Rethinking Zero-shot Action Recognition: Learning from Latent Atomic Actions

Yijun Qian, Lijun Yu, Wenhe Liu, and Alexander G. Hauptmann

Carnegie Mellon University,Pittsburgh PA 15213,USA
yijunqia@andrew.cmu.edu,lijun@cmu.edu,wenhel@andrew.cmu.edu,Alex@cs.cmu.edu

Abstract. To avoid time-consuming annotating and retraining cycle in applying supervised action recognition models, Zero-Shot Action Recognition (ZSAR) has become a thriving direction. ZSAR requires models to recognize actions that never appear in training set through bridging visual features and semantic representations. However, due to the complexity of actions, it remains challenging to transfer knowledge learned from source to target action domains. Previous ZSAR methods mainly focus on mitigating representation variance between source and target actions through integrating or applying new action-level features. However, the action-level features are coarse-grained and make the learned one-to-one bridge fragile to similar target actions. Meanwhile, integration or application of features usually requires extra computation or annotation. These methods didn't notice that two actions with different names may still share the same atomic action components. It enables humans to quickly understand an unseen action given bunch of atomic actions learned from seen actions. Inspired by this, we propose Jigsaw Network (JigsawNet) which recognizes complex actions through unsupervisedly decomposing them into combinations of atomic actions and bridging group to group relationships between visual features and semantic representations. To enhance the robustness of learned group-to-group bridge, we propose Group Excitation (GE) module to model intra-sample knowledge and Consistency Loss to enforce the model learn from inter-sample knowledge. Our JigsawNet achieves state-of-the-art performance on three benchmarks and surpasses previous works with noticeable margins.

Keywords: Computer Vision, Action Recognition, Zero-shot Learning

1 Introduction

Supervised action recognition has been a heated computer vision task and makes continuous progress thanks to the development of spatio-temporal modeling[32, 39, 28] and release of large datasets[18, 36]. However, while the performance of models has been boosted, the models also become deeper and deeper and require more and more annotated data for training. Meanwhile, the target actions vary in different scenes and may change or increase in real-world applications. Thus, the time-consuming cycle of gathering data, annotating, and retraining becomes

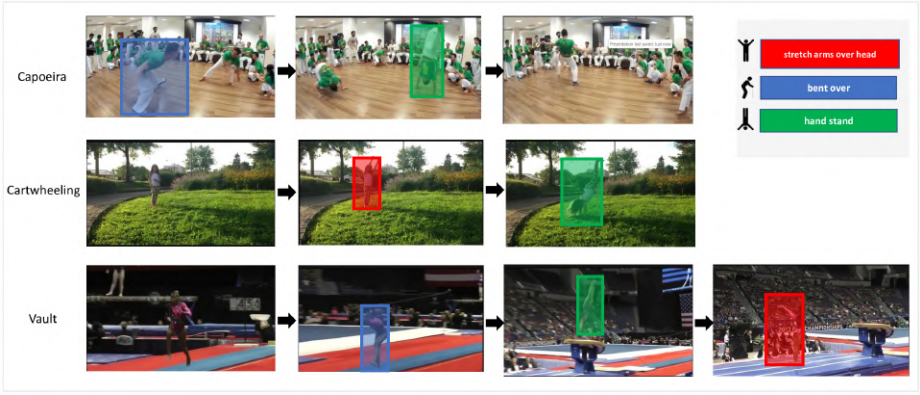

Fig. 1. The majority of actions in daily life are complicated and can be regarded as combinations of atomic actions. Complex actions with different name may still share the same atomic actions. (*e.g.* "hand stand" exists commonly in complex actions such as capoeira, cartwheeling, and vault.) To better understand the ZSAR procedure, we could regard target actions as "assembled puzzles" picture drawn on puzzle boxes, atomic actions are "puzzle pieces" and the videos are stacks of puzzle pieces. JigsawNet is determining which boxes the stacks belong to through matching the puzzle pieces in stack with the ones drawn on boxes.

inevitable to use these supervised action recognition models. To alleviate such burden, Zero-Short Action Recognition (ZSAR) has become a vigorous direction which enables models the capability of recognizing actions that never appear in the training set.

These years have witnessed many successful explorations[29, 46, 4, 8] in ZSAR. The task requires models to bridge semantic representations of actions and visual features extracted from videos. However, it remains challenging due to the complexity of videos and semantic variance between source and target actions. Previous methods mainly focus on mitigating the representation variance between source and target actions through integrating or applying new action-level features. The first series of works[29, 46] use manually designed attributes to represent video features and action semantics. Another series of works[22, 17] introduce the existence of objects as attributes. Features are generated through embedding verbal objects detected in action descriptions and visual objects recognized in video clips. The last series of works [44, 4] use embedding of label name as action semantics and end-to-end train a spatio-temporal modeling network to extract visual features. The most recent work[8] uses description instead of label name for semantic extraction. It also integrates both spatio-temporal features and semantics of detected verbal objects as visual representations. To the best of our knowledge, all the previous methods regard the videos and actions as single entities and enforce the model to build a one-to-one bridge between visual features and semantic representations. They project the visual features extracted from videos and semantic representations extracted from actions to

a shared sphere. The models are then optimized through minimizing distance between visual features and corresponding action semantics.

Although both visual and semantic variance between source and target actions will be mitigated to a extent, the features are still coarse-grained and may not as distinct on target actions domain as learned among source actions domain. It makes the the one-to-one bridge fragile and make the action classifier difficult to distinguish similar target actions. What's more, integration or application of new attributes usually requires extra computation or annotation which may limit the usage in real world scenes. In this work, we take the inspiration from how humans quickly understand a brand new action through factorizing it into combinations of atomic actions.

We use puzzles as an example to introduce the difference between our JigsawNet and previous works. As is shown in Figure 1, complex actions like "capoeira", "cartwheeling", and "vault" can be regarded as "assembled puzzles" drawn on boxes, atomic actions like "bent over","hand stand", and "stretch arms over head" can be regarded as "puzzle pieces", and videos can be regarded as stacks of these puzzle pieces. Then, a ZSAR task is to determine which boxes the stacks should be placed in. Previous methods[8] are trying to make the decision based on stack-level (action-level) features. For example, the number of edge puzzle pieces in stacks (number of people in videos) is a distinct stack-level feature between capoeira and cartwheeling. However, it's not as distinct between cartwheeling and vault. JigsawNet, instead, looks deeper and makes decision through matching the pieces in stacks with the ones drawn on boxes. It notices that different assembled puzzles still share the same pieces. Such piece-level features are easier to recognize and stable among different assembled puzzles. In other words, these atomic actions can be regarded as joint latent features shared by source and target actions, which are fine-grained and won't change drastically.

Given videos and descriptions of target actions crawled from wiki, the videos are split into groups of segments and the descriptions are separated into groups of verb phrases through dependency parsing. JigsawNet then recognizes unseen actions through unsupervisedly bridging a group-to-group relationship among video segments and semantics of atomic actions. JigsawNet can learn the relationship only based on target action labels assigned to the entire videos and require no extra annotation of atomic action label assigned to each segment. Meanwhile, to better model the latent features of atomic actions, we propose the Consistency Loss to learn from inter-sample knowledge and Group Excitation (GE) module to adaptively aggregate intra-sample representations. Consistency Loss enforces the model to extract similar features from segments which are predicted as the same atomic action but split from different videos. It also enforces the model to enlarge distance of similar features which are extracted from segments that are predicted as different atomic actions. For computation efficiency, a memory cache is implemented to remember past visual features grouped by atomic action labels. Visual features extracted from single segment has limited temporal receptive field. Thus, GE module will adaptively aggregate contextual spatio-temporal and object features and enables the model a better capability of

understanding latent atomic actions. To the best of our knowledge, the proposed algorithm is the first to explore atomic actions as joint latent features and build group-to-group bridge between visual features and semantic representations. It is also the first exploration in making full usage of both inter and intra sample knowledge in ZSAR.

In a nutshell, the contributions of our paper are summarized as follows:

1. We propose a novel view to implement atomic actions as joint latent features which is fine-grained and stable between source and target actions for ZSAR.
2. We propose JigsawNet which recognizes unseen actions through unsupervisedly decomposing them into combinations of atomic actions and then bridging a group-to-group relationship between visual features and semantic representations.
3. We propose the Consistency Loss and the Group Excitation (GE) module to make full usage of both inter and intra sample knowledge for ZSAR.
4. Our method achieves state-of-the-art (SOTA) performance on three ZSAR benchmarks (*i.e.* KineticsZSAR, HMDB51, and UCF101) which demonstrates the superiority of our work.

2 Related Work

Supervised Action Recognition Action Recognition has been a heated task in computer vision given its wide application in real world scenes. Given a target-centered action clip, supervised action recognition models can recognize the actions that it has seen in the training set. There are three major series of solutions.

The earliest works operate spatial-convolution independently over the temporal dimension and resort to temporal motion information like optical flow or RGB diff of adjacent frames for temporal modeling [23, 35, 41, 49]. However, the extraction of motion features is time consuming and limits the usage in real world applications.

Later works [21, 20, 11, 12] resort to 3D CNNs which can model spatial and temporal information simultaneously. They have achieved state-of-the-art (SOTA) results on many data sets, especially after the release of large video action data sets like Kinetics[24] and Activity Net[5]. For example, Du proposes a simple, yet effective approach for spatio-temporal feature learning using deep 3D CNN[38]. Carreira implements a two-stream inflated 3D convolution network [7]. However, 3D CNNs contain much more parameters and will easily overfit and still time consuming.

Given the drawbacks of traditional 2D CNN and 3D CNN mentioned above, most recent works[39, 47, 13, 34, 13] focus on factorizing the spatial and temporal modeling operations. For example, Du proposed R(2+1)D which factorizes the 3D convolutional filters into 2D spatial convolution kernel and 1D temporal convolution kernel[39].

Zero-shot Action Recognition The earliest works take manual-defined attributes[29, 46] to represent the action. For example, Gan *et al.* propose multi-source domain generalization method for attribute detection. However, the attributes of actions are numerous and harder to define compared with static images. Later works[16, 17, 22] resort to objects as attributes. For example, Jain *et al.* [22] detect objects in videos then project the object features and action labels to a shared sphere and calculate the similarity. Gao *et al.* propose a graph networks based model to learn the relationship between action and objects and match them according to action prototypes. These works are pretty efficient, however, they ignore the spatio-temporal information inherited in videos. Most recent works use word embeddings of action names[4, 31, 33, 44] or action descriptions[8] to extract semantic representations. For example, Brattoli *et al* [4] propose an end-to-end pipeline which directly projects the extracted spatio-temporal features from videos and semantic representations extracted from word embeddings of action names to a shared sphere. The most similar work to ours is ER [8] which also uses description, objects, and videos as input. However ER is different from ours in that, it still recognizes the videos and actions as single entities and enforces the model to build one-to-one bridge between visual features and semantic representations. The action-level features are coarse-grained and may not as distinct among target action as learned among source actions.

3 Proposed Approach

ZSAR task requires the model to recognize actions that never appears in training set. In the rest subsections, we will present our novel Group Alignment Module, Consistency Loss, and Group Excitation Module. We will then show how to combine these modules together as Jigsaw Network.

3.1 Group Alignment (GA) Module

For each action label y^i, we crawled a short description and split it into group of verb phrases through dependency parsing. Manual correction is applied to remove typos and modify incorrect descriptions. The action y^i is then represented as $d^i = \{w_1, ..., w_k\}$, where each w_j is a verb phrase and represents an atomic action. We implement a spatio-temporal extraction backbone $\mathcal{N}_{vid}$ to extract visual features from i^{th} video v^i. $\mathcal{F}_i = \mathcal{N}_{vid}(v^i)$, where $\mathcal{F}_i = \{f_1^i, ..., f_m^i\} \in \mathbb{R}^{m \times d}$, m represents the number of segments the v^i is split into, and d is the dimension of features extracted from each video segment. A semantic extraction backbone $\mathcal{N}_{text}$ is implemented to extract semantic representations from group of verb phrases $\mathcal{G}_i = \mathcal{N}_{text}(d^i)$, where $\mathcal{G}_i = \{g_1^i, ..., g_k^i\} \in \mathbb{R}^{k \times d}$. Besides extracting spatio-temporal features, we also use object types in videos as a kind of video attribute. An object classification backbone $\mathcal{N}_{obj}$ is implemented to recognize object classes $\mathcal{W}_{obj}^i = \mathcal{N}_{obj}(v^i)$, and $\mathcal{W}_{obj}^i = \{w_{obj}^1, ..., w_{obj}^l\}$. Each w_{obj}^k represents the name of a recognized object. The object names are concatenated and forwarded to $\mathcal{N}_{text}$ to extract verbal features $\mathcal{O}_i = \mathcal{N}_{text}(\mathcal{W}_{obj}^i)$, and $\mathcal{O}_i \in \mathbb{R}^d$.

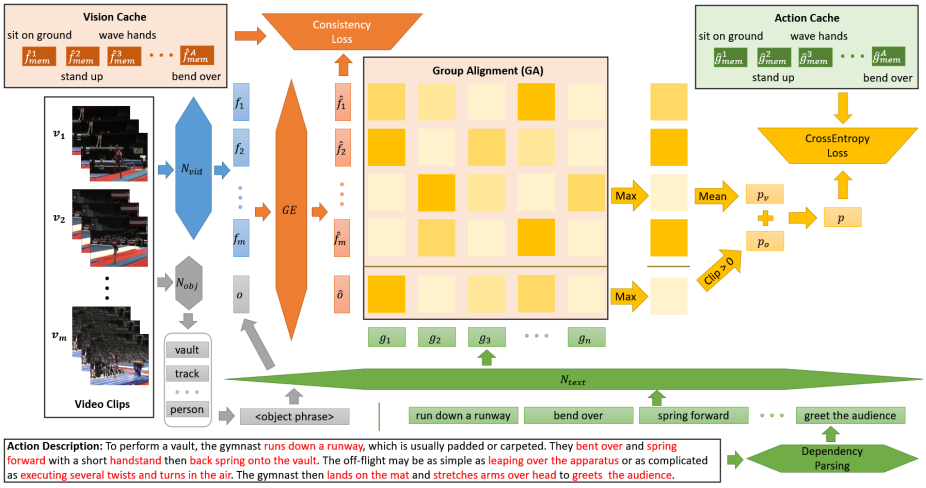

Fig. 2. Architecture of Jigsaw Network. Given videos and short descriptions of target actions, Jigsaw Network will adaptively learn to build group-to-group bridges. N_{vid} is the spatio-temporal extractor backbone, N_{text} is the semantic extractor backbone, and N_{obj} is the object recognition backbone. Green represents verbal features, blue represents visual features, and red represents fused features.

Current spatio-temporal feature f_x^i extracted from each segment has limited temporal receptive field, and the complete atomic action may be split into two segments. To solve this, we propose the Group Excitation (GE) Module $\mathcal{M}$ to adaptively aggregate intra sample features.

$$\hat{\mathcal{F}}_i, \hat{\mathcal{O}}_i = \mathcal{M}(\mathcal{F}_i \oplus \mathcal{O}_i) \tag{1}$$

$$\hat{\mathcal{F}}_i = \{\hat{f}_1^i, ..., \hat{f}_m^i\} \in \mathbb{R}^{m \times d}, \hat{\mathcal{O}}_i \in \mathbb{R}^d \tag{2}$$

where $\oplus$ represents concatenate. Each output $\hat{f}_j^i \in \hat{\mathcal{F}}_i$ will contain spatio-temporal features extracted from contextual segments and object feature $\mathcal{O}_i$ of the whole video sample. The details of $\mathcal{M}$ will be presented in later subsections. We then build the group-to-group bridge between the group of video segments and the group of atomic actions.

$$p_{xy}^{ij} = \hat{f}_x^i \cdot (g_y^j)^T \tag{3}$$

$$p_a^{ij} = \frac{\sum_x (\max_y (p_{xy}^{ij}))}{m} \tag{4}$$

$$p_o^{ij} = \max_y (\hat{\mathcal{O}}_i \cdot (g_y^j)^T) \tag{5}$$

$$p^{ij} = p_a^{ij} + \max(p_o^{ij}, 0) \tag{6}$$

where p_{xy}^{ij} represents the cosine similarity between the visual feature $\hat{f}_x^i$ extracted from the x^{th} video segment of v^i, and semantic representations g_y^j of the y^{th} verb phrase of d^i, p^{ij} is the probability that v^i is recognized as j^{th} target action

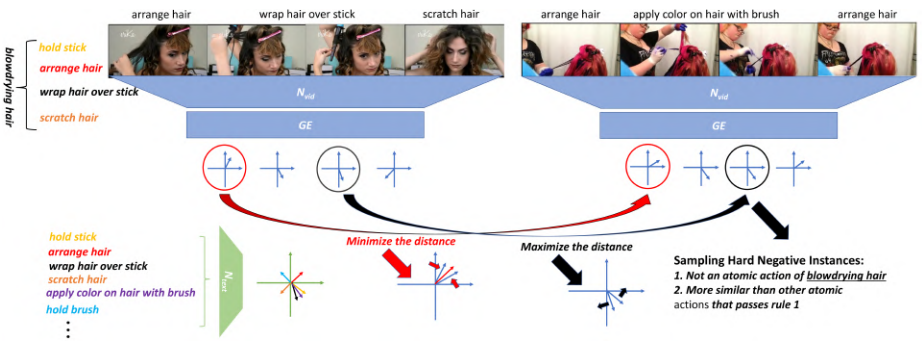

Fig. 3. Illustration of Consistency Loss. Although the first segments of the left and right video are both predicted as "arrange hair", their visual features may approaching the semantic representation in two directions and Consistency Loss will further minimize their distance. The visual features extracted from the second segments of left (seg_l) and right (seg_r) videos are similar. However, "apply color on hair with brush" is not a component of "blowdrying hair". Thus the seg_r is a hard negative sample of seg_l and there distances will be enlarged by Consistency Loss as well

(y^i=j), $\cdot$ denotes vector-matrix multiplication. We implement a cross entropy loss to train the GA module. $P^i = \{p^{i1}, p^{i2}, ..., p^{iB}\} \in \mathbb{R}^B$ is the probability vector of video v^i to B source actions. $Y^i = \{y^i_1, y^i_2, ..., y^i_B\} \in \mathbb{R}^B$ is the one-hot label vector extended from y^i. The cross entropy loss is shown as:

$$L(P^i, Y^i) = -\sum_{j=1}^{B} y^i_j \log \frac{\exp(p^{ij}/\lambda)}{\sum_{k=1}^{B} \exp(p^{ik}/\lambda)} \tag{7}$$

where λ is a temperature parameter. Although P^i is already calculated by both $\hat{\mathcal{F}}_i$ and $\hat{\mathcal{O}}_i$, we intend to enforce these two features similar to the semantic representation of target action as well. $P^i_a = \{p^{i1}_a, p^{i2}_a, ..., p^{iB}_a\} \in \mathbb{R}^B$ and $P^i_o = \{p^{i1}_o, p^{i2}_o, ..., p^{iB}_o\} \in \mathbb{R}^B$ are probability vectors of video v^i to B source actions calculated by $\hat{\mathcal{F}}_i$ and $\hat{\mathcal{O}}_i$ independently. Thus, the matching loss is shown as:

$$L_{act} = \frac{1}{N} \sum_{i=1}^{N} (L(P^i, Y^i) + L(P^i_a, Y^i) + L(P^i_o, Y^i)) \tag{8}$$

During inference stage, only P^i is used as the probability scores of v^i to C target actions.

3.2 Consistency Loss

L_{act} only uses B semantic representations of source actions for training which makes the model easily overfit. To enhance the robustness of group-to-group bridge, we propose Consistency Loss which optimizes the model unsupervisedly with inter sample knowledge. As is shown in Figure 3, although the model doesn't

have the ground truth atomic action label of each video segment, the segments should only contain atomic actions which are components of the aligned source action. Meanwhile, the same atomic action will be a joint latent feature shared by different source actions. Generally, the consistency regularization is two folded. Features extracted from segments of different videos but aligned to the same atomic action should be consistently similar. Segment features aligned to certain atomic action should be consistently different from those aligned to other atomic actions. The loss can be represented as:

$$\mathcal{F}_{mem} = \{\hat{f}^1_{mem}, \hat{f}^2_{mem}, ..., \hat{f}^A_{mem}\} \in \mathbb{R}^{A \times d} \tag{9}$$

$$\hat{j} = \underset{j \in \{1,...,B\}}{\operatorname{argmax}} \ p^{ij} \tag{10}$$

$$s^i_x = \Phi(\underset{y}{\operatorname{argmax}}\, p^{i\hat{j}}_{xy}) \tag{11}$$

$$n^i_x = \underset{m \notin W^{\hat{j}}}{\operatorname{argmax}} \hat{f}^i_x \cdot \hat{f}^m_{mem} \tag{12}$$

$$L_{cons} = \frac{\sum_i \sum_x \max(1 - \hat{f}^i_x \cdot \hat{f}^{s^i_x}_{mem} + \hat{f}^i_x \cdot \hat{f}^{n^i_x}_{mem}, 0)}{N} \tag{13}$$

where $\mathcal{F}_{mem}$ is the memorized visual feature buffer grouped by atomic action types, and A is the number of unique atomic actions of all source actions. $\hat{j}$ is the predicted action id of video v^i. $s^i_x \in \{1, 2, ..., A\}$ represents the predicted atomic action id of the x^{th} segment of v^i. Φ is a wrap function which projects the inner group id y to its global unique id in $\mathcal{F}_{mem}$. $W^{\hat{j}} = \{w^{\hat{j}}_1, w^{\hat{j}}_2, ..., w^{\hat{j}}_k\}$ is the ids of atomic actions belonging to source action $\hat{j}$, and $w^{\hat{j}}_m \in \{1, 2, ..., A\}$. Thus n^i_x represents the atomic action whose memorized feature is mostly similar to $\hat{f}^i_x$ but not a component of source action $\hat{j}$. L_{cons} minimizes the distance between $\hat{f}^i_x$ and $\hat{f}^{s^i_x}_{mem}$ and maximizes the distance between $\hat{f}^i_x$ and $\hat{f}^{n^i_x}_{mem}$. The rule of updating $\mathcal{F}_{mem}$ is:

$$\hat{f}^{s^i_x}_{mem} = \begin{cases} f^i_x & f^i_x \cdot \hat{g}^{s^i_x}_{mem} > \hat{f}^{s^i_x}_{mem} \cdot \hat{g}^{s^i_x}_{mem} \\ \hat{f}^{s^i_x}_{mem} & f^i_x \cdot \hat{g}^{s^i_x}_{mem} \leq \hat{f}^{s^i_x}_{mem} \cdot \hat{g}^{s^i_x}_{mem} \end{cases} \tag{14}$$

where $\mathcal{G}_{mem} = \{\hat{g}^1_{mem}, ..., \hat{g}^A_{mem}\}$ is the memorized verbal feature buffer of atomic actions. Meanwhile, we also implemented the ER Loss[8] L_{er} which resorts to recognized objects as weak supervision. Thus, the full loss function $\mathcal{L}$ is:

$$\mathcal{L} = L_{act} + L_{er} + L_{cons} \tag{15}$$

3.3 Group Excitation (GE) Module

Different from previous methods which extract spatio-temporal features from entire videos, our $\mathcal{N}_{vid}$ only extracts spatio-temporal features from video segments with limited temporal receptive field. Meanwhile, each segment may not cover the complete atomic action. To better understand each atomic action, we

propose the GE module to adaptively aggregate intra sample features. Inspired by recent success of implementing transformer[40] for both computer vision[30, 3] and NLP[10, 14] tasks, we propose GE, a multi-head transformer based aggregation module to fuse intra sample features $\hat{\mathcal{F}}_i$ and $\hat{\mathcal{O}}_i$. It enables the feature $\hat{f}_j^i$ of each video segment to contain contextual spatio-temporal feature of all other inter sample segments. GE is functioned as:

$$\begin{aligned} H =& \text{concat}(\mathcal{F}_i, \mathcal{O}_i) \\ =& \{f_1^i, ..., f_m^i, \mathcal{O}_i\} \in \mathbb{R}^{(m+1)\times d} \end{aligned} \tag{16}$$

$$head_j = \frac{HW_j^Q \cdot (HW_j^K)^T}{\sqrt{d}} HW_j^V \tag{17}$$

$$\begin{aligned} \mathcal{M}(I) =& \text{concat}(head_1, ..., head_h)W^O \\ =& \{\hat{f}_1^i, ..., \hat{f}_m^i, \hat{\mathcal{O}}_i\} \\ =& \{\hat{\mathcal{F}}_i, \hat{\mathcal{O}}_i\} \end{aligned} \tag{18}$$

where $W_j^Q \in \mathbb{R}^{d\times\frac{d}{h}}$,$W_j^K \in \mathbb{R}^{d\times\frac{d}{h}}$,$W_j^V \in \mathbb{R}^{d\times\frac{d}{h}}$, and $W^O \in \mathbb{R}^{d\times d}$.

3.4 Jigsaw Network (JigsawNet)

As is shown in Figure 2, we implement the JigsawNet with modules introduced above. For efficiency, JigsawNet has an action cache to memorize semantic representations extracted from verb phrases grouped by atomic actions. In each training iteration, $\mathcal{N}_{text}$ only extracts semantic representations from verb phrases belonging to labeled source actions and concatenated verbal objects of videos in batch. Before each training or validation epoch starts, the action cache will be initialized through extracting features from all A verb phrases of atomic actions with $\mathcal{N}_{text}$. In each validation iteration, since the $\mathcal{N}_{text}$ won't be updated, so the model will directly pick semantic representations from the action cache. JigsawNet also contains a vision cache to memorize visual features extracted from video segments. The memorized visual features are also grouped by predicted atomic action types. Different from the action cache, it's difficult to initialize the vision cache, in that the model doesn't have ground truth atomic action label for each video clip. Since the model aims to bridge between visual features and semantic representations, g_y^j will then become a "perfect" expected output for initialization. A threshold ϵ is set as current distance of $\hat{f}_{mem}^{s_x^i} \cdot \hat{g}_{mem}^{s_x^i}$, it enables the memorized feature get replaced by the real extracted visual features after several iterations. The vision cache is only used for optimization during training stages.

4 Experiments

4.1 Datasets

HMDB51 *and* ***UCF101*** HMDB51[26] is a human motion benchmark. It contains 6,849 videos divided into 51 action categories, each category contains a

minimum of 101 clips. UCF101[37] contains 13320 videos divided into 101 sports related actions. For robust and fair comparison, we followed the evaluation procedure proposed in [44]. The model is tested 50 times with 50 randomly generated splits, the average rank1 accuracy and standard deviation are reported for evaluation. In each split, 50% classes are used for training and the rest 50% classes are preserved for testing.

KineticsZSAR Given the size limit of previous ZSAR protocols, [8] proposes a new benchmark with videos and annotations selected from Kinetics400[7] and Kinetics600[6]. The 400 classes of Kinetics400[7] are selected as seen actions and the rest 220 actions from Kinetics600[6] are used as unseen actions for validation and testing. To be specific, we followed [8] to generate three splits. In each split, 60 actions are used for validation and the rest 160 actions are used for testing. The model will be tested three times. Many videos in the original val and test split can't be accessed, so we re-select the videos and preserve the same number of videos for each action to make a fair comparison. The average rank1 accuracy and standard deviation are reported for evaluation.

4.2 Implementation Details

For spatio-temporal extractor backbone $\mathcal{N}_{vis}$, we use a pretrained 34 layers R(2+1)D[39] model, and remove the temporal pooling layers. If not specified,

Table 1. ZSAR performance comparison on HMDB51 and UCF101. *FV* represents fisher vetor, BoW represents bag of words, O represents the model uses object as video attributes, V represents the model uses spatio-temporal features of videos, A represents manually designed attributes, W_N represents class label names, W_D represents descriptions of classes. For fair comparison, the rank1 accuracy (%) and standard deviation (±) are reported. We only list the average rank1 accuracy in table for several methods whose deviations are not provided

Method	Video Input	Action Input	HMDB51↑	UCF101↑
DAP[27]	FV	A	N/A	15.9 ± 1.2
IAP[27]	FV	A	N/A	16.7 ± 1.1
HAA[29]	FV	A	N/A	14.9 ± 0.8
SVE[43]	BoW	W_N	13.0 ± 2.7	10.9 ± 1.5
ESZSL[43]	FV	W_N	18.5 ± 2.0	15.0 ± 1.3
SJE[2]	FV	W_N	13.3 ± 2.4	9.9 ± 1.4
SJE[2]	FV	A	N/A	12.0 ± 1.2
MTE[45]	FV	W_N	19.7 ± 1.6	15.8 ± 1.3
ZSECOC[33]	FV	W_N	22.6 ± 1.2	15.1 ± 1.7
UR[50]	FV	W_N	24.4 ± 1.6	17.5 ± 1.6
O2A[22]	O	W_N	15.6	30.3
ASR[42]	V	W_D	21.8 ± 0.9	24.4 ± 1.0
TS-GCN[17]	O	W_N	23.2 ± 3.0	34.2 ± 3.1
E2E[4]	V	W_N	29.8	44.1
ER[8]	V+O	W_D	35.3 ± 4.6	51.8 ± 2.9
Ours	**V+O**	W_D	**38.7 ± 3.7**	**56.0 ± 3.1**

N_{vis} is initialized with weights pretrained on Kinetics400[7] and IG65M[18] for experiments on KineticsZSAR benchmark, and initialized with weights pretrained on Kinetics605[4] which removes all overlapped actions that appears in HMDB51 and UCF101 for experiments on these two benchmarks. For semantic extractor backbone $\mathcal{N}_{text}$, we use a pretrained 12-layer Bert[10] model. For object recognition backbone $\mathcal{N}_{obj}$, we use a BiT model [25] pretrained on ImageNet21K[9]. The top 5 recognized verbal objects are selected and forwarded to $\mathcal{N}_{text}$ to extract semantic representations. $\mathcal{N}_{obj}$ is frozen during the training stage. All layers of $\mathcal{N}_{vis}$ and last two layers of $\mathcal{N}_{text}$ are finetuned during training stages on KineticsZSAR. For experiments on HMDB51 and UCF101, only the last layer of $\mathcal{N}_{vis}$ and last two layers of $\mathcal{N}_{text}$ are finetuned. The dimension of shared sphere is set as $d = 768$, the threshold ϵ is set as 0.3, and the number of self-attention head $h = 8$. We use SGD with Momentum algorithm for optimization. The weight decay is 5e-4, the momentum is 0.9, and the initial learning rate is 1e-5 on Kinetics ZSAR and 1e-4 on HMDB51 and UCF101. The learning rate is updated with a plateau scheduler which monitors the rank-1 accuracy on validation set, the patience is set as 1, and the min learning rate is set as 1e-9. The model is trained 15 epochs on HMDB51 and UCF101, and 20 epochs on KineticsZSAR. All experiments are made on four TITAN RTX gpus.

4.3 Comparison with State-of-the-art Methods

Table 2. ZSAR performance comparison on KineticsZSAR. O represents the model uses object as video attributes, V represents the model uses spatio-temporal features of videos, W_N represents class label names, W_D represents descriptions of classes

Method	Video Input	Action Input	Rank1 Acc↑	Rank5 Acc↑
DEVISE[15]	V	W_N	23.8 ± 0.3	51.0 ± 0.6
DEM[48]	V	W_N	23.6 ± 0.7	49.5 ± 0.4
ALE[1]	V	W_N	23.4 ± 0.8	50.3 ± 1.4
ESZSL[43]	V	W_N	22.9 ± 1.2	48.3 ± 0.8
SJE[2]	V	W_N	22.3 ± 0.6	48.2 ± 0.4
GCN[19]	V	W_N	22.3 ± 0.6	49.7 ± 0.6
ER[8]	V+O	W_D	42.1 ± 1.4	73.1 ± 0.3
Ours	**V+O**	W_D	**45.9 ± 1.6**	**78.8 ± 1.0**

We make experiments on three benchmarks to evaluate our method against previous SOTAs. Since the N_{vid} needs to extract spatio-temporal features from video segments which has a different temporal receptive field with all off-the-shelf pretrained weights. We need to firstly pretrain the model on Kinetics605[4]. Kinetics605[4] removes the overlapped actions that also appears in HMDB51 or UCF101, so there is no data leakage caused by the pretraining. As is shown in Table 1, our model consistently outperforms previous SOTAs on both benchmarks with noticeable margins. When compared with E2E[4], which also pretrains the spatio-temporal extractor backbone on Kinetics605[4], our method achieves significant better performance with 8.9 and 11.9 gains on HMDB51 and

Table 3. Comparing models w or w/o GA module

GA Module	Rank1 Acc	Rank5 Acc
w/o	41.7 ± 1.3	73.4 ± 0.5
w	45.9 ± 1.6	78.8 ± 1.0

Table 4. Comparing models w or w/o Consistency Loss

Consistency Loss	Rank1 Acc	Rank5 Acc
w/o	43.2 ± 1.9	76.2 ± 1.3
w	45.9 ± 1.6	78.8 ± 1.0

UCF101. When compared with ER[8], which uses the same video and action input formats (Video,Object, and Description) as ours, JigsawNet still continuously outperforms it with 3.4 and 4.2 gains on HMDB51 and UCF101.

We also compare our method against previous SOTAs on the recently released KineticsZSAR[8] benchmark. The $\mathcal{N}_{vis}$ is initialized with weights pretrained on Kinetics400, because the val and test set of KineticsZSAR have no overlap with Kinetics400. We followed the procedures mentioned in [8] to test the model three times on three splits of KineticsZSAR. The average mean and deviation is provided in Table 2 for comparison. Our method consistently outperforms all previous works in Table 2 with noticeable margins. When compared with ER[8], which uses the same input types, our model performs better with 3.8 and 5.7 gains of rank1 and rank5 accuracy.

4.4 Ablation Studies

We also make ablation studies to analyze the contribution of each module to the final performance. All experiments below in this subsection are made on Kinetics ZSAR.

Is group-to-group alignment necessary for zero-shot action recognition? Although our model already achieves SOTA performance on three benchmarks, it's natural for people to ask how much benefit it earns changing from building one-to-one bridge to establishing gorup-to-group bridge? We design a comparison experiment for evaluation. Instead of extracting visual features from groups of video segments, the visual features are now directly extracted from whole videos, which means $\mathcal{F}^i = \mathcal{N}_{vid}(V_i), \mathcal{F}^i \in \mathbb{R}^d$. For action semantic extraction, the semantics are also extracted from entire descriptions instead of groups of verb phrases, $\mathcal{G}^j = \mathcal{N}_{text}(d^j), \mathcal{G}^j \in \mathbb{R}^d$. The one-to-one relationship between i^{th} video v^i and j^{th} action is represented as $p_a^{ij} = \mathcal{F}^i \cdot (\mathcal{G}^j)^T$, $p_o^{ij} = \mathcal{O}^i \cdot (\mathcal{G}^j)^T$,$p^{ij} = p_a^{ij} + \max(p_o^{ij}, 0)$. For fair comparison, Consistency Loss and GE are preserved with slight modifications. For Consistency Loss, the memorized cache are grouped by action types instead of atomic action types. For GE, it will now directly aggregate $\mathcal{F}^i$ and $\mathcal{O}^i$. As is shown in Table 3, the group-to-group model earns 4.2 and 5.4 gains of rank1 and rank5 accuracy, which is pretty significant.

Is the Consistency Loss beneficial? In Table 4, we compare models trained with or without Consistency Loss. As is shown by the results, the model trained

Table 5. Comparing features optimized by Consistency Loss

Optimized Features	Rank1 Acc	Rank5 Acc
spatio-temporal	45.1 ± 1.4	78.0 ± 0.8
aggregated	45.9 ± 1.6	78.8 ± 1.0

Table 6. Comparing models w or w/o GE Module

GE	Rank1 Acc	Rank5 Acc
w/o	42.7 ± 2.0	72.9 ± 1.7
w	45.9 ± 1.6	78.8 ± 1.0

Table 7. Comparing improvements brought by backbone

Model	Backbone	Rank1 Acc	Rank5 Acc
R(2+1)D[39]	ResNet18	42.9 ± 1.7	73.0 ± 1.3
R(2+1)D[39]	ResNet34	45.9 ± 1.6	78.8 ± 1.0
TSM[28]	ResNet50	45.3 ± 1.8	78.4 ± 1.2

with Consistency Loss builds a more robust bridge between vision features and action semantics. It's consistent with our expectation that the model can benefit from modeling inter-sample knowledge.

Which vision feature should be memorized, spatio-temporal feature or fused feature? We have shown the benefits brought by Consistency Loss, however, we want to dig deeper and show why aggregated feature, instead of spatio-temporal features are selected in Consistency Loss. Table 5 compares models with Consistency Loss optimized on aggregated features $\hat{f}_x^i$ and vision features f_x^i. According to the results, we can easily find that the aggregated feature is a better option. We assume it because the integration of object features enables the text extractor $\mathcal{N}_{text}$ also learn from inter-sample knowledge and get optimized by gradients back propagated from Consistency Loss. Meanwhile, the integration of object features enlarges the domain variance, which may also help in this training procedure.

Is the GE module beneficial? Table 6 compares models with or without GE module. According to the result, GE makes noticeable contributions to the final performance. GE not only enlarges the temporal receptive field of visual features extracted from video segments, but enables both spatio-temporal and object features to gain knowledge from a different domain.

How much improvements are from the pretrained R(2+1)D model? In Table7, we compare performance of models with different spatio-temporal extractor backbones. Compared with R(2+1)D-18, the deeper R(2+1)D-34 significantly boosts the performance and outperforms with 3.0 and 5.8 gains of rank1 and rank5 accuracy. We also notice that, when using the same sptio-temporal feature extractor[28] and the same input information types with ER[8], our model still achieves noticeable better performance with 3.2 and 5.3 gains of rank1 and rank5 accuracy.

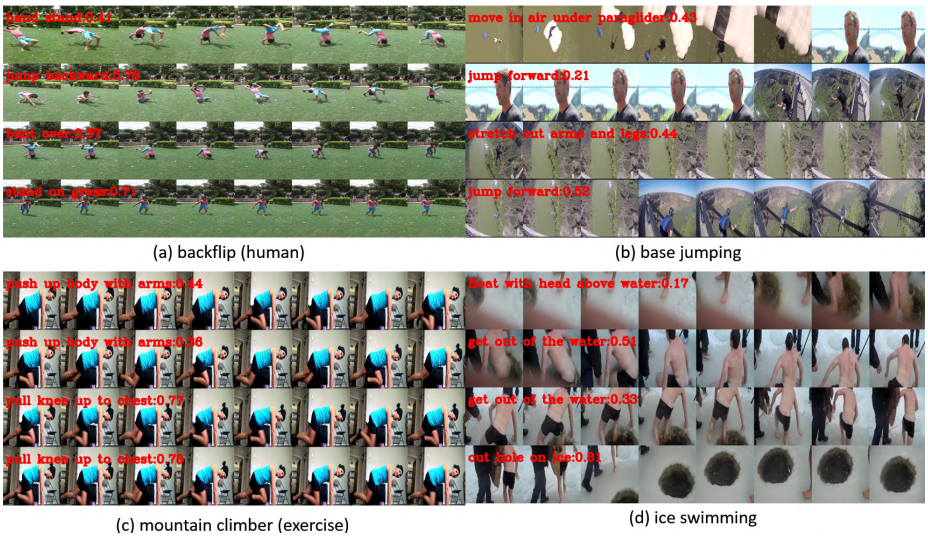

Fig. 4. Visualization of predicted atomic action types of segments when recognizing target actions

4.5 Visualization

To better understand the group-to-group bridge, we visualize the atomic actions our model assigned to each segment. As is shown in Figure 4, the model successfully learned the latent atomic actions unsupervisedly from source actions. For example, the model may learn "hand stand" and "jump backward" from gymnastics actions, and "push up body with arms" from fitness related actions in training set. Meanwhile, we also noticed the benefits brought by intra and inter samples modeling. In Figure 4(c), the model can successfully distinguish between "push up body with arms" and "pull knee up to chest" whose visual features are pretty similar. In Figure 4(d), the model still recognizes the second segment as "get out of the water" with the intra sample knowledge aggregated from contextual frames. Of course, we also found several places which can be improved by future works. For example, current strategy directly splits video into sequential segments, however, the atomic actions may locate between two segments. Although the model can still gain contextual information brought by GE, a soft temporal localization module may further improve the performance.

5 Conclusion

We propose a novel ZSAR model, JigsawNet.

6 Acknowledgment

This research was supported in part by the Defence Science and Technology Agency (DSTA).

References

1. Akata, Z., Perronnin, F., Harchaoui, Z., Schmid, C.: Label-embedding for image classification. IEEE transactions on pattern analysis and machine intelligence **38**(7), 1425–1438 (2015)
2. Akata, Z., Reed, S., Walter, D., Lee, H., Schiele, B.: Evaluation of output embeddings for fine-grained image classification. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2927–2936 (2015)
3. Arnab, A., Dehghani, M., Heigold, G., Sun, C., Lučić, M., Schmid, C.: Vivit: A video vision transformer. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 6836–6846 (2021)
4. Brattoli, B., Tighe, J., Zhdanov, F., Perona, P., Chalupka, K.: Rethinking zero-shot video classification: End-to-end training for realistic applications. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4613–4623 (2020)
5. Caba Heilbron, F., Escorcia, V., Ghanem, B., Carlos Niebles, J.: Activitynet: A large-scale video benchmark for human activity understanding. In: Proceedings of the ieee conference on computer vision and pattern recognition. pp. 961–970 (2015)
6. Carreira, J., Noland, E., Banki-Horvath, A., Hillier, C., Zisserman, A.: A short note about kinetics-600. arXiv preprint arXiv:1808.01340 (2018)
7. Carreira, J., Zisserman, A.: Quo vadis, action recognition? a new model and the kinetics dataset. In: proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 6299–6308 (2017)
8. Chen, S., Huang, D.: Elaborative rehearsal for zero-shot action recognition. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 13638–13647 (2021)
9. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition. pp. 248–255. Ieee (2009)
10. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
11. Diba, A., Fayyaz, M., Sharma, V., Hossein Karami, A., Mahdi Arzani, M., Yousefzadeh, R., Van Gool, L.: Temporal 3d convnets using temporal transition layer. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. pp. 1117–1121 (2018)
12. Diba, A., Fayyaz, M., Sharma, V., Karami, A.H., Arzani, M.M., Yousefzadeh, R., Van Gool, L.: Temporal 3d convnets: New architecture and transfer learning for video classification. arXiv preprint arXiv:1711.08200 (2017)
13. Fan, L., Buch, S., Wang, G., Cao, R., Zhu, Y., Niebles, J.C., Fei-Fei, L.: Rubiksnet: Learnable 3d-shift for efficient video action recognition. In: Proceedings of the European Conference on Computer Vision (ECCV) (2020)
14. Floridi, L., Chiriatti, M.: Gpt-3: Its nature, scope, limits, and consequences. Minds and Machines **30**(4), 681–694 (2020)
15. Frome, A., Corrado, G.S., Shlens, J., Bengio, S., Dean, J., Ranzato, M., Mikolov, T.: Devise: A deep visual-semantic embedding model. Advances in neural information processing systems **26** (2013)
16. Gan, C., Lin, M., Yang, Y., De Melo, G., Hauptmann, A.G.: Concepts not alone: Exploring pairwise relationships for zero-shot video activity recognition. In: Thirtieth AAAI conference on artificial intelligence (2016)

17. Gao, J., Zhang, T., Xu, C.: I know the relationships: Zero-shot action recognition via two-stream graph convolutional networks and knowledge graphs. In: Proceedings of the AAAI conference on artificial intelligence. vol. 33, pp. 8303–8311 (2019)
18. Ghadiyaram, D., Tran, D., Mahajan, D.: Large-scale weakly-supervised pre-training for video action recognition. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 12046–12055 (2019)
19. Ghosh, P., Saini, N., Davis, L.S., Shrivastava, A.: All about knowledge graphs for actions. arXiv preprint arXiv:2008.12432 (2020)
20. Guo, M., Chou, E., Huang, D.A., Song, S., Yeung, S., Fei-Fei, L.: Neural graph matching networks for fewshot 3d action recognition. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 653–669 (2018)
21. Hara, K., Kataoka, H., Satoh, Y.: Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet? In: Proceedings of the IEEE conference on Computer Vision and Pattern Recognition. pp. 6546–6555 (2018)
22. Jain, M., Van Gemert, J.C., Mensink, T., Snoek, C.G.: Objects2action: Classifying and localizing actions without any video example. In: Proceedings of the IEEE international conference on computer vision. pp. 4588–4596 (2015)
23. Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., Fei-Fei, L.: Large-scale video classification with convolutional neural networks. In: Proceedings of the IEEE conference on Computer Vision and Pattern Recognition. pp. 1725–1732 (2014)
24. Kay, W., Carreira, J., Simonyan, K., Zhang, B., Hillier, C., Vijayanarasimhan, S., Viola, F., Green, T., Back, T., Natsev, P., et al.: The kinetics human action video dataset. arXiv preprint arXiv:1705.06950 (2017)
25. Kolesnikov, A., Beyer, L., Zhai, X., Puigcerver, J., Yung, J., Gelly, S., Houlsby, N.: Big transfer (bit): General visual representation learning. In: European conference on computer vision. pp. 491–507. Springer (2020)
26. Kuehne, H., Jhuang, H., Garrote, E., Poggio, T., Serre, T.: HMDB: a large video database for human motion recognition. In: Proceedings of the International Conference on Computer Vision (ICCV) (2011)
27. Lampert, C.H., Nickisch, H., Harmeling, S.: Learning to detect unseen object classes by between-class attribute transfer. In: 2009 IEEE conference on computer vision and pattern recognition. pp. 951–958. IEEE (2009)
28. Lin, J., Gan, C., Han, S.: Tsm: Temporal shift module for efficient video understanding. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 7083–7093 (2019)
29. Liu, J., Kuipers, B., Savarese, S.: Recognizing human actions by attributes. In: CVPR 2011. pp. 3337–3344. IEEE (2011)
30. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 10012–10022 (2021)
31. Mandal, D., Narayan, S., Dwivedi, S.K., Gupta, V., Ahmed, S., Khan, F.S., Shao, L.: Out-of-distribution detection for generalized zero-shot action recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9985–9993 (2019)
32. Qian, Y., Kang, G., Yu, L., Liu, W., Hauptmann, A.G.: Trm: Temporal relocation module for video recognition. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 151–160 (2022)

33. Qin, J., Liu, L., Shao, L., Shen, F., Ni, B., Chen, J., Wang, Y.: Zero-shot action recognition with error-correcting output codes. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2833–2842 (2017)
34. Shao, H., Qian, S., Liu, Y.: Temporal interlacing network. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 34, pp. 11966–11973 (2020)
35. Simonyan, K., Zisserman, A.: Two-stream convolutional networks for action recognition in videos. In: Advances in neural information processing systems. pp. 568–576 (2014)
36. Smaira, L., Carreira, J., Noland, E., Clancy, E., Wu, A., Zisserman, A.: A short note on the kinetics-700-2020 human action dataset. arXiv preprint arXiv:2010.10864 (2020)
37. Soomro, K., Zamir, A.R., Shah, M.: Ucf101: A dataset of 101 human actions classes from videos in the wild. arXiv preprint arXiv:1212.0402 (2012)
38. Tran, D., Bourdev, L., Fergus, R., Torresani, L., Paluri, M.: Learning spatiotemporal features with 3d convolutional networks. In: Proceedings of the IEEE international conference on computer vision. pp. 4489–4497 (2015)
39. Tran, D., Wang, H., Torresani, L., Ray, J., LeCun, Y., Paluri, M.: A closer look at spatiotemporal convolutions for action recognition. In: Proceedings of the IEEE conference on Computer Vision and Pattern Recognition. pp. 6450–6459 (2018)
40. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017)
41. Wang, L., Xiong, Y., Wang, Z., Qiao, Y., Lin, D., Tang, X., Van Gool, L.: Temporal segment networks: Towards good practices for deep action recognition. In: European conference on computer vision. pp. 20–36. Springer (2016)
42. Wang, Q., Chen, K.: Alternative semantic representations for zero-shot human action recognition. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases. pp. 87–102. Springer (2017)
43. Xu, X., Hospedales, T., Gong, S.: Semantic embedding space for zero-shot action recognition. In: 2015 IEEE International Conference on Image Processing (ICIP). pp. 63–67. IEEE (2015)
44. Xu, X., Hospedales, T., Gong, S.: Transductive zero-shot action recognition by word-vector embedding. International Journal of Computer Vision **123**(3), 309–333 (2017)
45. Xu, X., Hospedales, T.M., Gong, S.: Multi-task zero-shot action recognition with prioritised data augmentation. In: European Conference on Computer Vision. pp. 343–359. Springer (2016)
46. Zellers, R., Choi, Y.: Zero-shot activity recognition with verb attribute induction. arXiv preprint arXiv:1707.09468 (2017)
47. Zhang, H., Hao, Y., Ngo, C.W.: Token shift transformer for video classification. In: Proceedings of the 29th ACM International Conference on Multimedia. pp. 917–925 (2021)
48. Zhang, L., Xiang, T., Gong, S.: Learning a deep embedding model for zero-shot learning. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2021–2030 (2017)
49. Zhou, B., Andonian, A., Oliva, A., Torralba, A.: Temporal relational reasoning in videos. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 803–818 (2018)
50. Zhu, Y., Long, Y., Guan, Y., Newsam, S., Shao, L.: Towards universal representation for unseen action recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 9436–9445 (2018)