the datasets used in our experiments.
- Section B: Description of the Temporal Graph Encoder architecture in CoMix.
- Section C: Additional implementation details of the experiments.
- Section D: Details of how background extraction was performed for CoMix.
- Section E: Some additional experiments and analysis of the results.
- Section F: Additional feature visualizations.

A Dataset Details

In this section, we provide the detailed description of the datasets we used to perform all the experiments for CoMix, namely, (1) UCF-HMDB [9], (2) Jester [57], and (3) Epic-Kitchens [54].

UCF-HMBD Dataset. The UCF-HMDB dataset (assembled by [9]) is derived from the original UCF101 [73] and HMDB51 [34]. It is constructed by collecting all the relevant and overlapping action classes or categories from both the datasets as two domains, resulting in 2 transfer tasks (UCF→HMDB and HMDB→UCF). The dataset possesses 12 action classes, namely, *Climb, Fencing, Golf, Kick_Ball, Pullup, Punch, Pushup, Ride_Bike, Ride_Horse, Shoot_Ball, Shoot_Bow,* and *Walk*. For some of the cases, multiple action classes from the original dataset are combined to form a single action super-class for that domain. E.g., *RockClimbingIndoor* and *RopeClimbing* classes in the HMDB51 [34] dataset are combined to form *Climb* class for the HMDB domain in the UCF-HMDB dataset. The detailed composition of the action classes is shown in Table 8. The dataset contains $3,209$ videos in total with 1438 training videos and 571 validation videos from UCF, and 840 training videos and 360 validation videos from HMDB (following the splits by [9]), with a class-wise distribution shown in Figure 6.

The datasets are publicly available to download at:
`https://www.crcv.ucf.edu/data/UCF101.php`
`https://serre-lab.clps.brown.edu/resource/hmdb-a-large-human-motion-database/`.

Table 8: **Action classes in UCF-HMDB Dataset.** The table shows the action class composition for the UCF-HMDB dataset from the original datasets (i.e., UCF101 and HMDB51) and their correspondence to each other for the video domain adaptation setting.

UCF-HMDB	HMDB51	UCF101
Climb	climb	RockClimbingIndoor RopeClimbing
Fencing	fencing	Fencing
Golf	golf	GolfSwing
Kick_Ball	kick_ball	SoccerPenalty
Pullup	pullup	PullUps
Punch	punch	Punch
Pushup	pushup	PushUps
Ride_Bike	ride_bike	Biking
Ride_Horse	ride_horse	HorseRiding
Shoot_Ball	shoot_ball	Basketball
Shoot_Bow	shoot_bow	Archery
Walk	walk	WalkingWithDog

Jester Dataset. The Jester [49] dataset is a large scale fine-grained dataset consisting of videos of humans performing pre-defined hand gestures. The original dataset consists of 148092 videos from 27 action classes. A cross-domain dataset is constructed (originally by [57]) as a subset of the original dataset by merging multiple action classes into a single action super-class and then split into source and target domain. E.g., *Swiping Left, Swiping Right, Swiping Up,* and *Swiping Down* are considered as a super-class *Swiping*. Then *Swiping Left, Swiping Up* are considered to be in the source domain,

while *Swiping Right, Swiping Down* to be in the target domain. Different sub-actions are put into different domains in order to maximize the domain discrepancy, as stated by [57]. The resulting cross-domain dataset possesses 7 action classes, namely, *Push and Pull, Rolling Hand, Sliding Two Fingers, Swiping, Thumps Up and Down, Turning Hand,* and *Zooming In and Out*. For Jester, we have only a single transfer task i.e. Jester(S)→Jester(T). The detailed composition of the action classes is shown in Table 9 with a class-wise distribution of videos depicted in Figure 7.

The dataset is publicly available to download at: `https://20bn.com/datasets/jester`.

Table 9: **Action Classes in Jester.** The table shows the action class composition for each of the domains (i.e. Source and Target) in the Jester dataset and their correspondence to each other for the domain adaptation setting.

Jester	Jester (S)	Jester (T)
Push and Pull	Pushing Hand Away	Pulling Hand In
	Pushing Two Fingers Away	Pulling Two Fingers In
Rolling Hand	Rolling Hand Forward	Rolling Hand Backward
Sliding Two Fingers	Sliding Two Fingers Left	Sliding Two Fingers Right
	Sliding Two Fingers Up	Sliding Two Fingers Down
Swiping	Swiping Left	Swiping Right
	Swiping Up	Swiping Down
Thumps Up and Down	Thumb Up	Thumb Down
Turning Hand	Turning Hand Counterclockwise	Turning Hand Clockwise
Zooming In and Out	Zooming Out With Full Hand	Zooming In With Full Hand
	Zooming Out With Two Fingers	Zooming In With Two Fingers

Epic Kitchens Dataset. The Epic-Kitchens [14] dataset is a challenging egocentric dataset consisting of videos (action segments) capturing daily activities performed in kitchens. The three largest kitchens, namely, P01, P22, and P08 form the three domains D1, D2, and D3, respectively. Moreover, the 8 largest action classes, namely, *take, put, open, wash, close, cut, pour,* and *mix* are used to form the dataset for the domain adaptation setting, following [54]. The dataset has 1543 training videos and 435 test videos from D1, 2495 training videos and 750 test videos from D2, and 3897 training videos and 974 test videos from D3. The class-wise distribution is shown in Figure 8. It can be seen that the dataset possesses high imbalance which makes it even more challenging.

The dataset is publicly available to download at: `https://epic-kitchens.github.io/2021`.

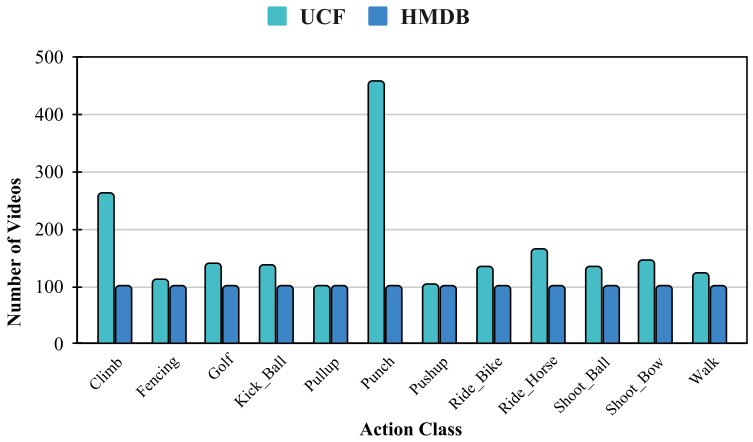

Figure 6: **Class-wise distribution of videos for UCF-HMDB.** The bar chart shows the distribution of videos across the 12 action classes of the UCF-HMDB dataset. Best viewed in color with zoom.

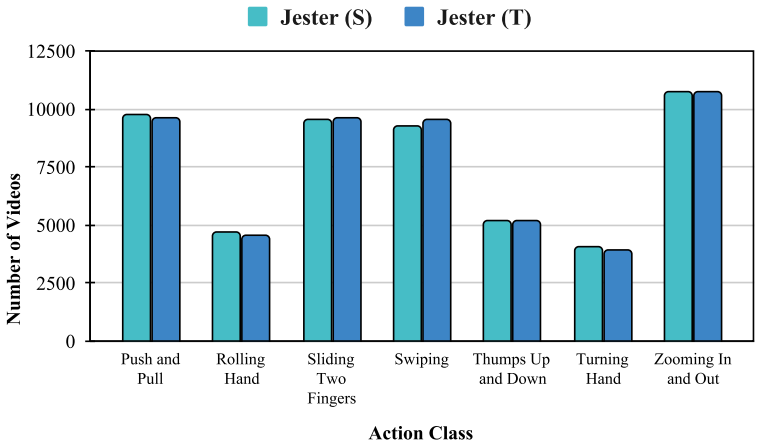

Figure 7: **Class-wise distribution of videos for Jester.** The bar chart shows the distribution of videos across the 7 action classes of the Jester dataset for the source and the target domains. Best viewed in color with zoom.

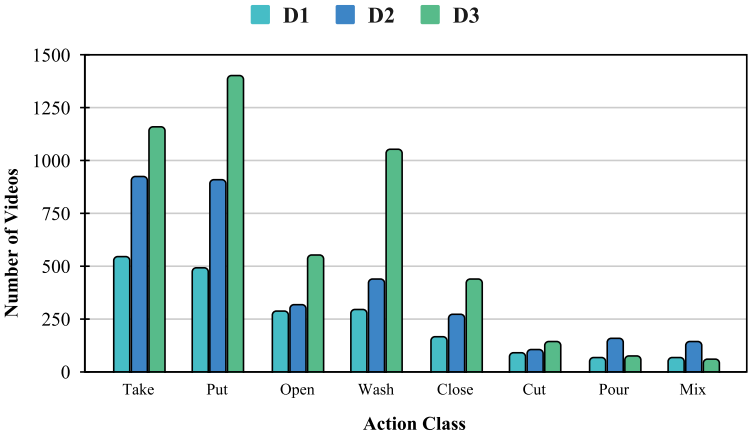

Figure 8: **Class-wise distribution of videos for Epic-Kitchens.** The bar chart shows distribution of videos across 8 action classes of Epic-Kitchens for three domains D1, D2, and D3. Best viewed in color with zoom.

B Temporal Graph Encoder

In this section, we provide the detailed description of the temporal graph encoder that we used for representing videos in our contrastive learning framework.

B.1 Graph Convolutional Network

The graph convolutional network (GCN) was originally proposed by [32] for node classification on graph structured data. Given an input graph $X \in \mathbb{R}^{N \times d}$ with N number of nodes with each node as a feature-vector of dimension d, the layer-wise propagation rule for a multi-layer GCN is:

$$H^{(l+1)} = \sigma(\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}H^{(l)}W^{(l)}) \tag{7}$$

where, $H^{(l)} \in \mathbb{R}^{N \times d_l}$ is the activation graph of the l^{th} layer with node feature dimension d_l; $H^{(0)} = X$. $\tilde{A} = A + I_N$ is the adjacency matrix of X with added self-connections through the indentity matrix I_N. $\tilde{D}_{ii} = \sum_j \tilde{A}_{ij}$ is the diagonal matrix used for normalization of $\tilde{A}$, and $W^{(l)}$ is the layer-specific trainable weight matrix. $\sigma(.)$ denotes the activation function, e.g. $\mathrm{ReLU}(.)$.

B.2 Videos as Similarity Graphs

Motivated by the importance of capturing long-range temporal structure in videos for action recognition and hence in cross-domain adaptation, we adopt a similarity graph to represent a video in our framework, as in [86]. Given a video $\mathbf{V}_n = \{\mathbf{v}_1, \mathbf{v}_2, ..., \mathbf{v}_n\}$ with n clips, with the corresponding clip-level feature vector representations as $\mathbf{Z}_n = \{\mathbf{z}_1, \mathbf{z}_2, ..., \mathbf{z}_n\}$, extracted by the feature encoder $\mathcal{F}$, each of dimension d. We construct a fully-connected graph X with n nodes from $\mathbf{Z}$ by considering the pairwise similarity or affinity between two feature vectors as:

$$F(\mathbf{z}_i, \mathbf{z}_j) = \phi(\mathbf{z}_i)^\top \phi'(\mathbf{z}_j) \tag{8}$$

where, $\phi(.)$ and $\phi'(.)$ represent two different transformation functions of the original feature vectors, defined as $\phi(\mathbf{z}) = \mathbf{w}\mathbf{z}$ and $\phi(\mathbf{z}) = \mathbf{w}'\mathbf{z}$. Here, the transformations are parameterized with the weights $\mathbf{w}$ and $\mathbf{w}'$ of dimension $d \times d$ each. Using such transformations helps learn the long-range correlations between the feature vectors to harness the rich temporal information of the video. We get a similarity matrix A^{sim} of dimension $n \times n$ by computing the affinity for all the possible pairs, using Eq. 8. The matrix is then normalized using a softmax function as:

$$A^{sim}_{ij} = \frac{\exp(F(\mathbf{z}_i, \mathbf{z}_j))}{\sum_{j=1}^{n} \exp(F(\mathbf{z}_i, \mathbf{z}_j))} \tag{9}$$

The normalized matrix A^{sim} is now considered as the adjacency matrix for the similarity graph, allowing us to learn the edge-weights between the nodes through back-propagation, by the help of the learnable weights $\mathbf{w}$ and $\mathbf{w}'$. Hence, the resulting similarity graph convolutional network has the following propagation rule, similar to Eq. 7:

$$H^{(l+1)} = \sigma(A^{sim(l)}H^{(l)}W^{(l)}) \tag{10}$$

where, $H^{(0)} = X$, and $A^{sim(l)}$ is the affinity/adjacency matrix computed using the node features of the l^{th} layer, similar to [86].

B.3 Scalability with Graph Convolutions

The learning strategy for graph representation used in CoMix ensures that the number of learnable parameters are independent of the number of graph nodes. As described in detail above, we construct the fully connected graph with the edge weights (pairwise similarity) obtained using two different transformation functions, and on the clip-level feature vectors (where each feature vector represents a node). This strategy makes the number of trainable parameters independent of the number of nodes in a GCN layer and hence, independent of the number of clips used for a video. While fully connected graph convolutions will increase the computation with longer clip sequences, we can adopt sparse video sampling [7] or techniques like [88] to tradeoff computation.

C Additional Implementation Details

In this section, we provide additional implementation details including hyperparameters with a detailed overview of the model architectures used in our approach.

C.1 Model Architectures

Feature Encoder. Following [12], we use I3D [4] as our feature encoder $\mathcal{F}$ for all our experiments. It takes clips (set of consecutive frames) of videos, of length 8, as input and maps them to the corresponding clip-level feature vector of length 1024. The layer-wise architectural view of the I3D feature encoder backbone is shown below:

```
InceptionI3D:
  (Conv3d_1a_7x7): Unit3D()
  (MaxPool3d_2a_3x3): MaxPool3dSamePadding()
  (Conv3d_2b_1x1): Unit3D()
  (Conv3d_2c_3x3): Unit3D()
  (MaxPool3d_3a_3x3): MaxPool3dSamePadding()
  (Mixed_3b): InceptionModule()
  (Mixed_3c): InceptionModule()
  (MaxPool3d_4a_3x3): MaxPool3dSamePadding()
  (Mixed_4b): InceptionModule()
  (Mixed_4c): InceptionModule()
  (Mixed_4d): InceptionModule()
  (Mixed_4e): InceptionModule()
  (Mixed_4f): InceptionModule()
  (MaxPool3d_5a_2x2): MaxPool3dSamePadding()
  (Mixed_5b): InceptionModule()
  (Mixed_5c): InceptionModule()
  (avg_pool): AvgPool3d()
  # Outputs features of length 1024.
```

Temporal Graph Encoder For the temporal graph encoder $\mathcal{G}$, we use a 3-layer similarity based graph convolutional neural network, as discussed in Section B. The graph encoder takes the output of the feature encoder $\mathcal{F}$ as input and gives the logits as the output. The layer-wise architectural view of the temporal graph encoder is shown below:

```
TemporalGraph:
  # Takes the output of InceptionI3d as input.
  (gc1): GraphConvolution(1024, 256)
  (relu): ReLU()
  (dropout): Dropout()
  (gc2): GraphConvolution(256, 256)
  (relu): ReLU()
  (dropout): Dropout()
  (gc3): GraphConvolution(256, num_classes)
  # Outputs the logits.
```

C.2 Hyperparameters

Below, we provide the exact values of the two loss weights λ_{bgm} and λ_{tpl} (refer to Eq. 6 in the main paper) for each of the datasets:

UCF-HMDB: UCF→HMDB: $\lambda_{bgm} = 0.1$, $\lambda_{tpl} = 0.01$; HMDB→UCF: $\lambda_{bgm} = 0.1$, $\lambda_{tpl} = 0.1$.

Jester: S→T: $\lambda_{bgm} = 0.1$, $\lambda_{tpl} = 0.1$.

Epic-Kitchens: $\lambda_{bgm} = 0.01$, $\lambda_{tpl} = 0.01$ was used for all the 6 tasks.

For all the experiments, the source-only models were trained for 4000 iterations and then our framework was trained for an additional 10000 iterations, initialized with the source-only models.

D Background Extraction Details

In this section, we provide more details about the background extraction including qualitative samples used in our temporal contrastive learning framework.

Temporal Median Filter. Temporal median filtering (TMF) is one of the most simple, intuitive, and fast methods for background generation. It has proven to be successful and commonly adopted in several recent deep learning pipelines [77, 47]. For videos, a pixel-wise temporal median filter is applied on the sequence of frames to obtain the corresponding background. The method is designed with the principle that for a given pixel location, in a sequence of frames, the most frequently repeated intensity along the temporal direction is most likely to be the background value for that pixel [62, 40]. It does so by computing the pixel-wise median values along the temporal direction. We adopt this method for extracting backgrounds for our framework because of its simplicity and effectiveness. It must be noted that the `CoMix` framework is agnostic to the method used for background extraction and can be incorporated with any other background extraction techniques for videos. Figure 9 shows some representative video clips randomly sampled from both the domains of the UCF-HMDB along with the corresponding background frame extracted using temporal median filtering. Note that we extract a single background frame per video from one domain and then mix it with all the frames of a video from the other domain to generate synthetic background mixed videos. The addition of a static background frame to all frames of a video does not hinder the temporal action dynamics (motion patterns) possessed by the video. We validate this hypothesis by obtaining optical flow for a given video before and after performing background mixing, and observe no significant change in them.

E Additional Experimental Results

Effect of Source-only Model Initialization. We follow the standard 'pre-train then adapt' procedure used in prior works [9, 12] and train the model with only source data to provide a warmstart before our approach is trained. However, in order to understand the contribution of source-only model initialization, we trained the models with the default random initialization keeping all the other hyperparameters same. The average performance dropped to 86.4% (-$\mathbf{3.9}\%$) on UCF-HMDB dataset. This validates that the source-only initialization plays an important role in providing a proper warm-start to the models which leads to an effective optimization, in consistent with prior works [9, 12].

Effect of Target Pseudo-label Threshold. In Table 10, we study the sensitivity of the final performance with respect to the pseudo label threshold on the UCF-HMDB dataset and notice that the performance of `CoMix` is quite stable with respect to this parameter (best performance at threshold set to 0.7). The slight decrease in performance with PL = 0.9 is understandable since very few target videos are getting selected as additional positives for the supervised contrastive loss.

Table 10: **Effect of Target Pseudo-label Threshold.** Performance on UCF-HMDB.

PL Threshold	U→H	H→U	Average
0.5	85.6	93.5	89.5
0.6	85.6	93.5	89.5
0.7	**86.7**	**93.9**	**90.3**
0.8	86.4	92.5	89.4
0.9	85.6	90.9	88.2

Effect of Mixed Backgrounds. We tried a variant of background mixing in which the backgrounds from both the domains are first convexly combined to form a mixed-background, which sort of represents a generalized background for both the domains. The obtained mixed-background is then used for the background mixing component and is mixed with the videos from both the domains. This alternate background mixing strategy provided an average performance of 89.4% on the UCF-HMDB dataset, which is 0.9% lower than the cross-domain background mixing (i.e., adding background from one domain to the other) used in the proposed approach.

Convergence and Multiple Seeds. The convergence of the proposed approach varies with dataset and task complexity ranging from 3000 iterations for HMDB→UCF dataset to 7000 iterations for UCF→HMDB and EpicKitchens datasets. We observe that the convergence is fairly stable across different seeds and report the average performance over three runs with different random seeds. To quantify, the standard deviations in performance obtained for UCF-HMDB, Jester and EpicKitchens datasets are 0.3, 0.1 and 0.2 respectively.

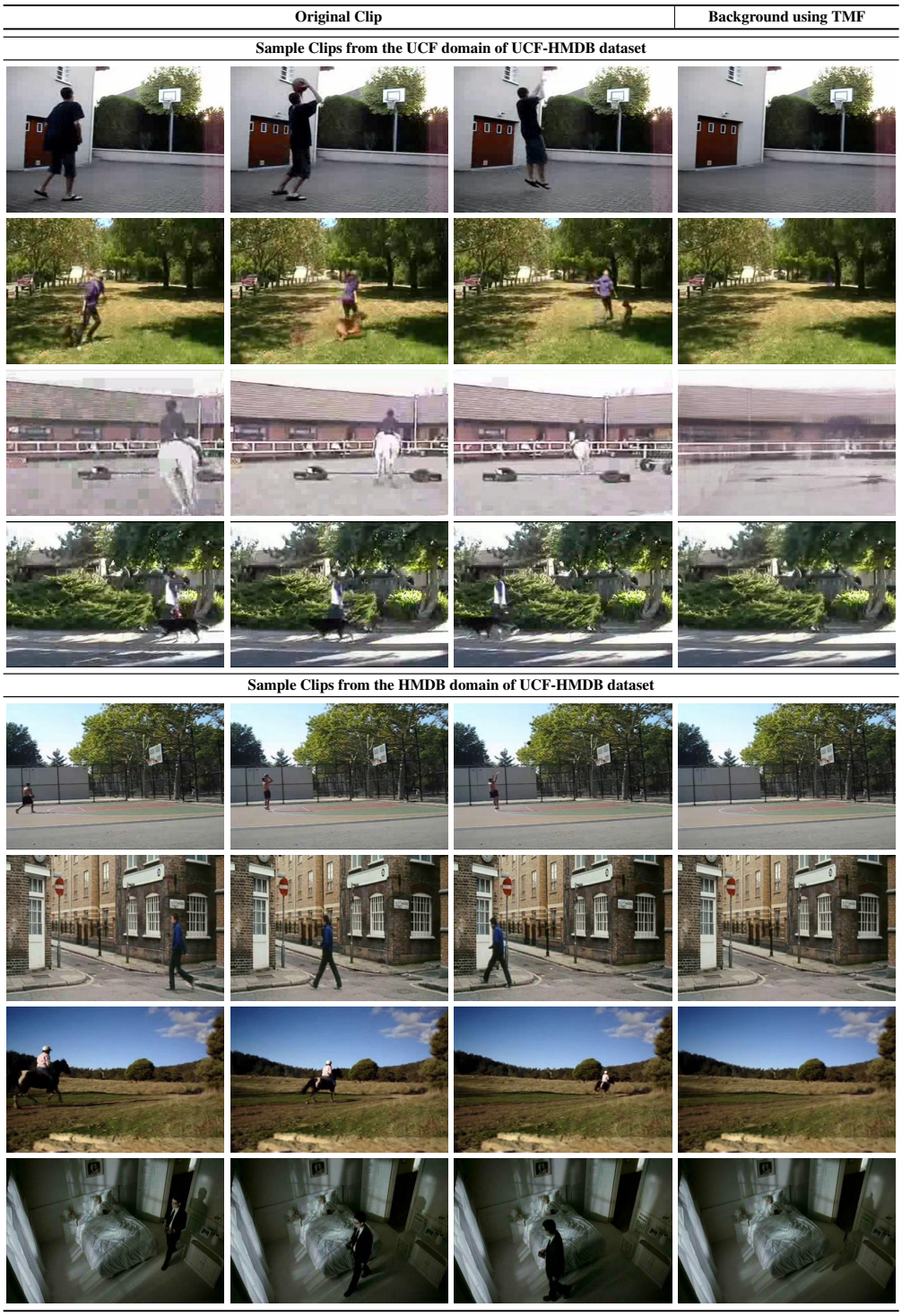

Figure 9: **Background Extraction.** The figure shows some representative video clips from UCF-HMDB dataset with corresponding extracted background using temporal median filtering (TMF). Best viewed in color.

F Additional Feature Visualizations

In this section, we provide additional t-SNE [46] plots to visualize the features learned using different components of our `CoMix` framework. We choose the Source only model as a vanilla method and add different components one-by-one to visualize their contributions in learning discriminative features

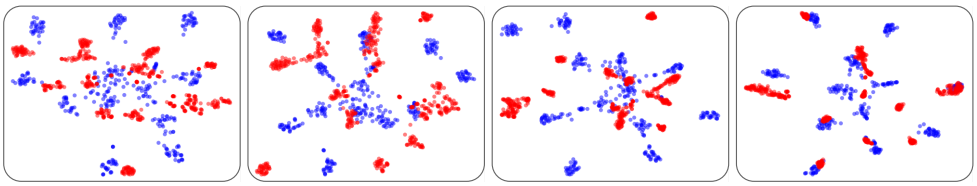

Figure 10: **Feature Visualizations using t-SNE.** Plots show visualization of our approach with different components on HMDB→UCF task from UCF-HMDB. Blue and red dots represent source and target data respectively. Features for both target and source domain become progressively discriminative and improve from left to right by adoption of our novel components within a contrastive learning framework. Best viewed in color.

for video domain adaptation. In the main paper, we have provided the plots for the UCF→HMDB task from the UCF-HMDB dataset (refer to Figure 5 in main paper). Here we provide the plots for the HMDB→UCF task in Figure 10. As can be seen from Figure 10, alignment of domains including discriminability improves as we adopt "TCL" and "BGM" to the vanilla Source only model. The best results are obtained when all three components "TCL", "BGM" and "TPL" i.e., `CoMix` are added and trained using an unified framework (Eq. 6 in main paper) for unsupervised video domain adaptation.