ZSTAD: Zero-Shot Temporal Activity Detection

Lingling Zhang[1,2], Xiaojun Chang[3]* Jun Liu[1,4], Minnan Luo[1,4], Sen Wang[5],
Zongyuan Ge[3], Alexander Hauptmann[6]

[1]School of Computer Science and Technology, Xi'an Jiaotong University, Xian, China
[2]Ministry of Education Key Lab For Intelligent Networks and Network Security, Xian, China
[3]Faculty of Information Technology, Monash University, Australia
[4]National Engineering Lab for Big Data Analytics, Xi'an Jiaotong University, Xian, China
[5]School of Information Technology and Electrical Engineering, The University of Queensland, Australia
[6]School of Computer Science, Carnegie Mellon University, USA

zhanglingling@stu.xjtu.edu.cn, cxj273@gmail.com, {liukeen, minnluo}@xjtu.edu.cn,
sen.wang@uq.edu.au, zongyuan.ge@monash.edu, alex@cs.cmu.edu

Abstract

An integral part of video analysis and surveillance is temporal activity detection, which means to simultaneously recognize and localize activities in long untrimmed videos. Currently, the most effective methods of temporal activity detection are based on deep learning, and they typically perform very well with large scale annotated videos for training. However, these methods are limited in real applications due to the unavailable videos about certain activity classes and the time-consuming data annotation. To solve this challenging problem, we propose a novel task setting called zero-shot temporal activity detection (ZSTAD), where activities that have never been seen in training can still be detected. We design an end-to-end deep network based on R-C3D as the architecture for this solution. The proposed network is optimized with an innovative loss function that considers the embeddings of activity labels and their super-classes while learning the common semantics of seen and unseen activities. Experiments on both the THUMOS'14 and the Charades datasets show promising performance in terms of detecting unseen activities.

1. Introduction

Given its importance to video analysis and surveillance, temporal activity detection is one of the most studied tasks in computer vision [31]. Most videos are untrimmed and only contain a few interesting activities among a long stream of nondescript scenes. Hence, the goal of temporal activity detection is to simultaneously recognize and categorize specific activities in a video along with their start and end times [32, 12]. As with many other tasks, deep learning has led to a step-change in the speed and accuracy of temporal activity detection, as demonstrated in studies like [48, 10, 9, 41, 6]. However, these deep methods rely heavily on a fully-supervised training scheme. Long videos where every activity class is already annotated are rare, and manual annotation is expensive and time-consuming, which means an alternative approach, like weakly-supervised [39, 25], semi-supervised or unsupervised learning, is needed.

In recent years, zero-shot learning (ZSL) has emerged as a novel approach to this very problem. ZSL can be thought of as a special case of supervised learning, where the training and testing classes (*i.e.*, the seen and unseen classes) are disjoint but semantically related [23, 28]. In fact, generally speaking, the training dataset does not even contain any instances labeled with testing classes [20, 19]. ZSL works by exploiting useful prior knowledge, such as common attributes or label embeddings, and then transfers the learned knowledge from seen classes to unseen classes. So far, studies on ZSL have been dominated by zero-shot recognition (ZSR), *i.e.*, classification problems. These approaches are often based on the assumption that each sample only contains a single dominant example from one of the classes (whether seen or unseen). Hence, the overall aim is to classify each unseen sample to an unseen class in the testing stage. On this proviso, ZSR would not be directly applicable to temporal activity detection for two reasons: (1) Temporal activity detection is of most value with long videos, which means the sample would almost certainly contain multiple activities. (2) Part of temporal activity detection is predicting the start and end times for each activity in ad-

---
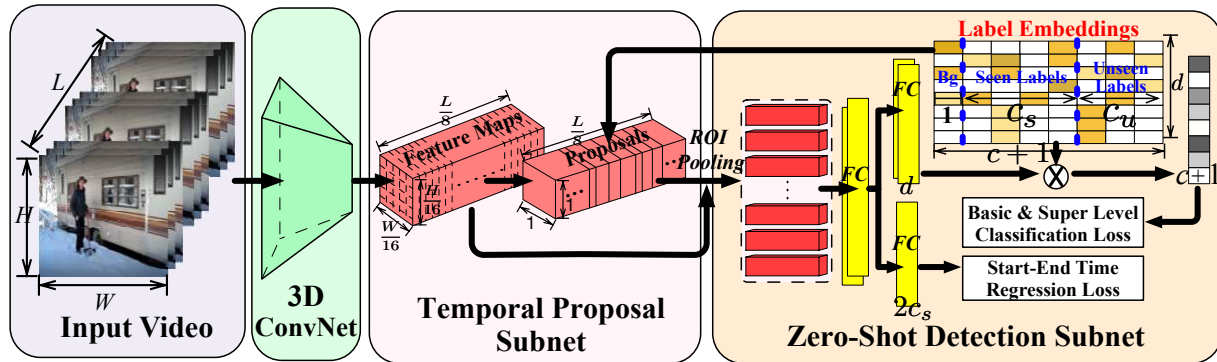*Corresponding author: Xiaojun Chang, cxj273@gmail.com

Figure 1: Diagram of zero-shot temporal activity detection network. Purple part: the input videos are resized into fixed dimensionality; Green part: 3D ConNet extracts the deep features for input videos; Pink part: Temporal proposal subnet (TPN) generates the proposals for seen and unseen activities; Orange part: Zero-shot detection subnet (ZSDN) detects the seen and unseen activities by utilizing their label embeddings.

dition to recognizing the class of behavior. Thus, in this paper, we present a novel task setting for ZSR, called zero-shot temporal activity detection (ZSTAD), that is designed specifically for long untrimmed videos and can also predict the temporal information surrounding those activities.

Unlike simple ZSR, ZSTAD not only recognizes activities that are not seen during training, but it also localizes the start and end times of the activities. ZSTAD is implemented in a novel end-to-end deep network, as shown in Figure 1. The network uses two types of prior information to mine the common semantics of seen and unseen activities: label embeddings from Word2Vec [11] and the superclasses of those labels. For ZSTAD, the label embedding for the background class is crucial, however, it does not directly correspond to any embeddings in Word2Vec. Hence we solve an optimization problem to obtain a special vector for the background. Note that this learned embedding is far away from all the activity label embeddings in the semantic space. With this prior information, the ZSTAD deep network thus consists of four parts: (1) the input process, which resizes the original video into a sequence of RGB frames with fixed dimensionality; (2) a 3D convolutional subnet (3D ConvNet), which extracts the deep features from the input videos; (3) a temporal proposal subnet (TPN), which uses the learned deep features and the embedding of the background class as inputs to generate proposals for seen and unseen activities; and (4) a zero-shot detection subnet (ZSDN), which uses the label embeddings of all activities to predict a proposal's class and start/end times.

To optimize the deep network, we design an innovative loss that combines the classification loss and boundary regression loss of both the TPN and ZSDN subnets. Note that the classification loss for ZSDN consists of two terms: the basic-class classification loss and the super-class clustering loss. The basic-class classification loss guarantees that the score for the true label is the largest of all labels. The super-

class clustering loss controls the labels that are similar to the true label, to obtain the higher scores than other labels. Four contributions of the paper are summarized as follows:

- We propose a new task setting called ZSTAD to simultaneously classify and localize the activities in long untrimmed videos, even if the activity classes have never been seen during training.

- A novel end-to-end deep network is developed for ZSTAD task, which uses prior information from label embeddings of the background, seen activities, and unseen activities. The background label embedding is derived by solving an optimization problem and is far away from the other label embeddings.

- We design an innovative classification loss for ZSDN subnet, which integrates the basic-class classification loss and super-class clustering loss together. Note that the super-class prior information carries the semantic correlation among activity labels.

- We conduct extensive experiments on two widely-used video datasets, THUMOS'14 and Charades, to evaluate the effectiveness and superiority of the proposed deep model.

2. Related Work

2.1. Temporal Activity Detection

The literature on temporal activity detection can be divided into two groups according to how the activity proposals are generated. In the first category, activity classifiers are applied to sliding windows generated by scanning across the video. For example, Wang *et al.* [38] used a temporal sliding window to generate video clips and then extracted motion and appearance features for action classification. Shou *et al.* [33] generated segments with varied

lengths via sliding windows, then fed them into 3D ConvNets for action recognition. And Gaidon *et al.* [8] described a sliding central frame based on a generative model of temporal structures. These sliding window methods not only have a high computation cost due to the large number of windows, but also constrain the boundary of detected activity segments to some extent [42].

To avoid these issues, the second category of methods detects activities with arbitrary lengths by modeling the temporal evolution of the video. For instance, Gao *et al.* [10] presented an effective temporal unit regression network for generating temporal action proposals. Xiong *et al.* [40] introduced a new proposal generation scheme called temporal actionness grouping that can efficiently generate candidates with accurate temporal boundaries. Xu *et al.* [41] put forward the first end-to-end activity detection model, RC3D, by combining the proposal generation and classification stages together. Compared to the sliding window methods, these temporal evolution methods have attracted more attention because of their added flexibility in predicting arbitrary activity start and end times.

2.2. Zero-Shot Learning

Zero-shot learning (ZSL) is designed to recognize samples of classes that are not seen during training [50, 45, 5, 17]. The idea is to learn shared knowledge from prior information and then transfer that knowledge from seen classes to unseen classes [21, 27, 2, 4, 3, 18, 46]. Common attributes, such as color, shape, and similar properties, are the typical forms of prior information. Lampert *et al.* [16] prelearned the attribute classifiers independently to accomplish ZSL on unseen classes, while Parikh *et al.* [24] learned relative attributes to enhance ZSL accuracy. Attribute-based methods have achieved promising results on ZSL, but they have poor scalability because the attributes need to be manually defined. Semantic embeddings of seen and unseen labels, which are another type of prior information, do not have this problem [43]. They are generally learned in an unsupervised manner with a method such as Word2Vec [11] or GloVe [26]. Socher *et al.* [36], for example, mapped samples and labels into a shared semantic space and introduced a binary variable to indicate whether an instance was in seen or unseen class. Zhang *et al.* [47] developed a semantic similarity embedding method by expressing seen and unseen data as a mixture of seen class proportions. Compared to the attribute-based methods, label embedding methods are more practical and popular because this type of prior information is easily accessible from open text corpora.

Beyond these studies, which focus on ZSR problems, four notable studies on zero-shot object detection (ZSOD) have recently appeared in the image processing literature. Zhu *et al.* [49] presented a novel ZSOD architecture that fused semantic attribute information with visual features

to predict the locations of unseen objects. Bansal *et al.* [1] proposed a background-aware ZSOD model based on visual-semantic embeddings. Demirel *et al.*'s [7] solution for ZSOD aggregated both label embeddings and convex combinations of semantic embeddings. Last, Rahman *et al.* [29] has proposed the first end-to-end deep network for ZSOD based on the Faster RCNN [30] framework.

To the best of our knowledge, this paper is the first to apply the idea of ZSL to temporal activity detection.

3. Zero-Shot Temporal Activity Detection

This section begins with a description of the problem setting of ZSTAD. We then introduce the detailed architecture of the proposed deep network, and design a novel objective loss to optimize the network's parameters.

3.1. Problem Description

In the framework of ZSTAD, there are n untrimmed videos $\mathcal{X} = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n\}$ about c activity classes. The first n_s videos $\mathcal{X}_s = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_{n_s}\}$ are labeled for training with temporal annotations that cover the first c_s seen activity classes. The remaining n_u videos $\mathcal{X}_u = \{\mathbf{x}_{n_s+1}, \mathbf{x}_{n_s+2}, \cdots, \mathbf{x}_n\}$ are unlabeled for testing, with each containing at least one activity in c_u unseen classes. Motivated by previous studies on ZSL, label embeddings generated from an unsupervised method, such as Word2Vec, are still used to measure the semantic relationships in both seen and unseen classes. The label embedding set of c_s seen activities is denoted as $\mathcal{L}_s = \{\ell_1, \ell_2, \cdots, \ell_{c_s}\}$, accordingly, $\mathcal{L}_u = \{\ell_{c_s+1}, \ell_{c_s+2}, \cdots, \ell_c\}$ for c_u unseen activities. Especially, the label embedding for background class is not directly available from Word2Vec and, without it, the model cannot determine whether the video contains any activity or not. Therefore, we propose the following optimization problem to overcome this problem:

$$\ell_{bg} = \arg\min_{\ell_{bg}} \sum_{j=1}^{c} max(0, s(\ell_{bg}, \ell_j) - \Delta_{bg})^2, \quad (1)$$

where Δ_{bg} is the margin hyper-parameter. Solving this problem results in a special vector ℓ_{bg} that is far away from all embeddings of activity labels. The function $s(\ell_{bg}, \ell_j)$ measures the semantic similarity between label embeddings ℓ_{bg} and ℓ_j, which is calculated according to cosine distance. In effect, this equation stipulates that the background label is at most Δ_{bg} similar to other activity labels in the semantic embedding space. The resulting set of all label embeddings is denoted as $\mathcal{L}_{bg} = \{\ell_{bg}, \ell_1, \ell_2, \cdots, \ell_c\}$.

Rahman *et al.* [29] proved that super-classes over label embeddings benefit ZSL. Therefore, we have incorporated super-classes into the embeddings $\mathcal{L}_{bg} = \{\ell_{bg}, \ell_1, \ell_2, \cdots, \ell_c\}$ as another type of prior information in
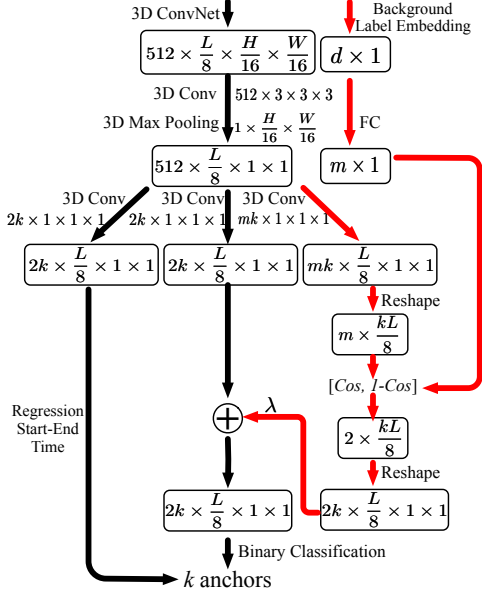
Figure 2: Diagram of the temporal proposal subnet.

the ZSTAD framework. More specifically, we partition c activity label embeddings into c^+ disjoint super-classes such that the activity labels in the same super-class share high semantic similarity, while ensuring that the labels from different super-classes are as dissimilar as possible. These activity super-classes are denoted as $\mathcal{Z} = \{z_1, z_2, \cdots, z_{c^+}\}$, where z_q is a set of label indexes whose corresponding embeddings are partitioned into the q-th super-class. Namely, $z_q = \{j \in [1, c], s.t., g(\ell_j) = q\}$, where the function $g(\cdot)$ maps each activity label embedding to its corresponding super-class $z_{g(.)}$. Practically, the mapping function $g(\cdot)$ could be performed by any number of popular clustering algorithms. Further, the background label embedding ℓ_{bg} is set into an individual super-class z_{bg} to ensure that it is clearly different from the other activity labels in the semantic embedding space. Ultimately, the overall super-classes is represented as $\mathcal{Z}_{bg} = \{z_{bg}, z_1, z_2, \cdots, z_{c^+}\}$.

To summarize, the ZSTAD model is trained on labeled videos $\mathcal{X}_s$ with temporal annotations of seen activities, with the objective of recognizing and localizing unseen activities in unlabeled videos $\mathcal{X}_u$. The common semantics between seen and unseen activities are reflected in the prior information from the label embeddings $\mathcal{L}_{bg}$ and super-classes $\mathcal{Z}_{bg}$.

3.2. Network Architecture

The deep network for ZSTAD is illustrated in Figure 1. Its backbone is the R-C3D framework [41] due to the R-C3D's superior performance in temporal activity detection. The four colored panels delineate the four essential components. The purple panel shows the input process, where videos are transformed into a sequence of RGB frames with

dimension $\mathbb{R}^{3 \times L \times H \times W}$. Here, L denotes the number of frames and H, W represent the height and width of each frame. The 3D ConvNet (the green panel) extracts the deep features from the input videos. The architecture of this subnet is similar to that proposed in [37], *i.e.*, eight 3D convolutional layers (conv1a to conv5b) and four max-pooling layers (pool1 to pool4), and the output is a feature map with dimension $\mathbb{R}^{512 \times \frac{L}{8} \times \frac{H}{16} \times \frac{W}{16}}$. The pink panel houses the TPN, which takes the video's feature map as its input and outputs proposal segments of variable lengths. The ZSDN, in the orange panel, recognizes the classes of the generated proposals and fine-tunes their start/end times. The two key components of the framework are obviously TPN and ZSDN. These are discussed in more detail in the following sections.

3.2.1 Temporal Proposal Subnet (TPN)

The architecture of the TPN subnet is illustrated in detail in Figure 2. This subnet generates high-quality proposals and can correctly distinguish whether the proposal contains an example of any activity. Its anchor segments are predefined multiscale windows centered on $L/8$ uniformly distributed temporal locations, where the maximum possible number of proposals for each location is k. Therefore, the total number of segments for each input video is $kL/8$. The output of the 3D ConvNet first passes through a 3D convolutional filter and a max-pooling layer, which is represented as a temporal-only feature map $C_{tpn} \in \mathbb{R}^{512 \times \frac{L}{8} \times 1 \times 1}$. Namely the 512-dimensional feature is produced for each temporal location. This feature map is then fed into three modules: a boundary regression module (left flow), a basic classification module (center flow), and an improved classification module (right flow). The boundary regression module is used to predict the offset and length of each segment. The purpose of two binary classification modules is to score each proposal as either an activity or background. Their outputs are added together to jointly determine the final scores.

Note that the architectures of both the boundary regression module and the basic classification module are the same as R-C3D. The improved classification module was designed by us to refine the results produced by the basic module by introducing the background label embeddings. The improved classification module contains an additional 3D convolutional layer with mk channels on the feature map C_{tpn}. The output is then reshaped as $m \times (kL/8)$, *i.e.*, each anchor segment is represented as an m-dimensional feature vector. The d-dimensional background label embedding is also mapped into m-dimensional space with a fully-connected layer. In this way, the improved classification module determines its binary scores by computing the cosine distance between m-dimensional features of the anchor segment and the background label embedding. The improved classification module can decrease the mistake

of regarding unseen activities as background in the testing stage, because the label embedding of background is far away from that of unseen activities.

3.2.2 Zero-Shot Detection Subnet (ZSDN)

The TPN subnet generates a set of candidate activity proposals by automatically ranking the confidence scores of all anchor segments. The non-maximum suppression algorithm is applied first to eliminate the proposals with large overlaps or low confidence scores. Next, three-dimensional region of interest (3D RoI) pooling with $1 \times 4 \times 4$ grids maps the selected proposals, which are of different lengths, into a fixed size of $512 \times 1 \times 4 \times 4$. Each proposal's pooled feature passes through two fully-connected layers, and then onwards into a zero-shot classification module and a boundary regression module. Descriptions follow.

- The zero-shot classification module includes two fully-connected layers that project the input proposal's features onto d-dimensional semantic space. A score vector $\mathbf{p}_i \in \mathbb{R}^{c+1}$ is calculated for each proposal by computing the cosine distance between the proposal's feature vector and all label embeddings $\mathcal{L}_{bg}$. Vector $\mathbf{p}_i$ consists of the possibilities that the proposal belongs to the background class and the c activity classes.

- The boundary regression module is one fully-connected layer at the top of the network, which generates $2 \times c_s$ outputs. These outputs are used to refine each proposal's start/end times.

3.3. Training and Inference

The network is trained by jointly optimizing the classification and regression losses for both the TPN and ZSDN subnets. Note that three losses, including the classification and regression losses of the TPN and the regression loss of the ZSDN, are set as the work in [41]. This last loss, *i.e.*, the classification loss of the ZSDN subnet, is integral to accomplishing the ZSTAD task, which is designed as follows:

$$L_{zs-cls} = \frac{1}{N} \sum_i (L_{bc}(\mathbf{p}_i, p_i^*) + \beta L_{sc}(\mathbf{p}_i, p_i^*)), \quad (2)$$

where N denotes the number of proposals in the ZSDN subnet, and i stands for their index during the training procedure. The vector $\mathbf{p}_i \in \mathbb{R}^{c+1}$ is the output of the ZSDN's zero-shot classification module. It represents the predicted probability distribution over the background and c activity classes. The integer $p_i^* \in \{0, 1, \cdots, c\}$ represents the ground-truth label of the i-th proposal. If $p_i^* = 0$, the proposal does not contain any activities, otherwise it belongs to the p_i^*-th activity class. On the whole, the classification loss $\mathcal{L}_{zs-cls}$ consists of two components: a basic-class classification loss $\mathcal{L}_{bc}$ and a super-class clustering loss $\mathcal{L}_{sc}$. β is

a hyper-parameter that controls the trade-off between these two terms. Given a proposal, the basic-class classification loss guarantees that the predicted score for its true class will be the largest in the $(c+1)$-dimensional score vector $\mathbf{p}_i$. It is evaluated with a general softmax loss as follows:

$$L_{bc}(\mathbf{p}_i, p_i^*) = -\log p_{ip_i^*}. \quad (3)$$

The L_{sc} clustering loss considers the prior information from the super-classes $\mathcal{Z}_{bg}$ of the activities. It controls the predicted scores of activity labels in the same super-class as the true label, should be higher than those of other activity labels in different super-classes. The value L_{sc} is derived from the following hinge loss:

$$L_{sc}(\mathbf{p}_i, p_i^*) = \frac{1}{N_{sc}} \sum_{j_1 \notin z_{s_i^*}} \sum_{j_2 \in z_{s_i^*}} \max(0, p_{ij_1} - p_{ij_2} + \Delta_{sc}),$$
$$(4)$$

where Δ_{sc} is a margin hyper-parameter that can be determined through cross-validation. The integer s_i^* is the super-class index of the ground-truth activity label embedding $\ell_{p_i^*}$, *i.e.* $s_i^* = g(\ell_{p_i^*})$. The number of N_{sc} is equal to $(c+1-|z_{s_i^*}|) \times |z_{s_i^*}|$, where $|z_{s_i^*}|$ denotes the total number of activity labels in the super-class $z_{s_i^*}$.

Note that no relative time offsets and lengths are determined for unseen activities because no unseen activity samples are included in the training procedure. Therefore, the testing stage simply produces a $(2 \times c_s)$ matrix denoting the two parameterized coordinates for each seen activity class. In this case, the relative offset and length of unseen activity are approximated through the coordinates of closely related seen activities. This strategy was first proposed by Rahman *et al.* [29], and it has proven to be effective for zero-shot detection because the visual features of unseen activities are usually similar to those of their close seen activities.

4. Experiment

4.1. Datasets

We conduct experiments with two video datasets: THUMOS'14 and Charades. Their details follow.

- **THUMOS'14** [14]: This dataset contains 20 activity classes for temporal activity detection and consists of four parts: training data, validation data, testing data, and background data. We use the validation data (200 untrimmed videos) to train our network and the test data (213 untrimmed videos) to evaluate the model's performance. 12 activities are selected as seen classes with the remaining 8 activities as the unseen classes.

- **Charades** [35]: This is another widely-used dataset for activity recognition and detection. It comprises 9848 videos over 157 daily indoor activities collected

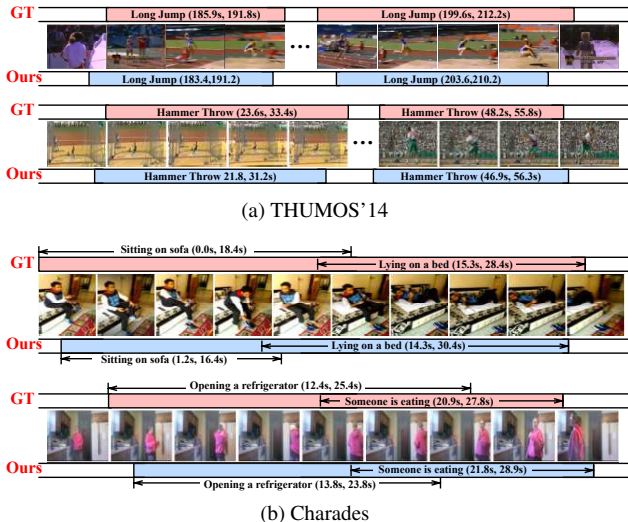(a) THUMOS'14

(b) Charades

Figure 3: Some prediction results of zero-shot temporal activity detection over two datasets.

through Amazon Mechanical Turk. We use 7985 videos for training and 1863 for testing. 120 activities are chosen as seen classes; 37 activities are unseen.

As described in Section 3.1, the two types of prior information used are the activity label embeddings and their super-classes. The labels in THUMOS'14 consist of one or two nouns, so we use the average of their 300-dimensional features from Word2Vec for the activity label embeddings. The labels in Charades are gerund phrases, such as "Taking a picture of something". To ensure the accuracy of label embeddings, we remove the quantifiers and prepositions from these phases, then represent the remaining words as mean Word2Vec features. The background label embedding is obtained by solving the optimization problem Eq. (1) with a parameter setting of $\Delta_{bg} = 0.1$. The resulting activity label embeddings are then grouped into several super-classes with the self-tuning spectral clustering algorithm [44]. The background label is assigned into a individual super class because it is far away from other labels.

4.2. Experimental Setup

To limit the GPU memory consumed, the network input with the THUMOS'14 dataset is set to a sequence of RGB frames with a dimension of $3 \times 512 \times 112 \times 112$, *i.e.*, the equivalent of a 30-second video clip. The input with the Charades dataset is limited to $3 \times 768 \times 112 \times 112$, which is about 154 seconds. Note that the ZSL setting demands that the training dataset does not contain any instances labeled with testing classes. Therefore, we remove any clips containing unseen activities from the training video clips. In addition, we ensure that each clip in the testing set contains at least one unseen activity. We build the network in the open-

Table 1: Zero-shot temporal activity detection results on THUMOS'14 w.r.t mAP (%) at different IoU thresholds.

	α=0.1	α=0.2	α=0.3	α=0.4	α=0.5
R-C3D+SE	13.96	12.61	10.81	7.91	5.11
R-C3D+ConSE	14.16	12.54	10.93	8.02	5.29
Ours ($-TPN_*-L_{sc}$)	16.76	14.76	11.87	9.01	7.37
Ours ($+TPN_*-L_{sc}$)	17.92	15.03	12.99	9.61	8.25
Ours ($-TPN_*+L_{sc}$)	19.50	16.72	13.81	11.23	8.88
Ours ($+TPN_*+L_{sc}$)	21.34	16.98	15.01	11.12	9.15

source Caffe framework [13], and the parameters of the 3D ConvNet is pre-trained on the Sports-1M dataset [15] to avoid overfitting. The number of anchor segments k are set to values within [2,4,5,6,8,9,10,12,14,16] for THUMOS'14 and within [1,2,3,4,5,6,7,8,10,12,14,16,20,24,28,32,40,48] for Charades. The three hyper-parameters are set as follows: the trade-off in the improved TPN subnet is set to $\lambda = 0.6$; the trade-off in zero-shot classification loss is set to $\beta = 0.1$ (Eq. 2); and the margin Δ_{sc} in the clustering loss function is set from [0.05,0.30] for the THUMOS'14 dataset and from [0.03,0.15] for Charades (Eq. (4)). Training is conducted with stochastic gradient descent at a learning rate of 0.0001, a momentum of 0.9, and a weight decay of 0.00005 to optimize the proposed deep ZSTAD network.

4.3. Comparative Results

Given this study is the first work in the novel direction of ZSTAD, there are no existing methods with which to compare our model. Therefore, we design two baselines called R-C3D+SE and R-C3D+ConSE by combining the R-C3D activity detection model with well-known ZSR frameworks SE [43] and ConSE [22] together. Note that SE is a primary method for zero-shot action recognition, by using semantic word vector space as the common space to embed videos and category labels. ConSE projects samples into a semantic embedding space via a convex combination of class label embeddings. In addition, we compare four versions of our model with the designed baseline to explore its efficacy. The four versions are:

- ($-TPN_*-L_{sc}$) without the improved classification module in TPN subnet and without super-class clustering loss for ZSDN subnet;
- ($+TPN_*-L_{sc}$) with the improved classification module in TPN subnet and without super-class clustering loss for ZSDN subnet;
- ($-TPN_*+L_{sc}$) without the improved classification module in TPN subnet and with super-class clustering loss for ZSDN subnet;
- ($+TPN_*+L_{sc}$) with the improved classification module in TPN subnet and with super-class clustering loss for ZSDN subnet.

Table 2: Per-unseen class AP (%) at IoU threshold $\alpha = 0.5$ on THUMOS'14 dataset.

	Baseball Pitch	Cricket Bowling	Diving	Hammer Throw	Long Jump	Shotput	Soccer Penalty	Tennis Swing
R-C3D+SE	2.23	3.09	3.13	9.21	12.15	3.42	3.38	4.29
R-C3D+ConSE	2.21	3.07	3.23	9.53	12.54	3.56	3.46	4.72
Ours ($-TPN_*$-L_{sc})	3.79	4.03	4.41	14.25	17.47	4.98	4.92	5.11
Ours ($+TPN_*$-L_{sc})	3.92	4.24	4.87	15.92	19.78	5.64	5.36	6.30
Ours ($-TPN_*$+L_{sc})	4.20	4.62	5.07	17.02	21.23	6.31	5.82	6.78
Ours ($+TPN_*$+L_{sc})	4.34	4.87	5.03	18.12	20.78	7.06	6.03	6.93

Table 3: Zero-shot temporal activity detection results on Charades w.r.t standard and post-process mAP (%).

	Standard mAP	Post-Process mAP
R-C3D+SE	5.13	9.17
R-C3D+ConSE	5.67	9.84
Ours ($-TPN_*$-L_{sc})	6.63	10.89
Ours ($+TPN_*$-L_{sc})	7.03	11.72
Ours ($-TPN_*$+L_{sc})	7.57	12.86
Ours ($+TPN_*$+L_{sc})	7.91	13.23

Results on THUMOS'14: Table 1 reports the detection performance for the eight unseen activities in terms of mean average precision (mAP) at IoU thresholds $[0.1, 0.5]$ (denoted as α). The average precision (AP) for each unseen class at IoU threshold 0.5 is shown in Table 2. From these results, we make the following four observations:

- All four versions of our method consistently perform better than the R-C3D+SE and R-C3D+ConSE baselines over each of the unseen class. This indicates that considering label embeddings in the ZSDN subnet is beneficial to ZSTAD task.
- In most cases, the performance with TPN_* is better than without. This illustrates that the new classification module in the TPN subnet, which introduces information on the background label embedding, is helpful for generating unseen activity proposals.
- Comparing the results of "$+L_{sc}$" and "-L_{sc}", we conclude that considering the semantic clustering information of label embeddings through the super-class clustering loss can improve performance.
- With all methods, the mAP scores worsen as the value of the IoU threshold α increases. This is reasonable because a larger IoU threshold tends to require a more accurate boundary for the predicted activities.

Results on Charades: Table 3 reports the results for the 37 unseen activities with the Charades data in terms of Sigurdsson *et al.*'s [34] standard and post-processed mAP evaluation metrics. The results are very consistent with those from the THUMOS'14 dataset. That is, ($+TPN_*$+L_{sc}) still delivers the best activity detection results with a 2.78% and
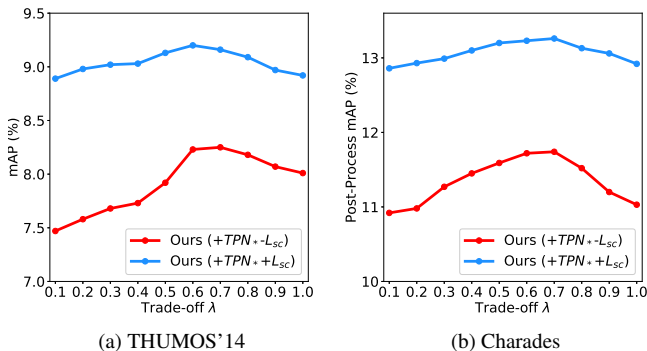

Figure 4: The influence of improved classification module (trade-off λ) in TPN subnet for ZSTAD.

2.24% performance improvement in standard mAP compared to R-C3D+SE and R-C3D+ConSE, respectively. We also note that the manually-designed post-processing effectively improves the performance in terms of mAP. In addition, Table 4 shows the standard AP values per-unseen activity class with the method ($+TPN_*$+L_{sc}). As the table shows, these unseen activities, such as "Sitting in a chair", "Someone is dressing", and "Sitting at a table", are relatively easy to detect in the testing stage.

4.4. Impact of the Improved Classification Module in the TPN Subnet

The purpose of the improved classification module in the TPN subnet is to refine the results of previous binary classification as to whether the segment contains an activity or is the background. It does this by introducing the label embeddings of background and weighting the output with a trade-off hyper-parameter λ. Therefore, it is necessary to test the influence of λ on the results. The variants ($+TPN_*$-L_{sc}) and ($+TPN_*$+L_{sc}), show the effect of this parameter with λ tuned from 0.1 to 1 in step of 0.1. Figure 4 shows both the mAP curves at an IoU threshold of $\alpha = 0.5$ for THUMOS'14, and the post-process mAP curves for Charades. We make two observations from the results:

- ($+TPN_*$-L_{sc}) and ($+TPN_*$+L_{sc}) consistently perform better than two baselines R-C3D+SE and R-

Table 4: Per-unseen class AP (%) on Charades dataset with our method ($+TPN_*+L_{sc}$).

Throwing clothes	10.80	Opening a door	11.53	Sitting at a table	16.44	Talking on a phone	5.28
Holding a bag	7.86	Taking a book	3.93	Reading at a book	11.66	Holding a towel/s	12.87
Taking from a box	3.58	Closing a box	4.08	Taking a laptop	3.45	Tidying up a blanket	5.93
Sitting in a chair	18.09	Putting food somewhere	10.94	Eating a sandwich	7.96	Taking shoes	10.88
Holding a pillow	7.91	Tidying a shelf	4.84	looking at a picture	5.64	Closing a window	3.67
Taking a broom	10.35	Holding a mirror	2.69	Turning off a light	4.97	Washing a cup	4.05
Opening a closet	7.54	Taking paper	4.11	Wash a dish	9.59	Sitting on sofa	14.41
Tidying on the floor	8.14	Holding medicine	5.04	Taking a vacuum	5.63	Lying on a bed	10.10
Watching television	11.12	Fixing a doorknob	2.87	Opening a refrigerator	4.50	Someone is eating	5.32
Someone is dressing	14.90						

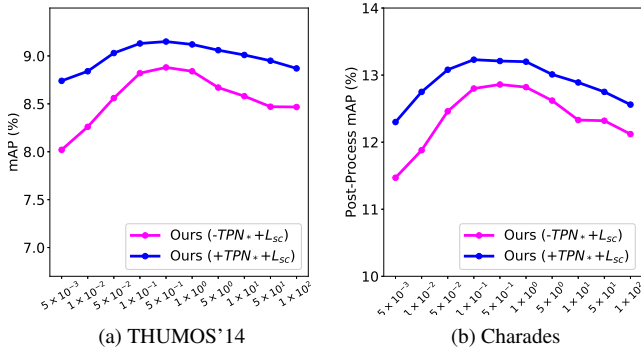(a) THUMOS'14   (b) Charades

Figure 5: The influence of super-class clustering loss (trade-off β) in ZSDN subnet for ZSTAD.

C3D+ConSE (see Tables 1 and 3) regardless of any improvement brought by the trade-off λ.

- As λ increases, performance improves to a point then gradually tapered. For THUMOS'14, the optimal value of λ is 0.6, and 0.7 for Charades.

Overall, we find that the improved classification module with a properly tuned λ enhances performance by generating better proposals for unseen activity classes in the testing stage, and the optimal value of λ will differ depending on the specific properties of the dataset.

4.5. Impact of the Super-Class Clustering Loss in the ZSDN Subnet

As part of the ZSDN subnet, the super-class clustering term L_{sc} in Eq. (2) aims to capture the common semantics among all activities from the prior information of super-classes over all activity label embeddings. Apparently, the influence of the trade-off parameter β in the classification loss needs to be examined. The variants ($-TPN_*+L_{sc}$) and ($+TPN_*+L_{sc}$) are applicable here, with β varied in interval $\{5 \times 10^{-3}, 1 \times 10^{-2}, 5 \times 10^{-2}, 1 \times 10^{-1}, 5 \times 10^{-1}, 1 \times 10^0, 5 \times 10^0, 1 \times 10^1, 5 \times 10^1, 1 \times 10^2\}$ for both datasets.

Figure 5 shows the mAP curves at an IoU threshold of 0.5 for the THUMOS'14 dataset along with the post-process

mAP curves for Charades. As the figures show, detection performance for all methods initially increases with an increase in β before stabilizing in the interval $[10^{-1}; 10^0]$. Once β exceeds 10^0, performance gradually declines. In addition, we note that the results for ($-TPN_*+L_{sc}$) are notably better than for ($-TPN_*-L_{sc}$) regardless of the hyper-parameter β, as Tables 1 and 3 show (*i.e.*, a 7.37% improvement with THUMOS'14 and a 10.89% improvement with Charades). Similarly, no matter the value of β, ($+TPN_*+L_{sc}$) consistently outperforms ($+TPN_*-L_{sc}$) on the two datasets. These results indicate that super-classes over activity label embeddings are conducive to mine the correlations between seen and unseen activities and significantly contribute to detecting unseen activities.

5. Conclusion

In this paper, we propose a novel problem setting for temporal activity detection in which activities that are not seen during the training stage can be recognized and localized simultaneously. The solution presents in this paper is the first attempt at zero-shot temporal activity detection (ZSTAD). To address this challenging problem, we design an end-to-end deep network that uses label embeddings and their super-classes as prior information to capture the semantics common to seen and unseen activities. The results on THUMOS'14 and Charades datasets show the approach is able to detect unseen activity to a high degree of accuracy.

References

[1] Ankan Bansal, Karan Sikka, Gaurav Sharma, Rama Chellappa, and Ajay Divakaran. Zero-shot object detection. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018.

[2] Xiaojun Chang, Zhigang Ma, Yi Yang, Zhiqiang Zeng, and Alexander G. Hauptmann. Bi-level semantic representation analysis for multimedia event detection. *IEEE Trans. Cybernetics*, 47(5):1180–1197, 2017.

[3] Xiaojun Chang, Yi Yang, Alexander G. Hauptmann, Eric P. Xing, and Yaoliang Yu. Semantic concept discovery for large-scale zero-shot event detection. In *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, July 25-31, 2015*, pages 2234–2240, 2015.

[4] Xiaojun Chang, Yi Yang, Guodong Long, Chengqi Zhang, and Alexander G. Hauptmann. Dynamic concept composition for zero-example event detection. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12-17, 2016, Phoenix, Arizona, USA*, pages 3464–3470, 2016.

[5] Xiaojun Chang, Yaoliang Yu, Yi Yang, and Eric P. Xing. They are not equally reliable: Semantic event search using differentiated concept classifiers. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 1884–1893, 2016.

[6] Xiaojun Chang, Yaoliang Yu, Yi Yang, and Eric P. Xing. Semantic pooling for complex event analysis in untrimmed videos. *IEEE Trans. Pattern Anal. Mach. Intell.*, 39(8):1617–1632, 2017.

[7] Berkan Demirel, Ramazan Gokberk Cinbis, and Nazli Ikizler-Cinbis. Zero-shot object detection by hybrid region embedding. In *Proceedings of the British Machine Vision Conference (BMVC)*, 2018.

[8] Adrien Gaidon, Zaid Harchaoui, and Cordelia Schmid. Temporal localization of actions with actoms. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, page 1, 2013.

[9] Jiyang Gao, Zhenheng Yang, and Ram Nevatia. Cascaded boundary regression for temporal action detection. *Proceedings of the British Machine Vision Conference (BMVC)*, 2017.

[10] Jiyang Gao, Zhenheng Yang, Chen Sun, Kan Chen, and Ram Nevatia. Turn tap: Temporal unit regression network for temporal action proposals. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017.

[11] Yoav Goldberg and Omer Levy. word2vec explained: deriving mikolov et al.'s negative-sampling word-embedding method. *arXiv preprint arXiv:1402.3722*, 2014.

[12] Jingjia Huang, Nannan Li, Tao Zhang, Ge Li, Tiejun Huang, and Wen Gao. Sap: Self-adaptive proposal model for temporal action detection based on reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2018.

[13] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. Caffe: Convolutional architecture for fast feature embedding. In *Proceedings of the ACM international Conference on Multimedia (ACMMM)*, pages 675–678, 2014.

[14] Y.-G. Jiang, J. Liu, A. Roshan Zamir, G. Toderici, I. Laptev, M. Shah, and R. Sukthankar. THUMOS challenge: Action recognition with a large number of classes. `http://crcv.ucf.edu/THUMOS14/`, 2014.

[15] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. Large-scale video classification with convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[16] Christoph H Lampert, Hannes Nickisch, and Stefan Harmeling. Attribute-based classification for zero-shot visual object categorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 36(3):453–465, 2014.

[17] Jingjing Li, Mengmeng Jing, Ke Lu, Zhengming Ding, Lei Zhu, and Zi Huang. Leveraging the invariant side of generative zero-shot learning. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 7402–7411, 2019.

[18] Zhihui Li, Lina Yao, Xiaojun Chang, Kun Zhan, Jiande Sun, and Huaxiang Zhang. Zero-shot event detection via event-adaptive concept relevance mining. *Pattern Recognit.*, 88:595–603, 2019.

[19] Changzhi Luo, Zhetao Li, Kaizhu Huang, Jiashi Feng, and Meng Wang. Zero-shot learning via attribute regression and class prototype rectification. *IEEE Transactions on Image Processing (TIP)*, 27(2):637–648, 2017.

[20] Min Meng and Jun Yu. Zero-shot learning via robust latent representation and manifold regularization. *IEEE Transactions on Image Processing (TIP)*, 28(4):1824–1836, 2019.

[21] Li Niu, Jianfei Cai, Ashok Veeraraghavan, and Liqing Zhang. Zero-shot learning via category-specific visual-semantic mapping and label refinement. *IEEE Transactions on Image Processing (TIP)*, 28(2):965–979, 2019.

[22] Mohammad Norouzi, Tomas Mikolov, Samy Bengio, Yoram Singer, Jonathon Shlens, Andrea Frome, Greg S Corrado, and Jeffrey Dean. Zero-shot learning by convex combination of semantic embeddings. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2014.

[23] Mark Palatucci, Dean Pomerleau, Geoffrey E Hinton, and Tom M Mitchell. Zero-shot learning with semantic output codes. In *Proceedings of the Advances in Neural Information Processing Systems (NIPS)*, pages 1410–1418, 2009.

[24] Devi Parikh and Kristen Grauman. Relative attributes. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 503–510, 2011.

[25] Sujoy Paul, Sourya Roy, and Amit K Roy-Chowdhury. W-talc: Weakly-supervised temporal activity localization and classification. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 563–579, 2018.

[26] Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove: Global vectors for word representation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, 2014.

[27] Jie Qin, Li Liu, Ling Shao, Fumin Shen, Bingbing Ni, Jiaxin Chen, and Yunhong Wang. Zero-shot action recognition with error-correcting output codes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2833–2842, 2017.

[28] Shafin Rahman, Salman Khan, and Fatih Porikli. A unified approach for conventional zero-shot, generalized zero-shot, and few-shot learning. *IEEE Transactions on Image Processing (TIP)*, 27(11):5652–5667, 2018.

[29] Shafin Rahman, Salman Khan, and Fatih Porikli. Zero-shot object detection: learning to simultaneously recognize and localize novel concepts. In *Proceedings of the Asian Conference on Computer Vision (ACCV)*, 2018.

[30] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Proceedings of the Advances in Neural Information Processing Systems (NIPS)*, pages 91–99, 2015.

[31] Zheng Shou, Jonathan Chan, Alireza Zareian, Kazuyuki Miyazawa, and Shih-Fu Chang. Cdc: Convolutional-de-convolutional networks for precise temporal action localization in untrimmed videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[32] Zheng Shou, Hang Gao, Lei Zhang, Kazuyuki Miyazawa, and Shih-Fu Chang. Autoloc: Weakly-supervised temporal action localization in untrimmed videos. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 154–171, 2018.

[33] Zheng Shou, Dongang Wang, and Shih-Fu Chang. Temporal action localization in untrimmed videos via multi-stage cnns. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1049–1058, 2016.

[34] Gunnar A Sigurdsson, Santosh Divvala, Ali Farhadi, and Abhinav Gupta. Asynchronous temporal fields for action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 585–594, 2017.

[35] Gunnar A. Sigurdsson, Gül Varol, Xiaolong Wang, Ali Farhadi, Ivan Laptev, and Abhinav Gupta. Hollywood in homes: Crowdsourcing data collection for activity understanding. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2016.

[36] Richard Socher, Milind Ganjoo, Christopher D Manning, and Andrew Ng. Zero-shot learning through cross-modal transfer. In *Proceedings of the Advances in Neural Information Processing Systems (NIPS)*, pages 935–943, 2013.

[37] 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 (ICCV)*, pages 4489–4497, 2015.

[38] Limin Wang, Yu Qiao, and Xiaoou Tang. Action recognition and detection by combining motion and appearance features. *THUMOS14 Action Recognition Challenge*, 1(2):2, 2014.

[39] Limin Wang, Yuanjun Xiong, Dahua Lin, and Luc Van Gool. Untrimmednets for weakly supervised action recognition and detection. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4325–4334, 2017.

[40] Yuanjun Xiong, Yue Zhao, Limin Wang, Dahua Lin, and Xiaoou Tang. A pursuit of temporal accuracy in general activity detection. *arXiv preprint arXiv:1703.02716*, 2017.

[41] Huijuan Xu, Abir Das, and Kate Saenko. R-c3d: region convolutional 3d network for temporal activity detection. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 5794–5803, 2017.

[42] Huijuan Xu, Bingyi Kang, Ximeng Sun, Jiashi Feng, Kate Saenko, and Trevor Darrell. Similarity r-c3d for few-shot temporal activity detection. *arXiv preprint arXiv:1812.10000*, 2018.

[43] Xun Xu, Timothy Hospedales, and Shaogang Gong. Semantic embedding space for zero-shot action recognition. In *IEEE International Conference on Image Processing (ICIP)*, pages 63–67. IEEE, 2015.

[44] Lihi Zelnik-Manor and Pietro Perona. Self-tuning spectral clustering. In *Proceedings of the Advances in neural information processing systems (NIPS)*, pages 1601–1608, 2005.

[45] Haofeng Zhang, Yang Long, Yu Guan, and Ling Shao. Triple verification network for generalized zero-shot learning. *IEEE Transactions on Image Processing (TIP)*, 28(1):506–517, 2019.

[46] Lingling Zhang, Jun Liu, Minnan Luo, Xiaojun Chang, and Qinghua Zheng. Deep semisupervised zero-shot learning with maximum mean discrepancy. *Neural Computation*, 30(5), 2018.

[47] Ziming Zhang and Venkatesh Saligrama. Zero-shot learning via semantic similarity embedding. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 4166–4174, 2015.

[48] Yue Zhao, Yuanjun Xiong, Limin Wang, Zhirong Wu, Xiaoou Tang, and Dahua Lin. Temporal action detection with structured segment networks. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, volume 2, 2017.

[49] Pengkai Zhu, Hanxiao Wang, Tolga Bolukbasi, and Venkatesh Saligrama. Zero-shot detection. *IEEE Transaction on Circuits and Systems for Video Technology (TCSVT)*, 2019.

[50] Yi Zhu, Yang Long, Yu Guan, Shawn Newsam, and Ling Shao. Towards universal representation for unseen action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9436–9445, 2018.