Weakly-Supervised Representation Learning for Video Alignment and Analysis

Guy Bar-Shalom*[1]
Verily
guy.b@cs.technion.ac.il

George Leifman
Verily
gleifman@google.com

Michael Elad
Verily
melad@google.com

Abstract

Many tasks in video analysis and understanding boil down to the need for frame-based feature learning, aiming to encapsulate the relevant visual content so as to enable simpler and easier subsequent processing. While supervised strategies for this learning task can be envisioned, self and weakly-supervised alternatives are preferred due to the difficulties in getting labeled data. This paper introduces LR-Prop – a novel weakly-supervised representation learning approach, with an emphasis on the application of temporal alignment between pairs of videos of the same action category. The proposed approach uses a transformer encoder for extracting frame-level features, and employs the DTW algorithm within the training iterations in order to identify the alignment path between video pairs. Through a process referred to as "pair-wise position propagation", the probability distributions of these correspondences per location are matched with the similarity of the frame-level features via KL-divergence minimization. The proposed algorithm uses also a regularized SoftDTW loss for better tuning the learned features. Our novel representation learning paradigm consistently outperforms the state of the art on temporal alignment tasks, establishing a new performance bar over several downstream video analysis applications.

1. Introduction

As in many other domains, deep learning techniques have brought a revolution to the field of video analysis and understanding in the past several years [23, 24, 41, 42]. Applications such as video classification [19, 43, 46], action detection [18, 25, 47], video captioning [33, 35], forecasting [6, 44], and many others, all have been getting new and highly effective AI-based solutions with unprecedented performance. Interestingly, within this impressive progress, the task of temporal alignment of video pairs has received relatively little attention. This paper offers a novel weakly-supervised approach towards representation learning for video, focusing on the temporal alignment application.

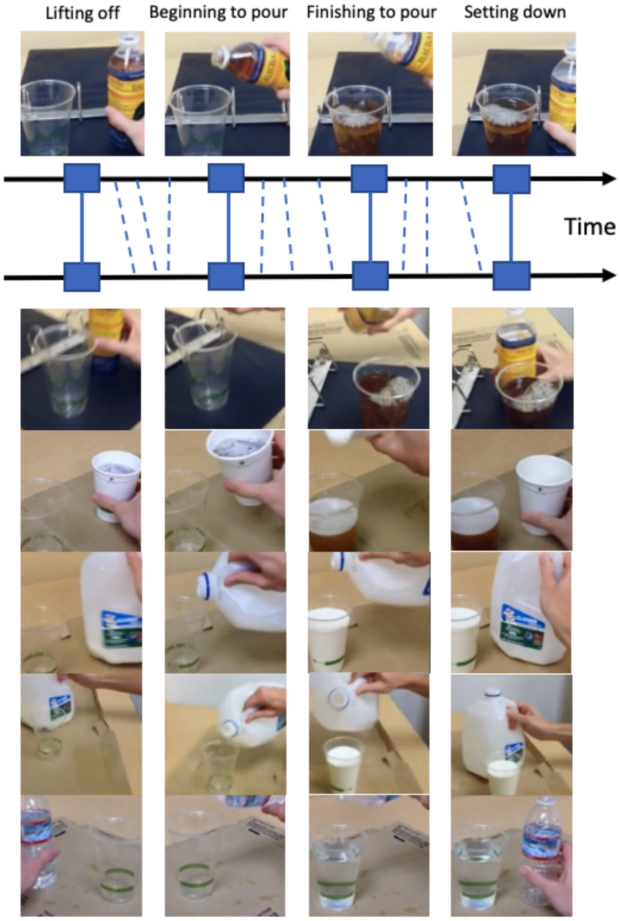

Figure 1. Video alignment (**Pouring** dataset) using **LRProp** features and the DTW algorithm. The first row shows a selected set of key events in a randomly selected video; the bottom shows the alignment results of these events with five other randomly chosen videos. As can be seen, **LRProp** leads to a successful capture of the key events in the query. Note that we show a single time-line for all the five selected videos for simplicity.

[*] Accomplished during Guy Bar-Shalom's internship at Verily.
[1] Technion - Israel Institute of Technology, Department of Computer Science.

Many events occur in a specific temporal sequence, such as specific actions in sport activity (e.g., baseball swing), portions of a person's daily routine, various repetitive medical procedures, sea tides, intervals within traffic control videos, or even simple actions such as pouring a glass of fluid. In all these and other cases, videos that capture such visual content contain not only information about the cause and effect of these events, but also the potential for temporal correspondences across multiple instances of the same process. For example, the key moment of reaching for a container and lifting it off the ground are common to all pouring sequences, despite differences in visual factors such as container style, illumination, surrounding items, and event speed. Broadly speaking, a reliable alignment of such videos can open the door to new abilities in video analysis, such as detecting anomaly behavior, enabling quick search of specific sub-events, identifying the phase within the whole event, measuring distances between videos for their clustering, and so much more.

So, how can videos be aligned? Earlier work [7,14,30] suggests to address this task by first learning spatio-temporal feature representations for each video frame. Given such representations, their sequential matching over time, which takes into account temporal correspondences, would result in the desired alignment [14, 20, 34]. This temporal matching could be achieved in a variety of ways, ranging from the simple nearest neighbor search, all the way to Dynamic Time Warping, DTW [2]. The sought representations should capture key visual information while discarding irrelevant details, and do this while also providing a dimensionality reduction for ease of later processing.

Representation learning could be achieved in a supervised fashion when frame-by-frame alignment is readily available. In this case, our task simply involves learning a common embedding space from pairs of aligned frames. A few approaches have been proposed for such supervised action recognition and segmentation [4,15,38]. However, in many real-world sequences, such frame-by-frame alignment does not exist naturally. Obtaining labels for every frame in a video might be time-consuming and ill-defined task, as it is unclear what set of labels would be necessary for a complete understanding of the fine-grained details of the video. A possible remedy to the above could be to artificially obtain aligned sequences by recording the same event from multiple cameras, but this method may not capture all the variations present in naturally occurring videos. All this brings us to self - or weak-supervision alternatives, as practiced in [7,13,20,26,28]. These methods may rely on video augmentation or use a SoftDTW [11] loss for training the representations.

Inspired by the above described work, we present **LRProp** (*Learning Representations by position PROPagation*) – a novel weakly-supervised approach that does not require explicit frame correspondences between different video sequences. To accomplish our goal, we adopt the transformer encoder [39], which has demonstrated effectiveness in extracting meaningful frame-level features, as shown in [1,7,31]. Our experiments suggest that the transformer encoder is particularly beneficial for video alignment, and we believe that this is due to its positional encoding. In each step within the learning process, our method involves taking pairs of videos that depict the same action category, feeding them into the transformer encoder to generate frame-level features, and using the DTW algorithm [2] to produce a path aligning the two. We use this alignment path to define a *pair-wise position propagation* (definitions follow), which we utilize to establish a soft one-to-one linkage between pairs of frames. This *pair-wise position propagation* is used to define a prior distribution for the correspondence between two distinct frames from different videos, and we minimize the KL-divergence between this prior and a probability distribution over the similarity of the frame-level features (extracted by the transformer encoder) of the specified pair of frames. Our learning also employs the SoftDTW algorithm [11] with appropriate regularization to prevent trivial solutions, and to learn a more accurate alignment path for pairs of videos. Figure 1 demonstrates temporal alignment using the features extracted by the proposed method.

To summarize, our contributions include the following, 1) we present a general weakly-supervised framework for learning frame-wise representations with a focus on video alignment. 2) The proposed *pair-wise position propagation* is shown to result in features that offer better temporal awareness compared to prior work. 3) Our approach achieves superior performance to the state-of-the-art on various temporal understanding tasks on the Pouring [36] and PennAction [49] datasets, setting a new performance benchmark for downstream tasks.

2. Related Work

Since labeled data can be expensive and time-consuming to collect and annotate, and may not always be available in sufficient quantities for certain tasks, self-supervised learning has become a widely studied area in the field of deep learning. As a result, numerous pretext tasks have been proposed for image-based methods to achieve self-supervision. These include relative patch prediction [12], jigsaw puzzle solving [32], colorization [48], rotation prediction [10], instance discrimination using strong data augmentation [8], knowledge distillation [3,9], and more. These methods and many others have been shown to be highly effective in various downstream tasks. In this study, we investigate the use of self-supervised and weakly-supervised (definitions follow) learning techniques to create representations from videos, taking advantage of both the spatial and temporal information contained within the video data.

As we transition from images to video, various supporting tasks that produce supervision signals have been employed in representation learning for video frames. These include predicting the sequence of frames in a video [37, 40] or predicting audio from video [45]. Recently, the incorporation of temporal ordering has been demonstrated to be a strong pretext task to obtain meaningful video representations. Sermanet et al. [36] proposed Time-Contrastive Networks (**TCN**), which uses attraction and repulsion between temporally close and far frames, respectively, in order to learn useful features. **Sal** [30] proposed to learn features by sampling tuples of frames and predicting whether the tuple is in the correct temporal order (obtaining the labels using the frame indices). Another related work (**TCC**) was proposed by [14], which learns representations by finding frame correspondences across videos.

In this context, we should mention two recent works, **SCL** and **VAVA** [7, 27], which also achieved impressive results in self-supervised learning for videos. Chen et al. [7] proposed strongly augmenting the input both temporally and spatially, and using a transformer model [39] as an encoder, which has been used for videos in recent works [1, 31]. They also used a contrastive loss function to encourage the embedding of nearby frames to be more similar than those that are far apart. Liu et al. [27] attempted to learn video representations while considering the possibility of background frames, redundant frames, and non-monotonic frames when aligning two videos in time.Whereas some work, as the above, assumes a self-supervised setting, in this paper we consider a weakly-supervised alternative, as described in [5]. More specifically, we refer to cases in which the videos of interest consist of the same action category sequence. In these cases, we are given an ordered list of actions during training, but the exact temporal boundaries or paste of each action are not provided. For example, in a video of pouring wine, the weak supervision might include the sequence "take the bottle, pour the wine, place the bottle back." This leads us to a powerful pretext task known as temporal video alignment, in which multitude of such corresponding videos can be leveraged for supervised learning.

Though there is a significant amount of literature on time series alignment, only a few of these ideas have been applied to aligning videos. While traditional methods for time series alignment, such as DTW [2], are not differentiable and therefore cannot be used directly for training neural networks, a smooth approximation of DTW, called SoftDTW, was introduced in [11]. Several recent papers [17, 20] attempted to apply a soft DTW approximation in video representations learning. As an example, we mention Haresh et al. [20], which introduced a technique called **LAV** (Learning by Aligning Videos in Time) for learning representative frame embeddings by aligning videos in time using SoftDTW during training. However, their method, and other methods that use a soft DTW approximation during training, do not take advantage of the alignment path explicitly during optimization. In contrast, our approach, as unfolded in the next section, integrates both the SoftDTW cost function and the DTW alignment path into the optimization process.

3. Method

In this section we present *Learning Representations by position PROPagation*, **LRProp**, a framework for learning frame-wise video representations. We learn an embedding space where videos with similar content can be aligned in time; this setting is commonly referred to as weakly-supervised learning, as discussed in Section 2. More specifically, our method involves taking pairs of videos that depict the same action category, feeding them into the transformer encoder to generate frame-level features for each, and using the DTW algorithm [2] to produce a path aligning the two. We use this alignment to define a *pair-wise position propagation*, and establish a soft one-to-one linkage between pairs of frames. We minimize the KL-divergence between a reference distribution and the probability function over the similarity of the frame-level features of the specified pairs of frames. We also use the SoftDTW algorithm [11] with appropriate regularization to prevent trivial solutions as part of our training process. A visualization of our pair-wise position propagation method is depicted in Figure 2. In what follows, we detail each of the above-described ingredients.

3.1. Notations and Definitions

We begin by introducing the necessary notations and definitions for our discussion. Let $\mathcal{V}^1$ and $\mathcal{V}^2$ be a pair[1] of videos, where each is represented as $\mathcal{V}^i \in \mathbf{R}^{F_i \times C \times W \times H}$. Here, F_i represents the number of frames in the ith video and C, W, and H are the number of channels, width, and height of each frame, respectively. To begin, we perform the same random sampling and data augmentation process described in [7]. This process takes a video $\mathcal{V}^i$ and uses temporal random cropping to generate two cropped videos of length T, (V_1^i, S_1^i) and (V_2^i, S_2^i), where T is a hyperparameter, and $S_{1/2}^i \in \mathbf{R}^T$ hold the frame indices. Next, several temporal-consistent spatial data augmentations are performed on each of the two sampled videos. After this step on $\mathcal{V}^1$ and $\mathcal{V}^2$, we are left with $(V_1^1, S_1^1), (V_2^1, S_2^1)$, and

[1] Although discussing video pairs per training step, our approach extends readily to any number of videos.

$(V_1^2, S_1^2), (V_2^2, S_2^2)$. We will use hereafter capital Latin letters to index the different videos, and greek letters to index the samplings.

We define a neural model, $f_\theta : \mathbf{R}^{T \times C \times W \times H} \rightarrow \mathcal{Z}$, which maps videos from an input space to an embedding space $\mathcal{Z}$[1]. We adapt the transformer model [39] used by [7]. Assuming the existence of a prior distribution that represents the similarity between a frame, indexed $[S_\alpha^{\mathrm{A}}]_i$, and a given frame, indexed $[S_\beta^{\mathrm{B}}]_j$, which we define by $p_{\mathrm{A},\mathrm{B},\alpha,\beta}(i|j)$, our goal is to enforce the model's embedding ($f_\theta(V) \triangleq Z$) similarity to follow this distribution. We propose to achieve this by minimizing the KL-divergence between p and the following distribution:

$$q_{\theta,\mathrm{A},\mathrm{B},\alpha,\beta}(i|j) \equiv \mathcal{Q}_\theta(i|j) = \frac{\exp(\mathrm{sim}([Z_\beta^{\mathrm{B}}]_j, [Z_\alpha^{\mathrm{A}}]_i)/\tau)}{\sum_{i'=1}^{T} \exp(\mathrm{sim}([Z_\beta^{\mathrm{B}}]_j, [Z_\alpha^{\mathrm{A}}]_{i'})/\tau)}, \tag{1}$$

where sim denotes the cosine similarity ($\mathrm{sim}(\mathbf{u}, \mathbf{v}) = \mathbf{u}^T\mathbf{v}/\|\mathbf{u}\|\|\mathbf{v}\|$) and τ is a hyper-parameter controlling the smoothness of the distribution. The question remains, what would be a good prior for a given pair of videos?

3.2. Prior Distribution for Frames in the Same Video

Let $(Z_\alpha^{\mathrm{A}}, S_\alpha^{\mathrm{A}}), (Z_\beta^{\mathrm{B}}, S_\beta^{\mathrm{B}})$ be a given pair of embeddings, and their corresponding chosen indices. Following [7], in the case where $\mathrm{A} = \mathrm{B}$ (i.e., two sampled versions of the same video), the prior distribution is chosen as

$$p_{\mathrm{A},\mathrm{B},\alpha,\beta}(i|j; \mathrm{A}=\mathrm{B}) \equiv \mathcal{P}_{\mathrm{A}=\mathrm{B}}(i|j) = \frac{\exp(-([S_\beta^{\mathrm{B}}]_j - [S_\alpha^{\mathrm{A}}]_i)^2/2\sigma^2)}{\sum_{i'=1}^{T} \exp(-([S_\beta^{\mathrm{B}}]_j - [S_\alpha^{\mathrm{A}}]_{i'})^2/2\sigma^2)}. \tag{2}$$

For the frame j in video B sampled version β, this expression forms a Gaussian of width σ for nearby frames, centered around $[S_\beta^{\mathrm{B}}]_j$. Using this prior and optimizing the following objective,

$$\mathcal{L}_{\mathrm{A}=\mathrm{B}}^{j} = D_{KL}\big(\mathcal{P}_{\mathrm{A}=\mathrm{B}}(\cdot|j) \parallel \mathcal{Q}_\theta(\cdot|j)\big), \tag{3}$$

we enforce that the similarity between the embeddings of a given frame and its neighboring ones in the same video, follow such a Gaussian distribution. This assumption is reasonable, as adjacent frames are more correlated than far away ones. By accumulating the above over j,

$$\mathcal{L}_{\mathrm{Same}}(\mathcal{P}_{\mathrm{A}=\mathrm{B}}, \mathcal{Q}_\theta) = \frac{1}{T}\sum_{j=1}^{T} \mathcal{L}_{\mathrm{A}=\mathrm{B}}^{j}, \tag{4}$$

we obtain the first loss component, which considers two different sampled versions of the same video. Observe that we omit the indices α, β for simplicity.

3.3. Prior Distribution for Frames in Different Videos

In the case where $\mathrm{A} \neq \mathrm{B}$, it is more challenging to define such a prior distribution. To match frames between different videos, we propose to rely on the path extracted by the Dynamic Time Warping (DTW) algorithm. To get a better understanding of our proposed method, we introduce the necessary notations and definitions regarding DTW.

Given two sets of extracted embeddings, $Z^1 = \{z_1^1, z_2^1, \ldots, z_n^1\}$ and $Z^2 = \{z_1^2, z_2^2, \ldots, z_m^2\}$ from two input videos of lengths n and m, respectively, we can compute the instantaneous distance matrix $D \in \mathbf{R}^{n \times m}$, where each entry is defined as $D(i, j) = d(z_i^1, z_j^2)$. The function $d : \mathcal{Z} \times \mathcal{Z} \rightarrow \mathbf{R}$ is a generic distance measure, implemented in this paper using the l_2-norm. DTW computes the alignment loss between Z^1 and Z^2 by identifying the path of minimum cost in the distance matrix D:

$$\mathit{dtw}(Z_1, Z_2) = \mathit{min}_{\hat{A} \in \mathcal{A}_{n,m}} \langle \hat{A}, D \rangle. \tag{5}$$

The matrix $\mathcal{A}_{n,m} \subset \{0, 1\}^{n \times m}$ denotes the collection of all possible alignment matrices that correspond to paths from the top-left corner to the bottom-right corner of D, using only $\{\rightarrow, \searrow, \downarrow\}$ moves. The alignment matrix $\hat{A} \in \mathcal{A}_{n,m}$ is binary, where $\hat{A}(i, j) = 1$ indicates that the embedding z_i^1 from Z^1 is aligned with the embedding z_j^2 from Z^2. DTW can be computed using dynamic programming by applying the following recursive function,

$$\begin{aligned} r(i, j) \quad = \quad & D(i, j) + \\ & \min\{r(i-1, j), r(i, j-1), r(i-1, j-1)\}. \end{aligned} \tag{6}$$

where $r(i, j)$ represents the (accumulated) DTW distance between the two videos till their frames, i and j, respectively. The minimum function is taken over all possible pairs of previous elements in the two sequences.

Returning to our video embedding task, we propose to use the alignment matrix $\hat{A}$ to model the prior distribution for pairs of videos where $\mathrm{A} \neq \mathrm{B}$. By generalizing the expression in Equation (2), this prior is defined as follows:

$$p_{\mathrm{A},\mathrm{B},\alpha,\beta}(i|j; \mathrm{A} \neq \mathrm{B}) \equiv \mathcal{P}_{\mathrm{A}\neq\mathrm{B}}(i|j) = \frac{\exp(-([S_\beta^{\mathrm{B}}]_j - [S_\beta^{\mathrm{B}}]_{\mathrm{argmax}_k\, \hat{A}(k,i)})^2/2\sigma^2)}{\sum_{i'=1}^{T} \exp(-([S_\beta^{\mathrm{B}}]_j - [S_\beta^{\mathrm{B}}]_{\mathrm{argmax}_k\, \hat{A}(k,i')})^2/2\sigma^2)}. \tag{7}$$

Here, i is a frame in video A, and j is a frame in the video B. The alignment matrix $\hat{A}$ has rows corresponding to video B and columns corresponding to video A; therefore, $\mathrm{argmax}_k\, \hat{A}(k, i)$ is the index of the frame in video B that has the maximum alignment score (which is 1) with frame i in video A. We define this process as *pair-wise position propagation*. If there are multiple frames with the same

[1] In our implementation $\mathcal{Z} = \mathbf{R}^{T \times 128}$.

maximum alignment score, we use the frame with the smallest index. We should emphasize that in the learning process this probability distribution is not differentiated with respect to the matrix $\hat{A}$. Rather, this alignment matrix is updated during training (due to the modified representations), and considered as fixed when optimizing for the representations. Therefore, similarly to Equation (3), given a frame j in video B, we propose to optimize

$$\mathcal{L}^{j}_{\mathrm{A}\neq\mathrm{B}} = D_{KL}\big(\mathcal{P}_{\mathrm{A}\neq\mathrm{B}}(\cdot|j) \parallel \mathcal{Q}_{\theta}(\cdot|j)\big), \tag{8}$$

and accumulate all these divergence values as our second loss component,

$$\mathcal{L}_{\text{Prop.}}(\mathcal{P}_{\mathrm{A}\neq\mathrm{B}}, \mathcal{Q}_{\theta}) = \frac{1}{T}\sum_{j=1}^{T} \mathcal{L}^{j}_{\mathrm{A}\neq\mathrm{B}}. \tag{9}$$

A visualization of the construction of Equation (8) is depicted in Figure 2.

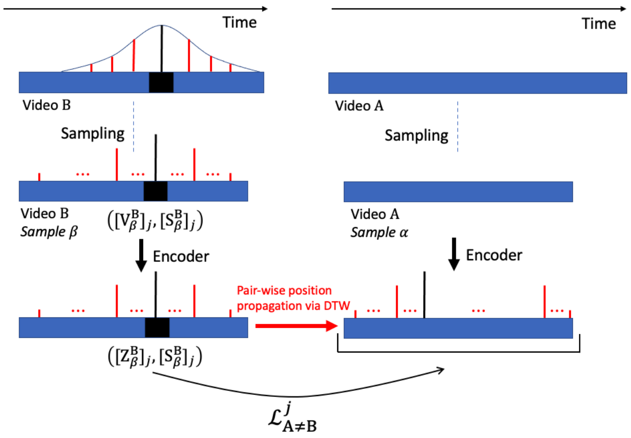
Time
Time
Video B
Video A
Sampling
Sampling
Video B
Sample β
$([V^{B}_{\beta}]_j, [S^{B}_{\beta}]_j)$
Video A
Sample α
Encoder
Encoder
Pair-wise position propagation via DTW
$([Z^{B}_{\beta}]_j, [S^{B}_{\beta}]_j)$
$\mathcal{L}^{j}_{\mathrm{A}\neq\mathrm{B}}$

Figure 2. Illustration of pair-wise position propagation, showing the loss computation for the jth frame in the β sample of video B. We first calculate the Gaussian distribution of timestamp distances, centered around this frame. Afterwards, we propagate this distribution from video B to video A and minimize the KL-divergence between that distribution and the embedding similarity one.

3.4. Better Alignment via SoftDTW

The pair-wise position propagation as described above is effective only when the features used are representative enough. To improve this alignment during training, we suggest to also optimize the smooth DTW (SoftDTW) distance [11]. This is defined as follows,

$$\begin{aligned} r^{\gamma}(i,j) \;&=\; D(i,j) + \\ &\min{}^{\gamma}\{r^{\gamma}(i-1,j), r^{\gamma}(i,j-1), r^{\gamma}(i-1,j-1)\}, \end{aligned} \tag{10}$$

The term $r^{\gamma}(i,j)$ holds the soft-DTW distance up to frames i and j in the two videos, respectively. The expression $\min^{\gamma}$ is a smooth (and therefore differentiable) version of the min function, defined,

$$\min{}^{\gamma}\{a_1, a_2, \ldots, a_n\} = -\gamma \log \sum_{i=1}^{n} e^{\frac{-a_i}{\gamma}}. \tag{11}$$

Note that $\min^{\gamma}$ converges to the discrete $\min$ operator as γ approaches zero. Therefore, when γ is near zero, the smooth DTW distance produces results that are similar to those of the discrete DTW.

Armed with the SoftDTW, given a single pair of videos with their randomly chosen indices, $(Z^{\mathrm{A}}_{\alpha}, S^{\mathrm{A}}_{\alpha}, Z^{\mathrm{B}}_{\beta}, S^{\mathrm{B}}_{\beta})$, we optimize the following objective function:

$$\begin{aligned} \mathcal{L}_{\textbf{LRProp}}(Z^{\mathrm{A}}_{\alpha}, S^{\mathrm{A}}_{\alpha}, Z^{\mathrm{B}}_{\beta}, S^{\mathrm{B}}_{\beta}) \;&=\; \\ \delta_{\mathrm{AB}} \cdot \mathcal{L}_{\text{Same}} \;&+\; \\ (1-\delta_{\mathrm{AB}}) \cdot (\lambda_1 \cdot \mathcal{L}_{\text{Prop.}} \;&+\; \lambda_2 \cdot \mathcal{L}_{\text{Sdtw}}). \end{aligned} \tag{12}$$

Here, δ_{AB} is the Kronecker delta, which is 1 if $\mathrm{A} = \mathrm{B}$ and 0 otherwise. $\mathcal{L}_{\text{Sdtw}} \equiv \mathcal{L}_{\text{Sdtw}}(Z^{\mathrm{A}}_{\alpha}, Z^{\mathrm{B}}_{\beta})$ is the smooth DTW distance between the two videos, which is computed using the embedding vectors Z^{A}_{α} and Z^{B}_{β} via Equation (10). λ_1 and λ_2 are hyper-parameters that control the relative importance of the different terms in this objective function. For further implementation details and hyper-parameters, see Section B in the supplementary material.

4. Empirical Study

In this section, we evaluate **LRProp** on two datasets using various evaluation metrics.

4.1. Datasets

The **PennAction dataset** [49] includes videos of humans performing various sports activities. We use 13 of these actions, following TCC [14]. The dataset includes 1140 videos for training and 966 for testing, with each action set containing 40-134 train videos and 42-116 test videos. For evaluation, we obtain per-frame labels from LAV [20]. These videos contain 18 to 663 frames.

Pouring dataset [36]. This dataset contain videos showing the process of a hand pouring a liquid from one object to another. The phase labels, based on the TCC [14], consist of five classes. Following TCC, we use 70 videos for training and 14 for testing. These videos contain 186 to 797 frames.

4.2. Evaluation Metrics

We use the following metrics to evaluate the frame-wise trained representations:

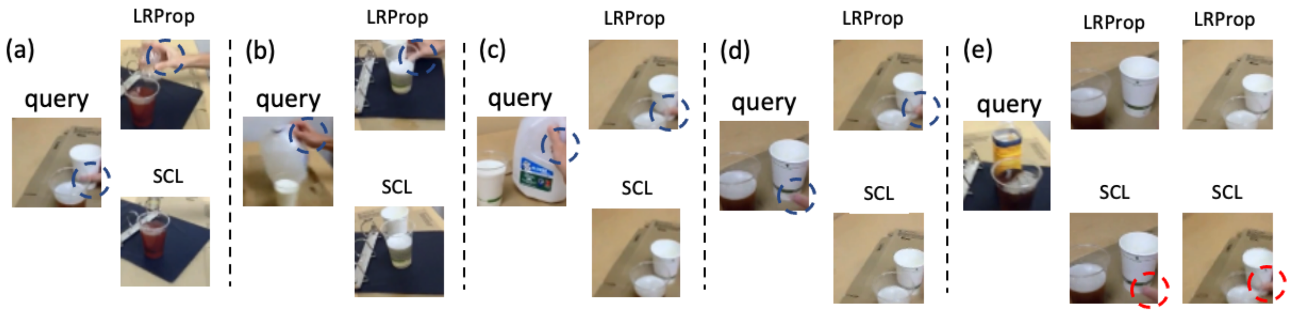

Figure 3. An example of aligned frames (**Pouring** dataset) using the DTW algorithm. In (a), (b), (c), and (d), **LRProp** is better at capturing the specific action of placing the bottle/cup, as indicated by the blue circles; in contrast, in SCL, the human hand is missing in the aligned frame. In (e), the query frame shows the end of the pouring action. Two aligned video results are shown; while **LRProp** provides a successful match, SCL features are inferior - observe the human hand still present in the frames, as indicated by the red circles.

Table 1. Comparison with state-of-the-art methods on **Pouring** using various evaluation metrics: Phase Classification@% (Classification@%), Phase Progression (Progress), Kendall's Tau (τ), Average Precision@K (AP@K), DTW Accuracy (DTW A). Best method is in **bold**, second best in underlined. Our proposed technique (**LRProp** - highlighted in gray) dominates all other methods.

Method	τ	*Progress*	*AP@* *K=5*	*K=10*	*K=15*	*Classification@* *10*	*25*	*50*	*100*	*DTW A*
SCL	99.2	93.5	90.04†	89.69†	88.92†	85.78†	87.14†	89.45†	93.73	84.68†
SAL	79.61	77.28	84.05	83.77	83.79	87.63	-	87.58	88.81	-
TCN	85.12	80.44	83.56	83.31	83.01	89.67	-	87.32	89.53	-
TCC	86.36	83.73	87.16	86.68	86.54	90.65	-	91.11	91.53	-
LAV	85.61	80.54	89.13	89.13	89.22	91.61	-	92.82	92.84	-
VAVA	87.55	83.61	-	-	-	91.65	-	91.79	92.84	-
LRProp	**99.46**	**94.09**	**92.41**	**90.33**	**90.86**	**92.7**	**93.88**	**94.44**	**94.36**	**90.22**

Phase Classification Accuracy [14] is a metric that measures the per-frame accuracy of Phase Classification. To calculate this metric, we first extract features from the frames in the training and test data. Then, we train a support vector machine (SVM) [22] classifier on the phase labels for the training data, using the extracted features as input. The classifier is then used to predict the phase labels for each frame in the test data. The Phase Classification Accuracy is calculated as the proportion of correctly predicted labels, and is reported for different percentages of the SVM training set, in order to evaluate the representativeness of the learned features.

Phase Progression [14] evaluates the accuracy of the embeddings in representing the advancement of a process or action. To compute this metric, we establish a rough estimate of the progress within a phase by calculating the difference in timestamps between a specific frame and key events, and then normalizing this difference by the total number of frames in the video. This measure is used as the target for a linear regression model, which is trained on the frame-wise embeddings. The Phase Progression metric is then calculated as the average R-squared measure (coefficient of determination) of the regression model on the test data. This metric captures how well the embeddings capture the relative progression of the phase within a video.

Average Precision@K [20] is a metric used to evaluate the accuracy of fine-grained frame retrieval, where K is the number of retrieved frames. To calculate this metric, we identify the K nearest frames (using KNN) to a given query frame, based on the frame-wise embeddings, and calculate the Average Precision, i.e., the average proportion of retrieved frames that have the same phase label as the query one. This metric is calculated for different values of K, to assess the performance of the frame-wise embeddings at different retrieval sizes. No extra training or fine-tuning required for this metric.

Kendall's Tau [14] is a correlation coefficient that evaluates the temporal alignment of two sequences. Unlike the other metrics discussed above, Kendall's Tau does not require additional labels for evaluation. To calculate this metric, we first sample pairs of frames (u_i, u_j) from the first video, which has n frames, and retrieve the corresponding nearest neighbors (in the feature space), (v_p, v_q), in the second video. This quadruplet of frame indices (i, j, p, q)

is considered *concordant* if $i < j$ and $p < q$ or $i > j$ and $p > q$. Otherwise, it is considered *discordant*. Kendall's Tau is then calculated as the ratio of concordant pairs to the total number of pairs, which captures how well the two sequences are aligned in time. More specifically, it is defined over all pairs of frames in the first video as: $\tau =$ (no. of concordant pairs - no. of discordant pairs)$/\binom{n}{2}$. Kendall's Tau is a measure of the alignment between two sequences in time, where a value of 1 indicates perfect alignment and a value of -1 indicates that the sequences are aligned in the reverse order. One limitation of this metric is that it assumes that there are no repetitive frames in a video. We average this metric across all video pairs in the validation set.

DTW Accuracy is the ultimate metric that measures the ability of the frame-wise embeddings to capture both the phase labels and the Phase Progression of actions or processes in videos. To calculate this metric, we first apply the dynamic time warping (DTW) algorithm to a pair of videos, using the frame-wise embeddings as input. This produces a sequence of connections between corresponding frames in the two videos. The DTW Accuracy is then calculated as the proportion of connections that connect frames with the same phase label. By definition, DTW does not allow for discordant indices, and it's accuracy metric is sensitive to both the ability of the embeddings to predict the phase labels and the ability to capture the Phase Progression. Overall, DTW Accuracy is a useful metric for evaluating the effectiveness of frame-wise embeddings at capturing the temporal structure of actions or processes in videos. We evaluate this metric on all pairs of videos in the validation set, and take the average as the final result.

Following the work in [7, 14, 20, 20], we use the first four metrics above to evaluate the effectiveness of frame-wise embeddings at capturing the temporal structure of actions or processes in videos. In addition, we propose to use the last metric (DTW Accuracy) in order to evaluate the suitability of the learned features for video alignment.

4.3. Comparison to State-of-the-Art

Pouring Dataset. In Table 1 we compare our method with state-of-the-art methods on the task of Pouring. The best method for each metric is shown in bold, and the second best is underlined. Our proposed method outperforms all previous work on this dataset, with SCL performing the second best overall. As not all of our proposed metrics were reported in their original paper, we reproduced their results using their Github repository[1] for a fair comparison with our results – these are marked by a †. Our method demonstrates superiority over all other methods in all tasks. In particular, with our approach we achieve a Phase Classification Accuracy of 93.88% by using only 25% of the available labels, surpassing all other methods, even if they use 100% of the labels. Additionally, our method excels at identifying frames with similar semantics from other videos, as demonstrated by an improvement of almost 2.5% in the Average Precision@K (AP@) column. We also see significant gains in Kendall's tau and Phase progression metrics compared to SCL, which has already shown a phenomenal improvement of more than 10% over all previous methods. Finally, the DTW Accuracy metric, which measures both Phase Classification and Phase Progression, shows a striking improvement of almost 6% over the state-of-the-art, indicating that our proposed approach is highly effective for video alignment. This is also supported by Figure 3, which demonstrates the superior performance of **LRProp** compared to SCL in the alignment task: when given a query frame, **LRProp** is able to capture fine-grained actions more effectively. For further demonstration of **LRProp** on video alignment, see Figure 1, and the supplementary material.

PennAction Dataset. As shown in Table 2, our proposed method demonstrates superior performance in comparison to all other state-of-the-art approaches on the PennAction dataset. Utilizing a weakly-supervised approach specialized for alignment, we follow the approach of [14] and train a separate model for each of the 13 action classes in this dataset. The results shown in the table are the average across all 13 actions; for a more detailed breakdown of results see Section A, Table 4 in the supplementary material.

Our method consistently outperforms all prior work, as highlighted in bold, with SCL performing the second best overall. It is important to note, however, that SCL trained a single model for all 13 action classes (the authors did not report results for each class individually), whereas our method utilizes individualized models for each class. Furthermore, our method achieves an accuracy of 93.17% using only 75% of available labels, surpassing all baselines in Phase Classification (even if they use all the labels). Additionally, we achieve more than a 1% improvement in the Phase Progression metric. This suggests that our method produces highly reliable features for video alignment and fine-grained retrieval, as supported by the small but consistent improvement in Kendall's tau and Average Precision@K metrics. As we did not re-train SCL on this dataset, we do not report the DTW Accuracy in the table. **LRProp** achieves an impressive average DTW Accuracy of 90.15%.

[1] https://github.com/minghchen/CARL_code.

Table 2. Comparison with state-of-the-art methods on **PennAction** using various evaluation metrics. Best method is in **bold** and second best is underlined. **LRProp** (highlighted in gray) outperforms all previous methods.

Method	τ	*Progress*	*AP@*			*Classification@*			
			K=5	*K=10*	*K=15*	*10*	*50*	*75*	*100*
SCL	98.5	91.8	92.28	92.1	91.82	-	-	-	93.07
SAL	76.12	69.6	-	-	-	74.87	78.26	-	79.96
TCN	81.2	72.17	77.84	77.51	77.28	81.99	83.67	-	84.04
TCC	81.35	73.53	76.74	76.27	75.88	81.26	83.35	-	84.45
LAV	80.5	66.13	79.13	79.98	78.9	83.56	83.95	-	84.25
VAVA	80.53	70.91	-	-	-	83.89	84.23	-	84.48
LRProp	**99.09**	**93.03**	**92.46**	**92.2**	**92.03**	**91.9**	**92.96**	**93.17**	**93.25**

4.4. Ablation Study

SoftDTW and pair-wise position propagation: Is it a winning combination? We now turn to address this question by assessing the performance of these two loss terms, $\mathcal{L}_{\text{Prop.}}$ and $\mathcal{L}_{\text{Sdtw}}$, and their contribution on the pouring dataset. We measure the Phase Classification Accuracy for a more diverse set of label portions, and similarly, we measure the Average Precision accuracy for a larger range of K values. We also measure the Phase Progression, Kendall's Tau and DTW Accuracy as in previous sections. The results are depicted in Table 3.

As can be seen, our design choices consistently improve across all evaluation metrics, with the exception of Kendall's Tau, which is already close to saturation. We observe that the first and second rows of Table 3 have similar results in the Kendall's Tau, Phase Progression, and Average Precision@K metrics. This suggests that the combination of $\mathcal{L}_{\text{Prop.}}$ and $\mathcal{L}_{\text{Sdtw}}$ is responsible for the improvement, rather than either one alone. Additionally, **LRProp** shows an average improvement of around 5% in the Phase Classification Accuracy metric for any choice of label percentage. Also, the striking improvement in the DTW Accuracy metric when using both $\mathcal{L}_{\text{Prop.}}$ and $\mathcal{L}_{\text{Sdtw}}$ is particularly noteworthy, as it suggests that the combined loss function is able to produce features that are highly effective for video alignment.

5. Conclusions

In this paper we introduce **LRProp**, a powerful method for extracting frame-level features that are particularly effective for aligning videos. Our approach involves using the dynamic time-warping (DTW) algorithm for establishing a prior distribution between frames from different videos based on their alignment path. In multiple experiments and commonly used metrics, we demonstrate that our method significantly outperforms various baselines in learning feature representations. We also evaluate DTW video alignment performance using the learned features and achieve a striking improvement of more than 5% compared to the previous state-of-the-art.

Limitations and future directions. A potential area for future research is to adapt our method for videos that may contain outlier frames, by identifying and extracting them from the learning/inference processes. This may enable more robust features and better overall alignment between videos. Another challenging task refers to much longer videos, for which current solutions may face a severe memory and computational barriers. To conclude, our results demonstrate that the proposed approach is highly effective at learning better frame-wise feature representations for videos. The use of *pair-wise position propagation*, combined with the SoftDTW, to relate frames between different videos is a particularly promising direction, as we demonstrate its potential to significantly improve the task of video alignment.

Table 3. An ablation study of $\mathcal{L}_{\text{Prop.}}$ and $\mathcal{L}_{\text{Sdtw}}$ using various evaluation metrics. Best result in each column is in **bold**. Our proposed technique is highlighted in gray.

$\mathcal{L}_{Same}$	$\mathcal{L}_{Prop.}$	$\mathcal{L}_{\text{Sdtw}}$	τ	*Progress*	*AP@*						*Classification@*						*DTW A*
					K=1	*K=5*	*K=10*	*K=15*	*K=20*	*K=25*	*5*	*10*	*25*	*50*	*75*	*100*	
✓	✓		98.97	92.58	92.74	90.4	88.68	87.49	86.64	85.94	81.99	83.99	85.08	87	86.62	86.42	81.38
✓		✓	**99.52**	92.25	93.45	90.38	89.02	88.26	87.55	86.69	86.66	87.66	88.61	89.75	89.69	89.69	85
✓	✓	✓	99.46	**94.09**	**94.09**	**92.41**	**91.66**	**90.86**	**90.45**	**90.07**	**91.8**	**92.7**	**93.88**	**94.44**	**94.46**	**94.36**	**90.22**

References

[1] Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lučić, and Cordelia Schmid. Vivit: A video vision transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6836–6846, 2021. 2, 3

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

[3] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9650–9660, 2021. 2

[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. 2

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

[6] Zheng Chang, Xinfeng Zhang, Shanshe Wang, Siwei Ma, and Wen Gao. Strpm: A spatiotemporal residual predictive model for high-resolution video prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13946–13955, 2022. 1

[7] Minghao Chen, Fangyun Wei, Chong Li, and Deng Cai. Frame-wise action representations for long videos via sequence contrastive learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13801–13810, 2022. 2, 3, 4, 7, 11

[8] 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. PMLR, 2020. 2, 11

[9] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey E Hinton. Big self-supervised models are strong semi-supervised learners. *Advances in neural information processing systems*, 33:22243–22255, 2020. 2

[10] Ting Chen, Xiaohua Zhai, Marvin Ritter, Mario Lucic, and Neil Houlsby. Self-supervised gans via auxiliary rotation loss. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12154–12163, 2019. 2

[11] Marco Cuturi and Mathieu Blondel. Soft-dtw: a differentiable loss function for time-series. In *International conference on machine learning*, pages 894–903. PMLR, 2017. 2, 3, 5

[12] Carl Doersch, Abhinav Gupta, and Alexei A Efros. Unsupervised visual representation learning by context prediction. In *Proceedings of the IEEE international conference on computer vision*, pages 1422–1430, 2015. 2

[13] Sixun Dong, Huazhang Hu, Dongze Lian, Weixin Luo, Yicheng Qian, and Shenghua Gao. Weakly supervised video representation learning with unaligned text for sequential videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2437–2447, 2023. 2

[14] Debidatta Dwibedi, Yusuf Aytar, Jonathan Tompson, Pierre Sermanet, and Andrew Zisserman. Temporal cycle-consistency learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1801–1810, 2019. 2, 3, 5, 6, 7

[15] Yazan Abu Farha and Jurgen Gall. Ms-tcn: Multi-stage temporal convolutional network for action segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3575–3584, 2019. 2

[16] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in neural information processing systems*, 33:21271–21284, 2020. 11

[17] Isma Hadji, Konstantinos G Derpanis, and Allan D Jepson. Representation learning via global temporal alignment and cycle-consistency. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11068–11077, 2021. 3

[18] Yunfei Han and Shan Tan. Twinlstm: Two-channel lstm network for online action detection. In *2022 26th International Conference on Pattern Recognition (ICPR)*, pages 3310–3317. IEEE, 2022. 1

[19] Yanbin Hao, Shuo Wang, Pei Cao, Xinjian Gao, Tong Xu, Jinmeng Wu, and Xiangnan He. Attention in attention: Modeling context correlation for efficient video classification. *IEEE Transactions on Circuits and Systems for Video Technology*, 2022. 1

[20] Sanjay Haresh, Sateesh Kumar, Huseyin Coskun, Shahram N Syed, Andrey Konin, Zeeshan Zia, and Quoc-Huy Tran. Learning by aligning videos in time. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5548–5558, 2021. 2, 3, 5, 6, 7

[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 11

[22] Marti A. Hearst, Susan T Dumais, Edgar Osuna, John Platt, and Bernhard Scholkopf. Support vector machines. *IEEE Intelligent Systems and their applications*, 13(4):18–28, 1998. 6

[23] Zeeshan Khan, C.V. Jawahar, and Makarand Tapaswi. Grounded video situation recognition. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022. 1

[24] Dahun Kim, Donghyeon Cho, and In So Kweon. Self-supervised video representation learning with space-time cubic puzzles. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 8545–8552, 2019. 1

[25] Jie Lei, Tamara L Berg, and Mohit Bansal. Detecting moments and highlights in videos via natural language

queries. *Advances in Neural Information Processing Systems*, 34:11846–11858, 2021. 1

[26] Mengze Li, Han Wang, Wenqiao Zhang, Jiaxu Miao, Zhou Zhao, Shengyu Zhang, Wei Ji, and Fei Wu. Winner: Weakly-supervised hierarchical decomposition and alignment for spatio-temporal video grounding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23090–23099, 2023. 2

[27] Weizhe Liu, Bugra Tekin, Huseyin Coskun, Vibhav Vineet, Pascal Fua, and Marc Pollefeys. Learning to align sequential actions in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2181–2191, 2022. 3

[28] Fuchen Long, Ting Yao, Zhaofan Qiu, Xinmei Tian, Jiebo Luo, and Tao Mei. Bi-calibration networks for weakly-supervised video representation learning. *International Journal of Computer Vision*, 131(7):1704–1721, 2023. 2

[29] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016. 11

[30] Ishan Misra, C Lawrence Zitnick, and Martial Hebert. Shuffle and learn: unsupervised learning using temporal order verification. In *European conference on computer vision*, pages 527–544. Springer, 2016. 2, 3

[31] Daniel Neimark, Omri Bar, Maya Zohar, and Dotan Asselmann. Video transformer network. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3163–3172, 2021. 2, 3

[32] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *European conference on computer vision*, pages 69–84. Springer, 2016. 2

[33] Boxiao Pan, Haoye Cai, De-An Huang, Kuan-Hui Lee, Adrien Gaidon, Ehsan Adeli, and Juan Carlos Niebles. Spatio-temporal graph for video captioning with knowledge distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10870–10879, 2020. 1

[34] Senthil Purushwalkam, Tian Ye, Saurabh Gupta, and Abhinav Gupta. Aligning videos in space and time. In *European Conference on Computer Vision*, pages 262–278. Springer, 2020. 2

[35] Mingyang Qiao and Tiantian Yuan. Action recognition based on video spatio-temporal transformer. In *2022 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA)*, pages 477–481. IEEE, 2022. 1

[36] Pierre Sermanet, Corey Lynch, Yevgen Chebotar, Jasmine Hsu, Eric Jang, Stefan Schaal, Sergey Levine, and Google Brain. Time-contrastive networks: Self-supervised learning from video. In *2018 IEEE international conference on robotics and automation (ICRA)*, pages 1134–1141. IEEE, 2018. 2, 3, 5

[37] Nitish Srivastava, Elman Mansimov, and Ruslan Salakhudinov. Unsupervised learning of video representations using lstms. In *International conference on machine learning*, pages 843–852. PMLR, 2015. 3

[38] 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. 2

[39] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 2, 3, 4, 11

[40] Carl Vondrick, Hamed Pirsiavash, and Antonio Torralba. Generating videos with scene dynamics. *Advances in neural information processing systems*, 29, 2016. 3

[41] Longguang Wang, Yulan Guo, Li Liu, Zaiping Lin, Xinpu Deng, and Wei An. Deep video super-resolution using hr optical flow estimation. *IEEE Transactions on Image Processing*, 29:4323–4336, 2020. 1

[42] Xiaolong Wang, Allan Jabri, and Alexei A Efros. Learning correspondence from the cycle-consistency of time. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2566–2576, 2019. 1

[43] Xiaofang Wang, Xuehan Xiong, Maxim Neumann, AJ Piergiovanni, Michael S Ryoo, Anelia Angelova, Kris M Kitani, and Wei Hua. Attentionnas: Spatiotemporal attention cell search for video classification. In *European Conference on Computer Vision*, pages 449–465. Springer, 2020. 1

[44] Bohan Wu, Suraj Nair, Roberto Martin-Martin, Li Fei-Fei, and Chelsea Finn. Greedy hierarchical variational autoencoders for large-scale video prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2318–2328, 2021. 1

[45] Ravindra Yadav, Ashish Sardana, Vinay P Namboodiri, and Rajesh M Hegde. Learning to predict speech in silent videos via audiovisual analogy. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8042–8046. IEEE, 2022. 3

[46] Runhao Zeng, Wenbing Huang, Mingkui Tan, Yu Rong, Peilin Zhao, Junzhou Huang, and Chuang Gan. Graph convolutional networks for temporal action localization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7094–7103, 2019. 1

[47] Jiecheng Zhai, Xunxiang Yao, Guangyuan Dong, Qun Jiang, and Yunfeng Zhang. 3d dual-stream convolutional neural networks with simple recurrent unit network: A new framework for action recognition. In *2022 4th International Conference on Communications, Information System and Computer Engineering (CISCE)*, pages 509–515. IEEE, 2022. 1

[48] Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *European conference on computer vision*, pages 649–666. Springer, 2016. 2

[49] Weiyu Zhang, Menglong Zhu, and Konstantinos G Derpanis. From actemes to action: A strongly-supervised representation for detailed action understanding. In *Proceedings of the IEEE international conference on computer vision*, pages 2248–2255, 2013. 2, 5