Representation Learning via Global Temporal Alignment and Cycle-Consistency

Isma Hadji, Konstantinos G. Derpanis, Allan D. Jepson
Samsung AI Centre Toronto
{isma.hadji, allan.jepson}@samsung.com k.derpanis@partner.samsung.com

Abstract

We introduce a weakly supervised method for representation learning based on aligning temporal sequences (e.g., videos) of the same process (e.g., human action). The main idea is to use the global temporal ordering of latent correspondences across sequence pairs as a supervisory signal. In particular, we propose a loss based on scoring the optimal sequence alignment to train an embedding network. Our loss is based on a novel probabilistic path finding view of dynamic time warping (DTW) that contains the following three key features: (i) the local path routing decisions are contrastive and differentiable, (ii) pairwise distances are cast as probabilities that are contrastive as well, and (iii) our formulation naturally admits a global cycle-consistency loss that verifies correspondences. For evaluation, we consider the tasks of fine-grained action classification, few shot learning, and video synchronization. We report significant performance increases over previous methods. In addition, we report two applications of our temporal alignment framework, namely 3D pose reconstruction and fine-grained audio/visual retrieval.

1. Introduction

Temporal sequences (*e.g.*, videos) are an appealing data source as they provide a rich source of information and additional constraints to leverage in learning. By far the main focus on temporal sequence analysis has been on learning representations targeting distinctions at the global signal level, *e.g.*, action classification, where abundant labeled data is available for training. In this paper, we target a weakly-supervised training regime for representation learning, capable of making fine-grained temporal distinctions.

Most previous approaches to temporally fine-grained understanding of sequential signals have considered fully-supervised training methods (*e.g.*, [56]), where labels are provided at the sub-sequence level, *e.g.*, frames. The major drawback of these methods is the expense in acquiring dense labels, and their subjective nature. In contrast, a key consideration in our work is the selection of training signals capable of scaling up to large amounts of data yet supporting finer-grained video understanding.

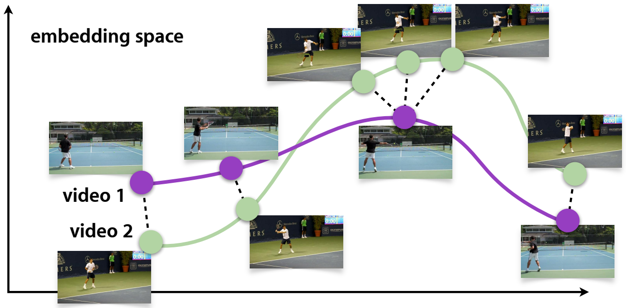

Figure 1. We introduce a representation learning approach based on (globally) aligning pairs of temporal sequences (*e.g.*, video) depicting the same process (*e.g.*, human action). Our training objective is to learn an element-wise embedding function that supports the alignment process. For example, here we illustrate the alignment (denoted by black dashed lines) in the embedding space between videos of the same human action (*i.e.*, tennis forehand) containing significant variations in their appearances and dynamics. Empirically, we show that our learned embeddings are sensitive to both human pose and fine-grained temporal distinctions, while being invariant to appearance, camera viewpoint, and background.

As outlined in Fig. 1, given a set of paired sequences capturing the same process (*e.g.*, tennis forehand) but highly varied (*i.e.*, different participants, action executions, scenes, and camera viewpoints), our method trains an embedding network to support the recovery of their latent temporal alignment. We refer to our method as weakly supervised, as only readily available sequence-level labels (*e.g.*, tennis forehand) are required to construct training pairings containing the same process. Given such a pair of sequences, we use their latent temporal alignment as a supervisory signal to learn fine-grained temporal distinctions.

Key to our proposed method is a novel dynamic time warping (DTW) formulation to score global alignments between paired sequences. DTW enforces a stronger constraint over simply considering local (soft) nearest neighbour correspondences [15], since the temporal ordering of the matches in the sequences are taken into account. We depart from previous differentiable DTW methods [30, 8, 6] by taking a probabilistic path finding view of DTW that encompasses the following three key features. First, we introduce a differentiable smoothMin operator that effectively selects each successive path extension. Moreover, we show that this operator has a contrastive effect across paths which is missing in previous differentiable DTW formulations [30, 8, 6]. Second, the pairwise embedding similarities that form our cost function are defined as probabilities, using the softmax operator. Optimizing our loss is shown to correspond to finding the maximum probability of any feasible alignment between the paired sequences. The softmax operator over element pairs also provides a contrastive component which we show is crucial to prevent the model from learning trivial embeddings. This forgoes the need for a downstream discriminative loss and the corresponding non-trivial task of defining negative alignments, *e.g.*, [8, 6]. Third, as an additional supervisory signal, our probabilistic framework admits a straightforward global cycle-consistency loss that matches the alignments recovered through a cycle of sequence pairings. Collectively, our method takes into account long-term temporal information that allows us to learn embeddings sensitive to fine-grained temporal distinctions (*e.g.*, human pose), while being invariant to nuisance variables, *e.g.*, camera viewpoint, background, and appearance.

Contributions. We make the following key contributions:

- A novel weakly supervised method for representation learning tasked with discovering the alignment between sequence pairings for the purpose of fine-grained temporal understanding.
- A differentiable DTW formulation with two novel features: (i) a smoothMin operation that admits a probabilistic path interpretation and is contrastive across alternative paths, and (ii) a probabilistic data term that is contrastive across alternative data pairs.
- A global cycle consistency loss to further enforce the temporal alignment.
- An extensive set of evaluations, ablations, and comparisons with previous methods. We report significant performance increases on several tasks requiring fine-grained temporal distinctions.
- Two downstream applications, namely 3D pose reconstruction and audio-visual retrieval.

Our code and trained models will be available at: https://github.com/hadjisma/VideoAlignment.

2. Related work

Representation learning. Most focus in representation learning with videos has been cast in a fully supervised setting, *e.g.*, [47, 43, 7, 16]. Self-supervised learning with images or videos has emerged as a viable alternative to supervised learning, where the supervisory signal is obtained from the data. For video, a variety of proxy tasks have been defined in lieu of training with annotations, such as classifying whether video frames are in the correct temporal order (*e.g.*, [32, 17, 28, 4, 55, 54]), predicting whether a video is played at a normal or modified rate [3], solving a spatiotemporal jigsaw puzzle task [1], predicting figure-ground segmentation [35], predicting pixel [57, 29, 50, 26] or region correspondences [52, 53, 25] across neighbouring video frames, and predicting some aspect of future frames conditioned on past frames [49, 51, 21]. Others have considered multimodal settings, such as predicting video-audio misalignment [34]. Similar to [15], our method is best characterized as weakly supervised, where sequence-level labels are used to determine sequence pairings for training.

Sequence alignment. Several methods [42, 40] assume paired, temporally synchronized videos of the same physical event for the purpose of representation learning. In contrast, and more closely related to our work, are methods that seek the alignment between sequences capturing the same process. One approach [15] is to cast the learning objective as maximizing the number of elements between sequences that can be brought into one-to-one correspondence via (soft) nearest neighbours. This method does not leverage the long-term temporal structure of the sequences as done in dynamic time warping (DTW) [38]. Given a cost function, DTW finds the optimal alignment between two sequences defined between elements comprising the sequences. Recent efforts [13, 30, 5, 8, 6] have explored differentiable approximations of the discrete operations underlying DTW to allow gradient-based training. Similar to recent work [8, 6], we also incorporate a relaxed DTW as our loss for sequence alignment. Our formulation is probabilistic and includes a contrastive definition of the element-wise similarities. A key distinction with these prior works and our own, beyond differences in the target application domain (*e.g.*, video-transcript alignment [8]), is that rather than incorporate contrastive modelling after the DTW step (*e.g.*, through the use of margin-based loss), our method includes contrastive signals in both the differentiable min approximation and the pairwise matching cost function used in our DTW framework.

Contrastive learning. One can also draw parallels with contrastive learning using the cross-entropy loss (*i.e.*, negative log softmax) [20, 33, 48], where the goal is to learn a representation that brings different views of the same data together (*i.e.*, positives) in the embedding space, while pushing views of different data (*i.e.*, negatives) apart. This amounts to encoding information shared across the views, while eschewing unique factors to each view. To construct different views, previous work has explored a variety of augmentation and sampling schemes [14, 45, 10, 21, 19, 23] and correspondences across different modalities (*e.g.*, video-audio, video-text, and luminance-depth

[11, 45, 44, 36, 31]). In these works, the positive and negative pairings are known by construction, *e.g*., via image augmentation. We also make use of contrastive losses, but note that the correspondences (*i.e*., positives) between the sequences are latent rather than known.

Cycle-consistency. In addition to alignment, our method also incorporates cycle-consistency as a supervisory signal, where the objective is to verify matches across sets. Similar to recent work [15], we apply cycle-consistency across two temporal sequences. A key difference is that previous work [15] applies cycle-consistency independently to local matches across sequences; whereas, we consider both local matches and their global temporal ordering, which we demonstrate empirically leads to improved alignments.

3. Technical approach

In this section, we describe our weakly-supervised approach to representation learning based on the alignment of sets of sequence pairs that capture the same process. Our learning objective is the training of a shared embedding function applied to each sequence element. In the case of multimodal sequences (*e.g*., audio-video), we have separate encoders for each modality that map their inputs to a common embedding space. Figure 2 provides an illustrative overview of our alignment approach to representation learning, which we fully unpack in the following subsections.

3.1. Background

Dynamic time warping (DTW) computes the optimal alignment between two sequences, $\mathbf{X}$ and $\mathbf{Y}$, subject to certain alignment constraints. Let $\mathbf{X} = \begin{bmatrix}\mathbf{x}_1 & \mathbf{x}_2 & \cdots & \mathbf{x}_M\end{bmatrix} \in \mathbb{R}^{D\times M}$ and $\mathbf{Y} \in \mathbb{R}^{D\times N}$ denote the two sequences, where D corresponds to the dimensionality of the constituent sequence elements and M, N the respective sequence lengths. Given a cost matrix, $\mathbf{C} \in \mathbb{R}^{M\times N}$, with elements $c_{i,j}$ defined by a cost function, $c(\mathbf{x}_i, \mathbf{y}_j)$, that measures the cost of matching elements $\mathbf{x}_i$ and $\mathbf{y}_j$, we seek a feasible path between $c_{0,0}$ and $c_{M,N}$ that minimizes the total accumulated cost. The feasible paths are subject to matching endpoints, monotonicity, and continuity constraints. While not used here, the endpoint constraint can be relaxed to allow for subsequence matching, *e.g*., [39, 6].

The number of feasible alignments in DTW grows exponentially with the sequence lengths. Fortunately, the structure of DTW with an appropriate cost function, $c(\mathbf{x}_i, \mathbf{y}_j)$, is amenable to dynamic programming [2] which shares both quadratic time and space complexity. The optimal alignment (*i.e*., path through the cost matrix) is found by evaluating the following recurrence [38]:

$$\begin{aligned}\mathbf{R}(i,j) = c(\mathbf{x}_i, \mathbf{y}_j) + \\ \min([\mathbf{R}(i-1,j-1) \quad \mathbf{R}(i-1,j) \quad \mathbf{R}(i,j-1)]^\top),\end{aligned} \tag{1}$$

where $\mathbf{R}(0,0) = 0$, $\mathbf{R}(0,:) = \mathbf{R}(:,0) = \infty$, and $\mathbf{R}(i,j)$ stores the partial accumulated cost along the optimal and feasible path ending with the alignment between $\mathbf{x}_i$ and $\mathbf{y}_j$. The minimum operation amounts to a first-order Markov assumption, where the local path routing is deterministic.

Due to the discrete nature of the min operator in (1), responsible for local correspondence decisions, several works [13, 8, 6] have considered smooth variants suitable for gradient-based training. In Sec. 3.2, we introduce a smooth relaxation of the min operator with favourable properties for our representation learning setting. Then in Sec. 3.3, we define our cost function which introduces a contrastive learning signal throughout the alignment process.

3.2. Local differentiable decisions

For brevity we use the notation

$$\mathbf{r}_{i,j} = [\mathbf{R}(i-1,j-1) \quad \mathbf{R}(i-1,j) \quad \mathbf{R}(i,j-1)]^\top \tag{2}$$

to denote the incoming optimal accumulated costs from the feasible paths leading into (i,j). We first modify (1) as

$$\mathbf{R}(i,j) = c(\mathbf{x}_i, \mathbf{y}_j) + \underbrace{[s(\mathbf{r}_{i,j}) - \min(\mathbf{r}_{i,j})]}_{d(\mathbf{r}_{i,j})} + \min(\mathbf{r}_{i,j}), \tag{3}$$

where $s(\mathbf{r}_{i,j})$ is a smooth approximation of the minimum operator. The term $d(\mathbf{r}_{i,j})$ can be seen as a (non-differentiable) additional penalty on any path that reaches (i,j). With this added penalty term, (3) reduces to simply

$$\mathbf{R}(i,j) = c(\mathbf{x}_i, \mathbf{y}_j) \; + s(\mathbf{r}_{i,j}). \tag{4}$$

For an appropriate choice of $s(\cdot)$ the right hand side is now differentiable. Note that any path that is optimal according to (4) will correspond to a feasible path for the original DTW problem (although perhaps not optimal for that problem). Moreover, the cost of such a path according to (4) will be the original cost plus the sum of the penalties $d(\mathbf{r}_{i,j})$ over all points (i,j) on that path.

We are left with choosing a smooth approximation $s(\cdot)$ for the minimum operator. Here, we use a standard relaxation of the min operator [27] (specifically, the expected value $E_{i\sim q(i)}[a_i]$ for $q(i) := \text{softmax}(\{-a_i/\gamma\})$):

$$\text{smoothMin}(\mathbf{a};\gamma) = \begin{cases}\min\{a_i \mid 1 \le i \le N\}, & \gamma = 0 \\ \frac{\sum_{i=1}^N a_i e^{-a_i/\gamma}}{\sum_{j=1}^N e^{-a_j/\gamma}}, & \gamma > 0\end{cases}, \tag{5}$$

where γ denotes a temperature hyper-parameter. We refer to solving the recurrence relation (4), with the function $s(\cdot)$ taken to be smoothMin, as the smoothDTW problem.

Previous alignment methods [13, 8, 6] have instead used the following $\min^\gamma$ formulation as a continuous approxima-

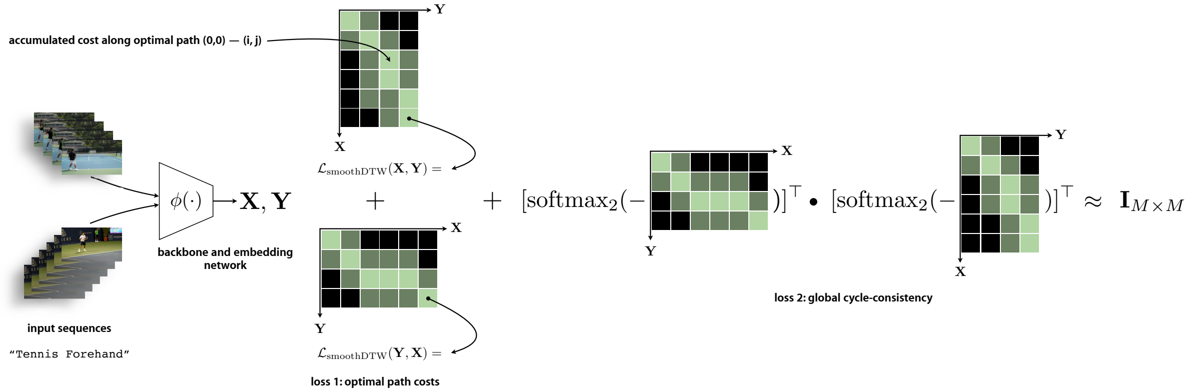
accumulated cost along optimal path (0,0) — (i, j)
Y
X
$\mathcal{L}_{\text{smoothDTW}}(\mathbf{X},\mathbf{Y}) =$
input sequences
"Tennis Forehand"
$\phi(\cdot)$
$\mathbf{X}, \mathbf{Y}$
backbone and embedding network
+
X
Y
$\mathcal{L}_{\text{smoothDTW}}(\mathbf{Y},\mathbf{X}) =$
loss 1: optimal path costs
$+ \; [\text{softmax}_2(-$
$)]^\top \bullet [\text{softmax}_2(-$
$)]^\top \approx \mathbf{I}_{M\times M}$
X
Y
Y
X
loss 2: global cycle-consistency

Figure 2. Our sequence alignment approach to representation learning begins by encoding each element comprising our sequences (*e.g*., image frames) using a trainable framewise backbone encoder plus embedding network, $\phi(\cdot)$, yielding two sequences of embeddings, $\mathbf{X}$ and $\mathbf{Y}$. The cost of matching these two sequences is expressed as negative log probabilities and consists of two parts: (i) alignment losses, smoothDTW$(\cdot,\cdot)$, from $\mathbf{X}$ to $\mathbf{Y}$ and $\mathbf{Y}$ to $\mathbf{X}$ based on the cumulative cost along the optimal respective paths and (ii) a global cyclic-consistency loss that verifies the correspondences computed between each ordered pair of sequences, where $\cdot$ denotes matrix multiplication and $\mathbf{I}_{M\times M}$ is the square identity matrix. Note, our alignment cost smoothDTW$(\cdot,\cdot)$ is not symmetric in its two arguments (due to the pairwise matching cost in (8)). Higher intensities in the cells comprising the accumulated cost matrices indicate lower values.

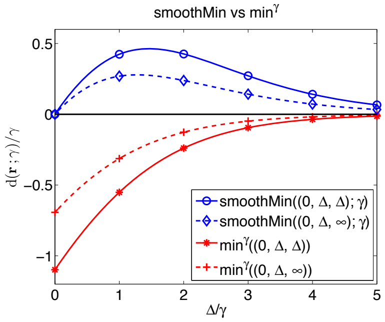

Figure 3. The penalty terms $d(\mathbf{r};\gamma) = s(\mathbf{r};\gamma) - \min(\mathbf{r})$ for the two smooth min approximations given in (5) and (6). For simplicity we can assume the r_i's are in sorted order. We evaluate two extreme cases where $r_2 = r_1 + \Delta$ for some $\Delta \geq 0$ and the next largest value, r_3, is either equal to r_2 or much larger. It can be shown that $d(\mathbf{r};\gamma) = \gamma d(\mathbf{r}/\gamma; 1)$ and therefore we need only plot one scale-invariant curve for each case.

tion of the min operator:

$$\min{}^\gamma(\mathbf{a};\gamma) = \begin{cases} \min\{a_i \mid 1 \leq i \leq N\}, & \gamma = 0 \\ -\gamma \log \sum_{i=1}^N e^{-a_i/\gamma}, & \gamma > 0 \end{cases}, \tag{6}$$

where again γ denotes a temperature hyper-parameter.

While both the min$^\gamma$ and smoothMin operators are differentiable approximations of the min operator (with the min operator subsumed as a special case), their different behaviours have profound effects on learning in our setting. These differences are illustrated in the plot in Fig. 3, where without loss of generality we assume the costs are sorted in increasing order, with $r_2 = r_1 + \Delta$ for some $\Delta \geq 0$ and $r_3 \geq r_2$. As can be seen, the min$^\gamma$ function is strictly monotonically increasing. As a result, with all other things being equal, minimizing this function encourages ties, *i.e*., the penalty is minimized only when $\Delta = 0$. This is an undesirable behaviour as we seek the resulting embeddings to yield a well-defined path (*i.e*., optimal alignment) in our cumulative cost matrix, $\mathbf{R}(i,j)$. In contrast, our smoothMin operator defines a contrastive watershed (at approximately 1.5γ), where values to the left of the watershed encourage ties, while to the right the values are encouraged to be well separated. Moreover, since $d(\mathbf{r};\gamma) \geq 0$ for smoothMin, our smoothDTW approach always provides an upper bound on the cost of the optimal path. The supplemental presents an expanded discussion and comparison.

3.3. Contrastive cost function

To complete the definition of our smoothDTW recurrence in (4), we now specify the cost function $c(\mathbf{x}_i, \mathbf{y}_j)$. Specifically, for each element $\mathbf{x}_i$ in $\mathbf{X}$, we wish to express the cost of matching $\mathbf{x}_i$ to any single item $\mathbf{y}_j$ in $\mathbf{Y}$, given that at least one of the elements in $\mathbf{Y}$ must match. Moreover, in keeping with our probabilistic path finding formulation, $c(\mathbf{x}_i, \mathbf{y}_j)$ should be the negative log probability of matching the given $\mathbf{x}_i$ to a selected $\mathbf{y}_j$ in $\mathbf{Y}$. This leads to the (non-symmetric) contrastive formulation

$$c(i,j;\mathbf{X},\mathbf{Y}) = -\log(\text{softmax}_2(\tilde{\mathbf{X}}^\top\tilde{\mathbf{Y}};\beta))_{i,j}, \tag{7}$$

$$= -\log\left(\frac{\exp(\tilde{\mathbf{x}}_i^\top\tilde{\mathbf{y}}_j/\beta)}{\sum_{k=1}^N \exp(\tilde{\mathbf{x}}_i^\top\tilde{\mathbf{y}}_k/\beta)}\right), \tag{8}$$

where the softmax$_2$ operator is defined as the standard softmax with a temperature hyper-parameter, β, over the second matrix dimension, *i.e.* columns. Also the notation $\tilde{\mathbf{x}}_i$ denotes L_2 normalization, so $\tilde{\mathbf{x}}_i = \mathbf{x}_i/||\mathbf{x}_i||_2$, and so on. The use of the negative log softmax operator over the set of correspondences in (8) encourages a (soft) winner-take-all, where one pair $(\mathbf{x}_i, \mathbf{y}_j)$ has a significantly higher cosine similarity than all the other options. The normalization across the column penalizes situations without a clear winner.

With this cost function the resulting values $\mathbf{R}(i,j)$ are the optimum value of the negative log probability for any feasible path starting at $(0,0)$ and ending at (i,j). Here, this negative log probability is the sum of the matching cost, $c(\mathbf{x}_m, \mathbf{y}_n)$ (cf. (8)), and the smoothness penalty $d(\mathbf{r}_{m,n})$ (cf. (14)), at each vertex, (m,n), along the optimal path ending at (i,j). Correspondingly, we define the alignment loss for matching $\mathbf{X}$ to $\mathbf{Y}$ as:

$$\mathcal{L}_{\text{smoothDTW}}(\mathbf{X},\mathbf{Y}) = \mathbf{R}(M,N). \tag{9}$$

We use the sum of the alignment losses for matching $\mathbf{X}$ to $\mathbf{Y}$ and vice versa as the overall alignment loss, as shown in the left panel in Fig. 2.

The combination of the softmax over elements in $\mathbf{Y}$, in (8), and the use of smooth DTW to formulate the alignment cost (9), rewards embeddings that are both: a) contrastive across the elements in $\mathbf{Y}$; and, moreover, b) have their best matching pairs $(\mathbf{x}_i, \mathbf{y}_j)$ arranged along a feasible path from $(0,0)$ to (M,N). We show in our ablation study that the ability to leverage these two properties during training are key to our utilization of the temporal alignment proxy. In contrast, dropping the softmax and simply computing the inner-product between elements could lead to the collapse of the embeddings around a single point during training, thus allowing the network to trivially minimize the alignment cost. To avoid such collapse, previous DTW-based methods have resorted to adding a discriminative loss downstream [8, 6].

3.4. Global cycle-consistency

An additional loss is based on the notion that the match from sequence X to Y, composed with the match from Y to X, should ideally be the identity. We formulate this directly in terms of the cumulative cost matrix $\mathbf{R}_{X,Y}$ for matching sequence X to Y (as defined by (4), (5), and (8)), along with the cost matrix for matching Y to X, namely $\mathbf{R}_{Y,X}$. Given the interpretation that $\mathbf{R}_{X,Y}(i,j)$ is the optimal negative log probability of a path from $(0,0)$ to (i,j) for matching X to Y, consider the implied conditional distribution for matching the prefix sequences $X(1\!:\!i)$ to $Y(1\!:\!j)$ for different j's, namely

$$\begin{aligned} p_{X,Y}(j \,|\, i) &:= [\text{softmax}_2(-\mathbf{R}_{X,Y}/\alpha)]_{i,j} \\ &= \frac{e^{-\mathbf{R}_{X,Y}(i,j)/\alpha}}{\sum_{k=1}^{N} e^{-\mathbf{R}_{X,Y}(i,k)/\alpha}}. \end{aligned} \tag{10}$$

Note that this distribution does not use any information for elements $k > i$ from sequence X, and is only obtained from the forward pass of matching X to Y. We use the notation

$$P_{X,Y} := [\text{softmax}_2(-\mathbf{R}_{X,Y}/\alpha)]^\top \tag{11}$$

to denote the $N \times M$ matrix with elements $(\mathbf{P}_{X,Y})_{n,m} = p_{X,Y}(n \,|\, m)$.

Ideally, the contrastive matching in (8) is sharp and forms a feasible path from $(0,0)$ to (i,j), thereby providing a strongly peaked conditional distribution $p_{X,Y}(j|i)$ for each i. However, without knowing the ground truth matching $\{(i_k, j_k)\}_{k=0}^{L}$, we cannot use an explicit log-likelihood loss. This issue can be avoided by considering the composed conditional distribution

$$p_{X,Y,X}(j \,|\, i) := \sum_{k=1}^{N} p_{Y,X}(j \,|\, k)\, p_{X,Y}(k \,|\, i), \tag{12}$$

which is formed by treating $p_{X,Y}(k \,|\, i)$ and $p_{Y,X}(j \,|\, k)$ as conditionally independent distributions. It is easy to verify that this is indeed a distribution over elements j of X. Moreover, following the above matrix notation, it is represented by the $M \times M$ matrix $\mathbf{P}_{Y,X}\mathbf{P}_{X,Y}$.

In the ideal case, this transport from elements in one sequence to another and back again should return to the same starting element. From (12) this corresponds to $\mathbf{P}_{Y,X}\mathbf{P}_{X,Y}$ equaling the identity matrix, $\mathbf{I}_{M\times M}$. Thus, our global cycle-consistency loss is the sum of cross-entropy losses:

$$\mathcal{L}_{\text{GCC}}(\mathbf{X},\mathbf{Y}) = -\sum_{i=1}^{M} \log((\mathbf{P}_{Y,X}\mathbf{P}_{X,Y})_{i,i}). \tag{13}$$

The right panel in Fig. 2 provides a summary of our global cycle-consistency loss.

3.5. Training and implementation details

Our final loss function is obtained by combining the contrastive alignment loss, (9), and the global cycle consistency loss, (13), according to

$$\begin{aligned} \mathcal{L}(\mathbf{X},\mathbf{Y}) &= \lambda_g \mathcal{L}_{\text{GCC}}(\mathbf{X},\mathbf{Y}) \\ &\quad + \lambda_s(\mathcal{L}_{\text{smoothDTW}}(\mathbf{X},\mathbf{Y}) + \mathcal{L}_{\text{smoothDTW}}(\mathbf{Y},\mathbf{X})), \end{aligned} \tag{14}$$

where λ_g and λ_s are weights used to balance the two losses and are empirically set to 1.0 and 0.1, respectively. The temperature hyper-parameters, γ and β used in (5) and (8) are both set to 0.1, while α in (10) is 1.

This overall loss is used to train a convolutional architecture composed of a backbone encoder applied framewise followed by an embedding network. Specifically, ResNet50-v2 [24] is used as our backbone encoder where we extract features from `Conv4c` layer. We adopt the same embedding network used in previous related work [15] comprised of two 3D convolutional layers, a global 3D max pooling layer, two fully connected layers, and a linear projection layer. The final embedding is L2 normalized. To learn over sequence pairs $(\mathbf{X},\mathbf{Y})$, we randomly extract $T = 20$ frames from each sequence. Sampling of video frames is random to avoid learning potential trivial solutions that may arise from strided sampling. For a fair comparison to previous approaches using the same architecture (*i.e*., [15, 40, 32]), we use the same batch size of four sequences. Finally, our learning rate is fixed to 10^{-4} for all our experiments.

4. Empirical evaluation

We evaluate the efficacy of our learned embeddings on challenging temporal fine-grained tasks, thereby going beyond traditional clip-level recognition tasks. In particular, our proposed loss is evaluated on fine-grained action recognition (*i.e*., action phase classification), few-shot fine-grained classification, and video synchronization. In addition, we also show that learning to align temporal sequences supports different downstream applications such as synchronous playback, 3D pose reconstruction and fine-grained audio/visual retrieval.

4.1. Datasets

To evaluate our method, we use the PennAction [58] and FineGym [41] video datasets. Both datasets contain a diverse set of videos of human-related sports or fitness activities. These datasets are selected as they allow for learning framewise alignments and evaluating on tasks where fine-grained temporal distinctions are critical.

PennAction [58] contains 2326 videos of humans performing 15 different sports or fitness actions. The videos are tightly cropped temporally around the start and end of the action. Notably, 13 categories contain non-repetitive actions. The dataset also includes ground truth 2D keypoint labels which we later use to demonstrate a 3D pose reconstruction application grounded on our alignment method.

FineGym [41] is a recent large-scale fine-grained action recognition dataset that was specifically designed to evaluate the ability of an algorithm to parse and recognize the different phases of an action. Each video in FineGym is annotated according to a three-level hierarchy denoting the *event* being performed in the video, the different *sets* involved in performing the event, and the framewise *elements* (*i.e*., action phases) involved in each set. To perform any *event-level* action, a gymnast may perform the different sets in any order. To train embeddings using our alignment-based method, we re-organize the FineGym dataset such that all sets belonging to the same event appear in the same order in any given video. An example of this re-organization is provided in the supplemental. Also, it should be noted that while the videos of the vault event (VT) in FineGym depict the gymnast performing the action in three phases, the first two phases of the action are not explicitly labeled in the original dataset. For the sake of completeness we use the provided start and end times for these phases, thereby adding two new fine-grained action labels in FineGym. The remaining events in FineGym are otherwise unchanged. To account for these minor additions to the annotations, we refer to the extensions of FineGym99 and 288 as FineGym101 and FineGym290, respectively. Notably, we also report results on the original FineGym99 and 288 in the supplemental. Importantly, we do not use the *element-level* (*i.e*., action phase) labels during training.

	Contrastive-Cost	GCC	FineGym101
SmoothDTW	-	-	28.20
	-	✓	28.20
	✓	-	47.32
	✓	✓	**49.51**
$\min^{\gamma}$	✓	✓	48.07

Table 1. Ablation study of the various components of our loss function, (14). Contrastive-Cost refers to our columnwise contrastive cost, (8), and GCC refers to our global cycle consistency loss, (13).

4.2. Baselines

We compare our approach to other weakly supervised [15, 40, 8] and self-supervised [3, 32] methods that entail temporal understanding in their definition. A detailed description of the baselines is provided in the supplemental.

4.3. Ablation study

We first present an ablation study that validates the contribution of each component of our loss. For this purpose, we evaluate fine-grained action recognition performance on FineGym101. Following previous work [15], we use a Support Vector Machine (SVM) classifier [12] on top of the learned embeddings to report framewise fine-grained classification accuracy. Notably, the classifier is trained on the extracted embeddings with no additional fine-tuning of the network. The results in Table 1 show the pivotal role of our contrastive cost. In fact, turning off the contrastive component of our cost, (8), and simply relying on the cosine distance always leads to no improvement in learning from the onset of training. Also, these results show the advantage of adding our global cycle consistency, which further validates the correspondences. Finally, we also compare the performance of our smoothMin definition vs. the more widely used $\min^{\gamma}$. The superiority of the adopted smooth definition supports the laid out arguments in Section 3.2.

4.4. Fine-grained action recognition

We now compare our fine-grained action recognition performance to our baselines using the FineGym dataset and consider two training settings for the backbone framewise encoder. **(i) scratch:** the backbone ResNet50 [24] is trained from scratch with our proposed loss, **(ii) only-bn:** we fine-tune batch norm layers of ResNet50 from a model pre-trained on ImageNet [37]. The embedder is otherwise trained from scratch in both cases. A third setting where all layers are fine-tuned is presented in the supplemental.

Method	Training	FineGym101	FineGym290
SpeedNet [3]	scratch	30.40	29.87
TCN [40]		36.52	37.40
SaL [32]		40.25	37.98
D^3TW^* [8]		32.10	32.15
TCC [15]		41.78	40.57
Ours		**45.79**	**43.49**
SpeedNet [3]	only bn	34.38	35.92
TCN [40]		41.75	39.93
SaL [32]		42.68	41.58
D^3TW^* [8]		38.21	34.04
TCC [15]		45.62	43.40
Ours		**49.51**	**46.54**

Table 2. Fine-grained action recognition accuracy on both organizations of FineGym.

The results summarized in Table 2 (and the supplemental) speak decisively in favour of our method, where we outperform all other weakly and self-supervised methods with sizable margins. The gap is especially striking in the case of SpeedNet. This poor performance can largely be attributed to the fact that the task optimized in SpeedNet does not require detailed framewise understanding. On the other hand, the closest approach to ours is TCC as it also relies on pairwise local matchings between videos of the same class; however, the matchings are realized independently and thus ignores informative long-term sequence structure. This is in contrast to SaL which uses frames from the same videos to solve tasks requiring temporal understanding. Importantly, the superiority of our results compared to TCC demonstrates that the global nature of our loss makes the learned embeddings more robust to the presence of repeated sub-actions as is the case for most of the FineGym videos. Interestingly, the results obtained with D^3TW^* highlight the limits of the downstream discriminative loss, which requires an explicit construction of positive and negative examples for training. Notably, while our method outperforms all alternatives under both training settings, the best overall results are obtained under the *only-bn* setting and it is therefore used for all other experiments reported in this paper.

We also considered classification results of each event separately where we also outperformed all alternatives. Importantly, visualizations of the learned features suggests that the proposed loss learns to adapt and identify the most reliable cues to learn the alignments. Please see supplemental for detailed results, discussions, and visualizations.

4.5. Few-shot fine-grained action recognition

An advantage of our proposed weakly supervised method is that it does not rely on framewise labels for training. To evaluate this advantage, we also report few-shot classification results. In this case, the entire training set is used to learn the embeddings, but only a few videos per class are used to train the classifier. In particular, we use FineGym101 for this experiment with an increasing number of videos per class to train the classifier, starting from the 1-shot setting all the way to using the entire dataset.

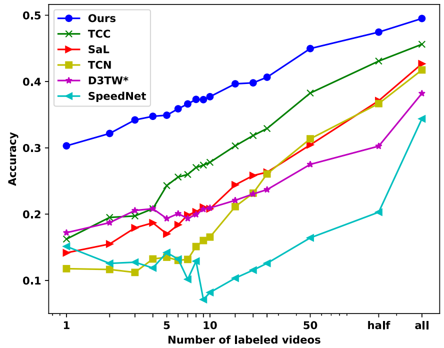

Figure 4. Few-shot fine-grained action recognition accuracy on FineGym101.

Method	Kendall's Tau	Phase classification
TCN [40]	65.29	69.3
SaL [32]	53.87	69.4
TCC [15]	71.0	77.51
Ours	**74.84**	**78.90**

Table 3. Video alignment results using Kendall's Tau metric and action phases classification on PennAction. Different from previous work, these results were obtained by training a single network for all classes, thereby learning a joint representation.

The plot in Fig. 4 further confirms the superiority of our proposed method. As can be seen, our method outperforms all others across the range of number of labeled training videos used, with an especially strong performance even under the challenging 1-shot setting.

4.6. Video synchronization

To evaluate the quality of the synchronization (*i.e.*, alignment) between two videos we use Kendall's Tau metric, which does not require framewise labels. However, this metric assumes little to no repetitions in the aligned videos. We therefore follow previous work [15] and report results only on the 13 classes without repetition in PennAction (*i.e.*, strumming guitar and jumping rope are not included). Importantly, while previously reported results [15] were obtained by training a different network for each class of PennAction, we consider the more challenging setting of learning a joint representation over all classes (*i.e.*, as done in all previous experiments with FineGym). For a fair comparison, all baselines are also trained over all classes of PennAction rather than one network per class.

Table 3 summarizes our video alignment results, where we once again achieve state-of-the-art performance compared to the baselines. Importantly, our performance is also superior to previously reported results under the multi-network training setting [15]. We report results under this setting in the supplemental.

In addition to the quality of synchronization, we also re-

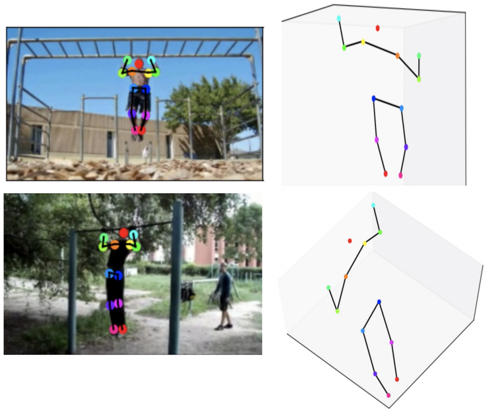

Figure 5. 3D pose reconstruction results derived from the learned alignments. Synchronized video frames from two sample training videos in the pullup category of PennAction with overlayed 2D keypoints are shown on the left. Reconstructed 3D pose from two different viewpoints is shown on the right.

port in Table 3 state-of-the-art results on action phase classification. Notably, the action phase labels used here are provided by the original authors of the PennAction dataset [58] as the labels used previously [15] were not made publically available.

5. Downstream applications

3D pose reconstruction. As demonstrated by our evaluations, our proposed weakly-supervised method is capable of temporally aligning videos of similar actions even while they are captured in different environments, at different execution rates, and from different viewpoints. As a result, we can align poses of the same action from different viewpoints by forcing different videos to play synchronously. Examples of this video synchronization are provided in the supplemental. Importantly, the synchronization of videos taken at various viewpoints can serve as the basis for 3D pose reconstruction. To demonstrate this ability, we use videos from PennAction and their corresponding ground truth 2D keypoint labels. In particular, given a random query video from a given class in PennAction, we first start by aligning the remaining videos (from the same class) to it. Given these aligned frames and their corresponding 2D keypoints we use the Tomasi-Kanade factorization algorithm [46], followed by bundle adjustment [22], to compute a temporally aligned 3D model of the action performed. Fig. 5 presents an example of our 3D pose reconstructions; additional results are available in the supplemental material.

Audio-visual alignment. Finally, we demonstrate that our alignment based representation learning method can be applied to other types of sequences by applying it on separate audio and visual inputs. Given that any audio-visual pair is by default aligned and to avoid learning trivial solutions, we sample audio segments and video frames differently to make the task of learning the alignment harder on the networks and consequently learn strong audio and visual embeddings. In particular, while the audio signal is uniformly sampled into consecutive one second long segments, video frames are on the other hand randomly sampled along the temporal dimension. This sampling strategy encourages our model to learn different alignment paths due to the randomness in the video frame selection. For visual features, the same backbone and embedding network described in Sec. 3.5 is used to encode video frames, while we use VGGish [18] to encode audio signals. In particular, each one second long audio segment is first converted into a log mel spectogram and used as an input for the VGGish network. Training is otherwise performed as described in Sec. 3.5.

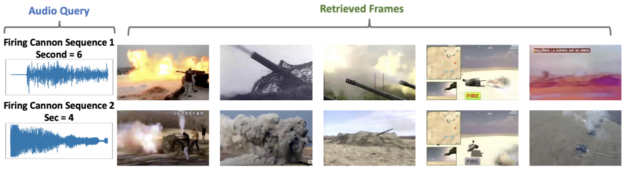

Figure 6. Sample fine-grained audio/visual retrieval using nearest neighbor matches in embedding space, showing that the audio of the explosion can be used to retrieve corresponding visual event.

For this sample application, we use the firing cannon class from the VGGSound dataset [9] for training and testing. This class is selected as it is strongly visually indicated with an easily identifiable salient auditory signal (*i.e.*, the explosion sound emitted upon firing of a cannon).

To demonstrate the quality of the learned audio and visual embeddings, we evaluate them on the task of fine-grained audio/visual retrieval. In particular, given a one second long audio signal corresponding to the moment of firing a cannon, we extract its corresponding audio embedding from the VGGish network trained using our approach. This embedding is then used to query corresponding visual embeddings from the vision network. The top-5 nearest frame embeddings from all videos in the test set are extracted. Sample correspondences, shown in Fig. 6, clearly depict that the corresponding visual embeddings also capture the moment of the firing visually. More details and sample results are presented in the supplemental.

6. Conclusion

In summary, this work introduced a novel weakly supervised method for representation learning relying on sequence alignment as a supervisory signal and taking a probabilistic view in tackling this problem. Because the latent supervisory signal entails detailed temporal understanding, we judge the effectiveness of our learned representation on tasks requiring fine-grained temporal distinctions and show that we establish a new state of the art. In addition, we present two applications of our temporal alignment framework, thereby opening up new avenues for future investigations grounded on the proposed approach.

References

[1] Unaiza Ahsan, Rishi Madhok, and Irfan A. Essa. Video jigsaw: Unsupervised learning of spatiotemporal context for video action recognition. In *WACV*, pages 179–189, 2019. 2

[2] Richard Bellman. On the theory of dynamic programming. *PNAS*, 8(38):716–719, 1952. 3

[3] 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 *CVPR*, pages 9919–9928, 2020. 2, 6, 7

[4] Uta Büchler, Biagio Brattoli, and Björn Ommer. Improving spatiotemporal self-supervision by deep reinforcement learning. In *ECCV*, pages 797–814, 2018. 2

[5] Xingyu Cai, Tingyang Xu, Jinfeng Yi, Junzhou Huang, and Sanguthevar Rajasekaran. DTWNet: A dynamic time warping network. In *NeurIPS*, pages 11636–11646, 2019. 2

[6] Kaidi Cao, Jingwei Ji, Zhangjie Cao, Chien-Yi Chang, and Juan Carlos Niebles. Few-shot video classification via temporal alignment. In *CVPR*, pages 10615–10624, 2020. 1, 2, 3, 5

[7] João Carreira and Andrew Zisserman. Quo vadis, action recognition? A new model and the kinetics dataset. In *CVPR*, pages 4724–4733, 2017. 2

[8] Chien-Yi Chang, De-An Huang, Yanan Sui, Li Fei-Fei, and Juan Carlos Niebles. D3TW: Discriminative differentiable dynamic time warping for weakly supervised action alignment and segmentation. In *CVPR*, pages 3546–3555, 2019. 1, 2, 3, 5, 6, 7

[9] Honglie Chen, Weidi Xie, Andrea Vedaldi, and Andrew Zisserman. Vggsound: A large-scale audio-visual dataset. In *ICASSP*, pages 721–725, 2020. 8

[10] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020. 2

[11] Soo-Whan Chung, Joon Son Chung, and Hong-Goo Kang. Perfect match: Improved cross-modal embeddings for audio-visual synchronisation. In *ICASSP*, pages 3965–3969, 2019. 3

[12] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995. 6

[13] Marco Cuturi and Mathieu Blondel. Soft-DTW: A differentiable loss function for time-series. In *ICML*, pages 894–903, 2017. 2, 3

[14] Alexey Dosovitskiy, Philipp Fischer, Jost Tobias Springenberg, Martin A. Riedmiller, and Thomas Brox. Discriminative unsupervised feature learning with exemplar convolutional neural networks. *TPAMI*, 38(9):1734–1747, 2016. 2

[15] Debidatta Dwibedi, Yusuf Aytar, Jonathan Tompson, Pierre Sermanet, and Andrew Zisserman. Temporal cycle-consistency learning. In *CVPR*, pages 1801–1810, 2019. 1, 2, 3, 5, 6, 7, 8

[16] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. SlowFast networks for video recognition. In *ICCV*, pages 6201–6210, 2019. 2

[17] Basura Fernando, Hakan Bilen, Efstratios Gavves, and Stephen Gould. Self-supervised video representation learning with odd-one-out networks. In *CVPR*, pages 5729–5738, 2017. 2

[18] Jort F. Gemmeke, Daniel P. W. Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R. Channing Moore, Manoj Plakal, and Marvin Ritter. Audio set: An ontology and human-labeled dataset for audio events. In *ICASSP*, pages 776–780, 2017. 8

[19] Daniel Gordon, Kiana Ehsani, Dieter Fox, and Ali Farhadi. Watching the world go by: Representation learning from unlabeled videos. *CoRR*, abs/2003.07990, 2020. 2

[20] Michael Gutmann and Aapo Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *AISTATS*, pages 297–304, 2010. 2

[21] Tengda Han, Weidi Xie, and Andrew Zisserman. Video representation learning by dense predictive coding. In *ICCVW*, pages 1483–1492, 2019. 2

[22] Richard Harltey and Andrew Zisserman. *Multiple view geometry in computer vision (2ed.)*. Cambridge University Press, 2004. 8

[23] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross B. Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, pages 9726–9735, 2020. 2

[24] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 5, 6

[25] Allan Jabri, Andrew Owens, and Alexei A. Efros. Space-time correspondence as a contrastive random walk. In *NeurIPS*, pages 19545–19560, 2020. 2

[26] Joel Janai, Fatma Güney, Anurag Ranjan, Michael J. Black, and Andreas Geiger. Unsupervised learning of multi-frame optical flow with occlusions. In *ECCV*, pages 713–731, 2018. 2

[27] Mandy Lange, Dietlind Zühlke, Olaf Holz, and Thomas Villmann. Applications of lp-norms and their smooth approximations for gradient based learning vector quantization. In *ESANN*, 2014. 3

[28] Hsin-Ying Lee, Jia-Bin Huang, Maneesh Singh, and Ming-Hsuan Yang. Unsupervised representation learning by sorting sequences. In *ICCV*, pages 667–676, 2017. 2

[29] Simon Meister, Junhwa Hur, and Stefan Roth. UnFlow: Unsupervised learning of optical flow with a bidirectional census loss. In *AAAI*, pages 7251–7259, 2018. 2

[30] Arthur Mensch and Mathieu Blondel. Differentiable dynamic programming for structured prediction and attention. In *ICML*, pages 3459–3468, 2018. 1, 2

[31] Antoine Miech, Jean-Baptiste Alayrac, Lucas Smaira, Ivan Laptev, Josef Sivic, and Andrew Zisserman. End-to-end learning of visual representations from uncurated instructional videos. In *CVPR*, pages 9876–9886, 2020. 3

[32] Ishan Misra, C. Lawrence Zitnick, and Martial Hebert. Shuffle and learn: Unsupervised learning using temporal order verification. In *ECCV*, pages 527–544, 2016. 2, 6, 7

[33] Andriy Mnih and Koray Kavukcuoglu. Learning word embeddings efficiently with noise-contrastive estimation. In *NeurIPS*, pages 2265–2273, 2013. 2

[34] Andrew Owens and Alexei A. Efros. Audio-visual scene analysis with self-supervised multisensory features. In *ECCV*, pages 639–658, 2018. 2

[35] Deepak Pathak, Ross B. Girshick, Piotr Dollár, Trevor Darrell, and Bharath Hariharan. Learning features by watching objects move. In *CVPR*, pages 6024–6033, 2017. 2

[36] Mandela Patrick, Yuki Markus Asano, Ruth Fong, João F. Henriques, Geoffrey Zweig, and Andrea Vedaldi. Multimodal self-supervision from generalized data transformations. *CoRR*, abs/2003.04298, 2020. 3

[37] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *IJCV*, 115(3):211–252, 2015. 6

[38] Hiroaki Sakoe and Seibi Chiba. Dynamic programming algorithm optimization for spoken word recognition. *TASSP*, 26(1):43–49, 1978. 2, 3

[39] Yasushi Sakurai, Christos Faloutsos, and Masashi Yamamuro. Stream monitoring under the time warping distance. In *ICDE*, pages 1046–1055, 2007. 3

[40] Pierre Sermanet, Corey Lynch, Yevgen Chebotar, Jasmine Hsu, Eric Jang, Stefan Schaal, and Sergey Levine. Time-contrastive networks: Self-supervised learning from video. In *ICRA*, pages 1134–1141, 2018. 2, 6, 7

[41] Dian Shao, Yue Zhao, Bo Dai, and Dahua Lin. FineGym: A hierarchical video dataset for fine-grained action understanding. In *CVPR*, pages 2613–2622, 2020. 6

[42] Gunnar A. Sigurdsson, Abhinav Gupta, Cordelia Schmid, Ali Farhadi, and Karteek Alahari. Actor and observer: Joint modeling of first and third-person videos. In *CVPR*, pages 7396–7404, 2018. 2

[43] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In *NeurIPS*, pages 568–576, 2014. 2

[44] Chen Sun, Fabien Baradel, Kevin Murphy, and Cordelia Schmid. Contrastive bidirectional transformer for temporal representation learning. *CoRR*, abs/1906.05743, 2019. 3

[45] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. *CoRR*, abs/1906.05849, 2019. 2, 3

[46] Carlo Tomasi and Takeo Kanade. Shape and motion from image streams under orthography: A factorization method. *IJCV*, 9(2):137–154, 1992. 8

[47] Du Tran, Lubomir D. Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3D convolutional networks. In *ICCV*, pages 4489–4497, 2015. 2

[48] Aäron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *CoRR*, abs/1807.03748, 2018. 2

[49] Carl Vondrick, Hamed Pirsiavash, and Antonio Torralba. Anticipating visual representations from unlabeled video. In *CVPR*, pages 98–106, 2016. 2

[50] Carl Vondrick, Abhinav Shrivastava, Alireza Fathi, Sergio Guadarrama, and Kevin Murphy. Tracking emerges by colorizing videos. In *ECCV*, pages 402–419, 2018. 2

[51] Jiangliu Wang, Jianbo Jiao, Linchao Bao, Shengfeng He, Yunhui Liu, and Wei Liu. Self-supervised spatio-temporal representation learning for videos by predicting motion and appearance statistics. In *CVPR*, pages 4006–4015, 2019. 2

[52] Xiaolong Wang and Abhinav Gupta. Unsupervised learning of visual representations using videos. In *ICCV*, pages 2794–2802, 2015. 2

[53] Xiaolong Wang, Allan Jabri, and Alexei A. Efros. Learning correspondence from the cycle-consistency of time. In *CVPR*, pages 2566–2576, 2019. 2

[54] Donglai Wei, Joseph J. Lim, Andrew Zisserman, and William T. Freeman. Learning and using the arrow of time. In *CVPR*, pages 8052–8060, 2018. 2

[55] Dejing Xu, Jun Xiao, Zhou Zhao, Jian Shao, Di Xie, and Yueting Zhuang. Self-supervised spatiotemporal learning via video clip order prediction. In *CVPR*, pages 10334–10343, 2019. 2

[56] Serena Yeung, Olga Russakovsky, Ning Jin, Mykhaylo Andriluka, Greg Mori, and Li Fei-Fei. Every moment counts: Dense detailed labeling of actions in complex videos. *IJCV*, 126(2-4):375–389, 2018. 1

[57] Jason J. Yu, Adam W. Harley, and Konstantinos G. Derpanis. Back to basics: Unsupervised learning of optical flow via brightness constancy and motion smoothness. In *ECCVW*, pages 3–10, 2016. 2

[58] Weiyu Zhang, Menglong Zhu, and Konstantinos G. Derpanis. From actemes to action: A strongly-supervised representation for detailed action understanding. In *ICCV*, pages 2248–2255, 2013. 6, 8