Int J Comput Vis (2016) 120:61–77
DOI 10.1007/s11263-016-0893-6

Recognizing an Action Using Its Name: A Knowledge-Based Approach

Chuang Gan[1] · Yi Yang[2] · Linchao Zhu[2] · Deli Zhao[3] · Yueting Zhuang[4]

Received: 7 August 2014 / Accepted: 15 February 2016 / Published online: 2 March 2016

Abstract Existing action recognition algorithms require a set of positive exemplars to train a classifier for each action. However, the amount of action classes is very large and the users' queries vary dramatically. It is impractical to predefine all possible action classes beforehand. To address this issue, we propose to perform action recognition with no positive exemplars, which is often known as the zero-shot learning. Current zero-shot learning paradigms usually train a series of attribute classifiers and then recognize the target actions based on the attribute representation. To ensure the maximum coverage of ad-hoc action classes, the attribute-based approaches require large numbers of reliable and accurate attribute classifiers, which are often unavailable in the real world. In this paper, we propose an approach that merely takes an action name as the input to recognize the action of interest without any pre-trained attribute classifiers and positive exemplars. Given an action name, we first build an analogy pool according to an external ontology, and each action in the analogy pool is related to the target action at different levels. The correlation information inferred from the external ontology may be noisy. We then propose an algorithm, namely adaptive multi-model rank-preserving mapping (AMRM), to train a classifier for action recognition, which is able to evaluate the relatedness of each video in the analogy pool adaptively. As multiple mapping models are employed, our algorithm has better capability to bridge the gap between visual features and the semantic information inferred from the ontology. Extensive experiments demonstrate that our method achieves the promising performance for action recognition only using action names, while no attributes and positive exemplars are available.

Keywords Action recognition · Semantic correlation · Adaptive multi-model rank-preserving mapping (AMRM)

Communicated by Deva Ramanan.

✉ Yi Yang
yee.i.yang@gmail.com

Chuang Gan
ganchuang1990@gmail.com

Linchao Zhu
zhulinchao7@gmail.com

Deli Zhao
zhaodeli@gmail.com

Yueting Zhuang
yzhuang@zju.edu.cn

[1] IIIS, Tsinghua University, Beijing, China

[2] QCIS, University of Technology, Sydney, Australia

[3] HTC Research, Beijing, China

[4] Zhejiang University, Hangzhou, China

1 Introduction

Video collections on the Web contain a multitude of actions and events. Current work in computer vision and multimedia has explored the problem of action recognition in the real world videos and made significant progress over the last decade. In literature, reliable low-level features such as STIP (Laptev et al. 2008), MoSIFT (Chen and Hauptmann 2009), dense trajectory (Wang et al. 2011) and improved dense trajectory (Wang and Schmid 2013), combined with a modern machine learning algorithm such as Support Vector Machines (SVMs), have achieved promising recognition results.

To obtain good performances in action recognition, existing approaches require sufficient positive exemplars to train a series of action classifiers. However, due to the large number of action classes, it is difficult to obtain adequate positive

exemplars that exactly match a target action. Fortunately, recent progress shed light on circumventing the challenge of reducing human labor for supervision. Wang et al. (2012) developed a semi-supervised learning algorithm, aiming to improve the action recognition performance when the number of positive exemplars is few. In Ma et al. (2014), proposed to adapt the knowledge of clean, lab generated action data to recognize the action in the real world videos. Liu et al. (2011) proposed to recognize actions by a piece of well-structured attribute lists, which is probably the first attempt to recognize actions only using texts. To recognize an action, however, users have to indicate whether each of the attributes is positive, which involves tedious human interactions (Liu et al. 2011). For example, to recognize the action *walking*, they need to define that the attributes such as *arm pendulum-like motion* and *translation motion* are positive, but the attributes like *torso up-down motion* and *torso twist* are negative. In addition, the description template designed in Liu et al. (2011) is static, making it difficult to scale up to a variety of ad-hoc actions. Therefore, in the experiments of Liu et al. (2011), action classes are restricted to a few simple ones such as *walk* and *jump forward*, in the clean and lab-generated video datasets such as the KTH dataset (Laptev et al. 2008) and the Weizmann dataset (Blank et al. 2005). It remains unclear how to recognize the complex actions such as *soccer penalty*, with a limited number of pre-specified attributes.

Pattern recognition based on visual attributes (Farhadi et al. 2009; Lampert et al. 2009; Rohrbach et al. 2010; Akata et al. 2013; Yu et al. 2013; Ma et al. 2013; Cai et al. 2012; Wang et al. 2012) has received much attention in the computer vision and multimedia fields over the past decade. The term "attribute" often refers to human-nameable properties that are shared across different classes. As discussed in object categorization (Farhadi et al. 2009; Lampert et al. 2009), action recognition (Liu et al. 2011) and multimedia event recognition (Liu et al. 2013), the ability of characterizing objects and actions by attributes is not only helpful for recognizing available objects, actions and events, but also powerful for recognizing classes that have never been seen before, for which no positive exemplars are available. This problem is also called zero-shot learning.

One main challenge of attribute-based zero-shot learning arises from lack of well-defined reliable attribute classifiers. Previous research (Ma et al. 2013) has revealed the inherent uncertainty in terms of the accuracy and reliability of attribute representation. As action recognition directly relies on the attribute representation, the performance will degrade if attribute classifiers are inaccurately trained. In addition, to ensure the maximum coverage of action types, we need to build a large number of attribute classifiers, which in turn requires a lot of labeling efforts. Moreover, it remains unclear how many attributes will be sufficient and what kinds of attributes will be particularly suitable for an unknown action. Thus, it is a non-trivial task of designing a static attribute pool for different actions, because actions are dynamic and diverse.

After carefully analyzing different classes of actions, we find that a series of action classes may share some elements if they are semantically similar to each other. Instead of explicitly modeling the shared information by using attribute representation, we propose to learn a series of mapping functions which preserve the semantic correlation between an unseen action class and the known training classes. In this paper, we propose to leverage the knowledge from an external ontology, i.e., *wikipedia*, and train a series of rank-preserving mapping functions for action recognition. As the example illustrated in Fig. 1, if we ask to retrieve videos that are highly related to *field hockey penalty*, related to *soccer juggle*, but unrelated to

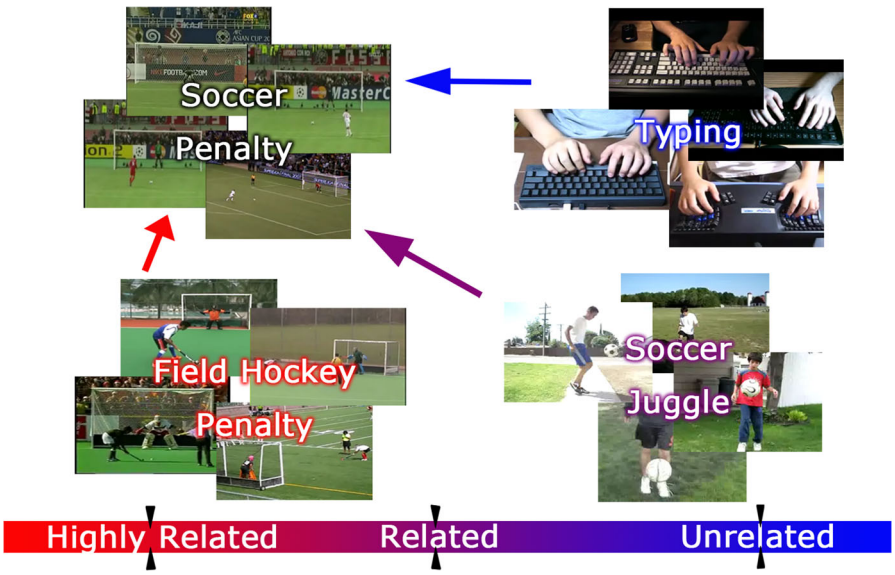

Fig. 1 An example of recognizing the target action *soccer penalty*. The action of *soccer juggle* and *field hockey penalty* are related to *soccer penalty* at different levels

basketball, it is very likely that the videos of *soccer penalty* will be returned before the videos of *volleyball spiking* and even *field hockey penalty*. Compared to the attribute-based approaches, our method does not depend on the attribute-based representation that require additional annotations but still unreliable to obtain. The semantic correlations mined from the external knowledge base may contain certain noise mainly because some words in action names may occur less frequently in *wikipedia*, such as *yo yo*, *dunk*, *salsa* and so on. In addition, the real world videos may have dramatic variations even though they have the same semantic labels. To relieve the noise, we propose to adaptively adjust the ranking scores on a per video basis. Inspired by Carreira et al. (2012), the feature of each video is represented via second-order pooling, which can preserve the spatial structures. Instead of converting the matrix feature into a high-dimensional vector, we directly map the high-order matrix representation to the action classes for efficient recognition. To enhance the flexibility of the mapping function, multiple models are trained simultaneously. Our work makes the following contributions:

- We propose a principled AMRM framework for zero-shot action recognition, which is flexible, efficient, and able to adaptively utilize the training exemplars on a per video basis. If available, positive samples could also be added in the proposed framework to further improve the recognition performance.
- Our algorithm does not require pre-trained attribute classifiers that are often noisy and unreliable, thereby avoiding the tedious human interactions of defining reliable attributes and labeling high-quality attribute exemplars.
- We conduct extensive experiments to demonstrate the effectiveness of our approach for the unseen action recognition and achieve promising results on the large-scale UCF101 and TRECVID MED 2011 datasets.

The rest of the paper is organized as follows. In Sect. 2, we review the related work of zero-shot learning and action recognition. In Sect. 3, we formulate our adaptive multi-model rank-preserving mapping (AMRM) approach in detail. Experimental settings and evaluation results are presented in Sect. 4. Section 5 concludes the paper.

2 Related Work

Our framework involves two research directions: zero-shot learning and action recognition, which will be presented in this section, respectively.

2.1 Zero-Shot Learning

The task of zero-shot learning is to recognize classes that have never been seen before. Namely, there are no positive exemplars available. Thus it requires the ability of transferring knowledge from classes that we have training data to classes that no training data are available. A popular solution to zero-shot learning is to embed an intermediate layer, referred as attribute, in the algorithmic architecture. Most recent methods harvest the attributes by manual labelling (Farhadi et al. 2009; Lampert et al. 2009; Yu et al. 2013) mining knowledge from other domains (Rohrbach et al. 2010), or extracting the features themselves (Yu et al. 2013; Liu et al. 2011). After obtaining attributes, the effectiveness of knowledge transferring always depends on the performances of trained classifiers independently (Lampert et al. 2009) or the mapping function between low-level features and attribute labels (Akata et al. 2013).

To alleviate the burden of annotating attributes, a first line of research considered to treat the training object/action classes as a special kind of attributes, and directly transfer them to the unseen classes. Given an unseen object/action class, these approaches firstly identify the related classes from the available classes by knowledge transfer from class hierarchical relationship, *wikipedia*, or web image search engine, as described in Torresani et al. (2010), Liu et al. (2013), Hauptmann et al. (2007), Rohrbach et al. (2011), and Kankuekul et al. (2012). And then they train their classifiers individually and combine their scores on the testing data to justify their fit to the unseen class. For example, if we want to recognize an unseen action *front crawl swimming*, which is related to actions *breaststroke* and *crawl*, one would run two classifiers and combine them (*breaststroke* & *crawl*) to indirectly get a classifier for recognizing the unseen classes. However, this may not be the most effective or efficient solution. Conjunctions of actions *breaststroke* and *crawl* may have a very characteristic appearance, and combine these two classes together to train one classifier should result in more accurate and faster recognition results for the unseen action class. However, it is difficult to define all possible combinations and weighting schemes for different action classes beforehand, which serves as the motivation of our work. A second line of research tried to learn a visual-semantic embedding function (Socher et al. 2013; Frome et al. 2013) for the zero-shot object recognition. However, it remains unclear how to extend these framework to conduct zero-shot video activity recognition, mainly due to the complexity and diversity of video data.

Our idea of zero-shot action recognition is also related to sentence generation task (Guadarrama et al. 2013; Sun et al. 2015). Given a video, they try to provide a title or description to describe it. However, the goal of our paper is different, as we deal with the opposite direction of ranking videos that match the query.

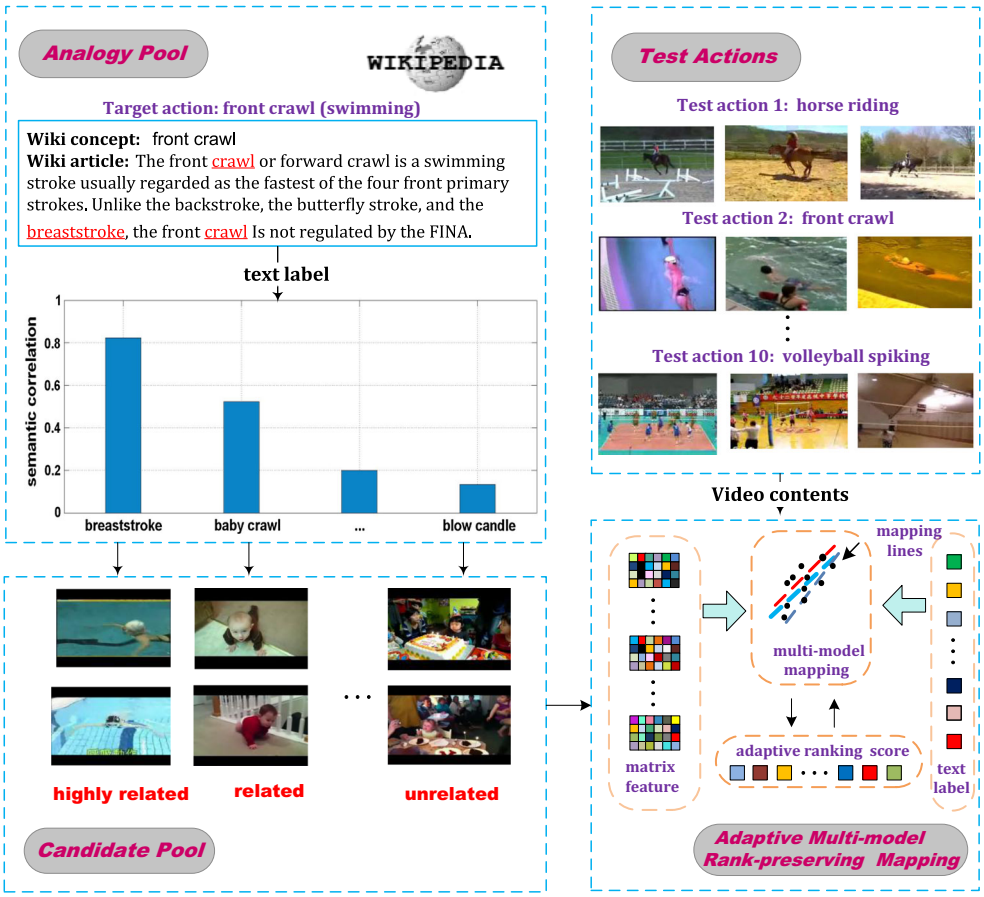

Fig. 2 The pipeline of our approach. Using action names, we first build an analogy pool, consisting a series of videos which are semantically similar to the target action, and then apply the adaptive multi-model rank-preserving mapping (AMRM) model to train a classifier for the target action recognition

2.2 Action Recognition

The problem of action recognition has been widely explored in the community of computer vision and multimedia. A detailed survey can be found in Jiang et al. (2013). Recently, researches focus on realistic datasets collected from movies (Laptev et al. 2008) and web videos (Reddy and Shah 2013). UCF101 (Soomro et al. 2012) is a large-scale action recognition dataset and has driven more difficult action recognition. Most successful approaches are based on some local space-time forms of features that are then represented by bag-of-word (BOW) histograms or fisher vector (FV). They are finally fed into a SVM classifier to train specific action recognition models. In Wang et al. (2013), Wang et al. proposed to use dense trajectories and motion boundary descriptors for action recognition, which achieved good performances on a variety of datasets. In addition, several mid-level representations also draw attention in action recognition. Action bank (Sadanand and Corso 2012) has been proposed as a new mid-level feature based on atomic action. Besides, several works proposed attributes as the mid-level representation and also applied it to zero-shot action recognition (Laptev 2005; Rohrbach et al. 2012; Fu et al. 2014, 2015), and few-shot (Rohrbach et al. 2013) action recognition. However, they relied on manually-defined and data-driven attributes, so it is not applicable for large-scale setting. Recently, in order to improve action recognition performances when the number of positive exemplars is few, Wang et al. (2012) proposed a semi-supervised learning approach, and then Yang et al. (2014) proposed a semi-supervised active learning approach. Duan et al. (2012) proposed a domain adaptation approach by leveraging loosely labeled videos. However, these approaches still require positive exemplars.

3 Action Recognition by Class Names

The framework of our approach is shown in Fig. 2. We first infer a series of ranking scores, one for each action class in the analogy pool, by leveraging external knowledge. A higher ranking score indicates that the corresponding video is more closely related to the target action. In particular, we use the ontology *wikipedia* as a knowledge base to achieve this goal.

Once the ranking scores of the analogy videos are obtained, we then propose to train a function to map the visual features to the ranking scores. The ranking preserving mapping function is then used as a classifier for action recognition. The ranking scores directly inferred from *wikipedia* is noisy. To ameliorate this impediment, we propose to adjust the ranking scores adaptively. Multiple models are used to constitute the mapping function to have a larger space for optimization.

3.1 Knowledge-Based Ranking

In order to infer the relation between source video classes and the target action, we expect to merely use action names to build an analogy pool. Our solution is to leverage the external ontology, such as *wikipedia* and *WordNet*, to calculate semantic correlations between action names, according to which a ranking score is assigned to each video in the analogy pool. However, mining semantic relationship from *WordNet* between verbs (actions) is more difficult than discovering relationship between nouns (objects), as verbs do not have the same well-built ontological relationship found with nouns. Therefore, we explore the semantic relationship between action class names through *wikipedia*, since *wikipedia* is the largest online collaboratively built encyclopedia with more than three million articles for English version. It contains pages for concepts and each page provides a detailed and human-edited descriptions of the corresponding concept.

We apply the Explicit Semantic Analysis (ESA) measure (Gabrilovich and Markovitch 2007) to measure semantic correlations between action names. Each *wikipedia* concept is represented as a vector of frequencies that words occur in the corresponding article. Entries of these vectors are assigned with values using the $tf\text{-}idf$ scheme. Then we build an inverted index, which maps each word into a list of concepts in which it appears. Given a text fragment, the semantic interpreter iterates over the text words, retrieves corresponding entries from the inverted index, and merges them into a weighted vector that represents the given text. Let $G = \{d_1, d_2, \ldots, d_n\}$ be the input texts, such as action names. Each is represented as $tf\text{-}idf$ vectors. Denote w_i as the weight of word d_i, and $C = \{c_1, \ldots, c_N\}$ as whole concepts, where N is the total number of *wikipedia* concepts. $m_j \in R^{N\times 1}$ is an inverted index entry for word d_i, and m_j quantifies the strength of correlation of word d_i with *wikipedia* concepts C. Then, the semantic interpretation vector $v \in R^{N\times 1}$ for text G is defined as

$$v = \sum_{d_i \in G} w_i \cdot m_j. \tag{1}$$

Entries of this vector reflect the relevance of the corresponding concepts to text G. To obtain semantic correlation of a pair of text fragments, we compute cosine similarities between their vectors. Taking a training action name i and a testing action name j for an example, we firstly do stemming and lemmatization for these two names, and then represent them as two concept vectors $v_i \in R^{N\times 1}$ and $v_j \in R^{N\times 1}$. Finally we compute cosine distance between two vectors as the text label of training sample i, denoted as

$$y_i = \frac{{v_i}^T \cdot v_j}{||v_i|| \cdot ||v_j||}, \tag{2}$$

where $||\cdot||$ denotes the norm of a vector. In this way, we can figure out whether each action class in the training set is related to the target action or not, and then form the well-built analogy pool that consists of related action names and candidate pool (corresponding videos). The framework is illustrated in Fig. 3.

3.2 Second-Order Pooling

As reported in Carreira et al. (2012), the second-order pooling method has not only obtained better performances but also

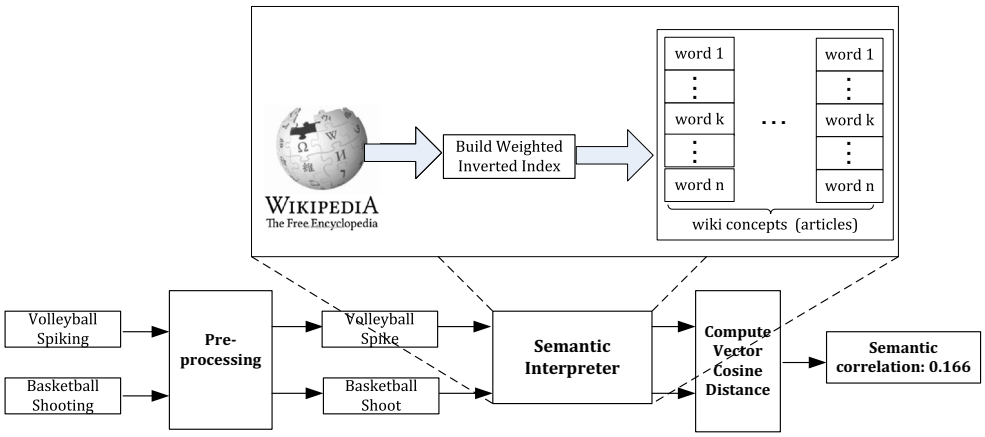

Fig. 3 The framework of computing semantic correlation between two action names

saved more computational efforts than other feature coding approaches, such as bag-of-words (BOW) and Fisher Vector (FV) (Sánchez et al. 2013; Oneata et al. 2013), we adopt it in our paper.

Denote the collection of features for a video i as $X_i \in R^{d\times n_i}$, where d is the dimension of local features and n_i is the number of local features for the video i. Following Carreira et al. (2012), we pool the local features to form the global features as follows

$$G_i = \log(X_i \cdot {X_i}^T). \tag{3}$$

After the second-order pooling, each video is represented by a feature matrix $G_i \in R^{d\times d}$.

3.3 Adaptive Multi-model Rank-preserving Mapping (AMRM)

After selecting the related classes in the training set, we employ an adaptive multi-model rank-preserving mapping (AMRM) framework to train a classifier for the target action recognition.

Different from Carreira et al. (2012), which converts the second-order matrix representation to high-dimension vectors followed by a SVM classifier, we propose to directly map the second-order representations to the semantic space. The reasons lie in that vectorization suffers from some drawbacks. Firstly, the spatial relationship of images and the temporal relationship of frames may be destroyed. Secondly, it causes the higher dimensionality and increases the computational complexity.

To begin with, we denote $X = [X_1, \ldots, X_n] \in R^{p\times q\times n}$ as the training set, where $X_i \in R^{p\times q}$ is the matrix feature for i-th videos and n is the number of the training video. Let $Y = [y_1, \ldots, y_n] \in R^{1\times n}$ be the ranking scores, where $y_i \in [0, 1]$. One direct way to replace the traditional vector-based projection is to introduce a tensor counterpart, *e.g.* $r_j^T X_i s_j$, where $r_j \in R^{p\times 1}$ and $s_j \in R^{q\times 1}$ are the left and right projection vectors. As $p+q$ is much smaller than $p\times q$, using the 2D representations for action recognition is much faster at the prediction stage than converting it to a vector representation.

Using a single model could be too restrictive. For example, p and q value in a tensor base rs^T only have $p+q$ degrees of freedom. In practice, it may increase the regression errors. To handle that, we introduce m couples of projection vectors. They are denoted as $\{r_j\}_{j=1}^m$ and $\{s_j\}_{j=1}^m$. This is different from the traditional single mapping model as several mapping models are integrated. In this way, these mapping models work collaboratively in the learning process, which provides us with larger space to search the optimal solutions. In a consequence, the recognition performance can be effectively enhanced. The multi-model mapping (MM) can be formulated as

$$\min_{r_j,s_j} \sum_{i=1}^{n} \left(\sum_{j=1}^{m} r_j^T X_i s_j - y_i \right)^2 + \mu \sum_{j=1}^{m} ||r_j s_j^T||_{\text{Fro}}^2, \tag{4}$$

where $||\cdot||_{\text{Fro}}$ denotes the Frobenius norm of a matrix. In our setting, the value of y_i is computed from text sources, and the larger value of y_i means that the i-th training data is more related to the target class. However, as analyzed in the previous section, the labels extracted from external ontology may contain noise. In order to better differentiate video classes, we use a vector $A = [a_1, a_2, \ldots, a_n] \in R^{1\times n}$ to indicate the degree of relatedness between the training data and the target class. If X_i belongs to the highly related action class, $a_i = 1$; If X_i belongs to the related action class, $a_i = 0$; If X_i belongs to the unrelated class, $a_i = -1$.

In order to adaptively infer a ranking score for each source video, we also introduce a new non-negative variables $E = [e_1, \ldots, e_n] \in R^{1\times n}$ to be simultaneously optimized. Thus the ranking scores are $\widehat{Y} = Y + A \odot E$, where $\odot$ is Hadamard product, *i.e.* the entry-wise product. Therefore, given a highly related exemplar X_i, its adaptive ranking score should be $\widehat{y_i} = y_i + e_i$. If X_i is a related action exemplar, its adaptive ranking score should be the same as y_i; And if X_i is an unrelated exemplar, its adaptive ranking score should be $\widehat{y_i} = y_i - e_i$. Moreover, we also introduce the variable $F = [f_1, \ldots, f_n] \in R^{1\times n}$ to relax the label constraint in order to handle the noises in semantic correlation. Then we learn the parameters $\{r_j\}_{j=1}^m$ and $\{s_j\}_{j=1}^m$ by solving the following optimization

$$\begin{aligned} \min_{r_j,s_j,F,E} & \sum_{i=1}^{n} \left(\sum_{j=1}^{m} r_j^T X_i s_j - (f_i + a_i \odot e_i) \right)^2 \\ & + \lambda (F-Y)(F-Y)^T + \mu \sum_{j=1}^{m} ||r_j s_j^T||_{\text{Fro}}^2, \\ \text{s.t.} \quad & E > 0, \end{aligned} \tag{5}$$

where the first term minimizes the empirical error loss between the projection of the low-level features and the adaptive ranking scores, and the second term is the distances between the ranking scores and the text labels obtained from text resources. The last term is the regularization on r_j and s_j. μ and λ are the tradeoff parameters. As the ranking scores $F + A \odot E$ are parameters in the optimization, the model can utilize the related exemplars on per exemplar basis. Taking a training sample X_i for an example, its action class is the highly-related video class, but itself is less related to the target action. Then a smaller e_i will be added from f_i to reduce the least regression errors. Finally, the learned projections $\{r_j\}_{j=1}^m$ and $\{s_j\}_{j=1}^m$ can be used to compute the recognition scores of testing data. The detailed optimization method is presented in the following section.

3.4 Optimization

We apply the iterative approach to solve the objective function in (5)

(1) **Fix** s_j and **optimize** r_j and F.

Denote $b_i^j = X_i s_j$ and $D_s = \text{diag}(s_1^T s_1 I_p, s_2^T s_2 I_p, \ldots, s_m^T s_m I_p)$, where diag$(\cdot)$ denotes the diagonal matrix and $I_p \in R^{p\times p}$ is an identity matrix. The formulation can be rewritten as

$$\min_{r_j,F,E} \sum_{i=1}^{n} \left((r_1^T\ r_2^T\ \cdots\ r_m^T) \begin{pmatrix} b_i^1 \\ b_i^2 \\ \cdots \\ b_i^m \end{pmatrix} - f_i - a_i \odot e_i \right)^2 + \lambda(F-Y)(F-Y)^T + \mu(r_1^T\ r_2^T\ \cdots\ r_m^T) D_s \begin{pmatrix} r_1^T \\ r_2^T \\ \cdots \\ r_m^T \end{pmatrix}. \tag{6}$$

Let $R = (r_1^T, r_2^T, \ldots, r_m^T)^T \in R^{pm\times 1}$ and $B_i = (b_i^{1T}, b_i^{2T}, \ldots, b_i^{mT})^T \in R^{pm\times 1}$. Then we reformulate the objective function as

$$\min_{F} \sum_{i=1}^{n} (R^T B_i - f_i - a_i \odot e_i)^2 + \lambda(F-Y)(F-Y)^T + \mu R^T D_s R. \tag{7}$$

Denote $B = [B_1, B_2, \ldots, B_i] \in R^{pm\times n}$ and tr$(\cdot)$ as the trace of matrix. Then the optimization can be put as

$$\min_{R,F,E} \text{tr}\left((R^T B - F - A \odot E)^T (R^T B - F - A \odot E)\right) + \lambda(F-Y)(F-Y)^T + \mu R^T D_s R. \tag{8}$$

It can be further written as

$$\min_{R,F,E} \text{tr}\left(R^T (BB^T + \mu D_s) R\right) - 2\text{tr}\left(B^T R(F + A \odot E)\right) + \text{tr}\left((F + A \odot E)^T (F + A \odot E)\right) + \lambda(F-Y)(F-Y)^T. \tag{9}$$

Vanishing the derivative of Formula (9) with respect to R yields

$$2(BB^T + \mu D_s)R - 2B(F + A \odot E)^T = 0. \tag{10}$$

Then we get

$$R = (BB^T + \mu D_s)^{-1} B(F + A \odot E)^T = G_s B(F + A \odot E)^T, \tag{11}$$

where $G_s = (BB^T + \mu D_s)^{-1}$. Substituting R into Formula (9), we derive

$$\min_{R,F,E} -\text{tr}\left((F + A \odot E)B^T G_s{}^T B(F + A \odot E)^T\right) + \text{tr}\left((F + A \odot E)^T (F + A \odot E)\right) + \lambda(F-Y)(F-Y)^T. \tag{12}$$

Vanishing the derivative of Formula (12) with respect to F gives

$$(F + A \odot E)(-B^T G_s^T B + I_n) + \lambda(F - Y) = 0. \tag{13}$$

Then we get

$$F = (\lambda Y - A \odot E + A \odot EB^T G_s{}^T B)(-B^T G_s{}^T B + (\lambda + 1)I_n)^{-}1. \tag{14}$$

Optimizing E is equivalent to the following problem

$$\min_{E>0} \text{tr}\left((R^T B - F - A \odot E)^T (R^T B - F - A \odot E)\right). \tag{15}$$

The optimal solution to Formula (15) can be obtained by

$$e_i = \max\left((R^T B_i - f_i)/a_i, 0\right). \tag{16}$$

(2) **Fix** r_j and **optimize** s_j and F.

We denote $c_i^j = X_i^T r_j$, $S = (s_1^T, s_2^T, \ldots, s_m^T)^T$, $C_i = (c_i^1, c_i^2, \ldots, c_i^m)^T \in R^{qm\times 1}$, $C = [C_1, C_2, \ldots, C_i] \in R^{qm\times n}$, $D_r = \text{diag}(r_1^T r_1 I_q, \ldots, r_m^T r_m I_q)$. The formulation in (5) can be rewritten as

$$\min_{S,F,E} \text{tr}\left((S^T C - F - A \odot E)^T (S^T C - F - A \odot E)\right) + \lambda(F-Y)(F-Y)^T + \mu S^T D_r S. \tag{17}$$

Setting the derivatives of Formula (17) with respect to S to be zeros, we get

$$S = (CC^T + \mu D_r)^{-1} C(F + A \odot E)^T = G_r C(F + A \odot E)^T, \tag{18}$$

where $G_r = (CC^T + \mu D_r)^{-1}$. Substituting S into Formula (17), we get

$$\min_{S,F,E} -\text{tr}\left((F + A \odot E)C^T G_r C(F + A \odot E)^T\right) + \text{tr}\left((F + A \odot E)^T (F + A \odot E)\right) + \lambda(F-Y)(F-Y)^T. \tag{19}$$

Vanishing the derivative of Formula (19) with respect to F yields

$$(F + A \odot E)(-C^T G_r^T C + I_n) + \lambda(F - Y) = 0. \quad (20)$$

Then we can get

$$F = (\lambda Y - A \odot E + A \odot EC^T G_r{}^T C)(-C^T G_r^T C + (\lambda + 1)I_n)^{-}1, \quad (21)$$

and E is the projection

$$e_i = \max\left((S^T C_i - f_i)/a_i, 0\right). \quad (22)$$

To proceed, the optimizations of R, S, F and E are cyclically iterated until it converges. The predicted recognition score of test data x_k is computed by $\sum_{j=1}^{m} r_j^T x_k s_j$. For convenience of readers' reference, we list the specific procedures of our AMRM in Algorithm 1.

Algorithm 1 AMRM Algorithm

Input:
Training data $X \in R^{p \times q \times n}$,
Training data label $Y \in R^n$,
Training data related label $A \in R^n$.
Parameters μ and λ.

Initialize S , E and F.
while relative error $> \varepsilon$ **do**
Updating parameter:
Update F^{t+1} using Eq. (14).
Update R^{t+1} using Eq. (11).
Update E^{t+1} using Eq. (16).
Update F^{t+1} using Eq. (21).
Update S^{t+1} using Eq. (18).
Update E^{t+1} using Eq. (22).
end while

Output: Parameter R and S.

4 Experiment

We conduct the experiments on the large-scale action recognition dataset UCF101 and video activity recognition TRECVID MED 2011 dataset. The experimental settings, evaluation criteria, experimental results and the discussions have been presented in this section.

4.1 Experiment on UCF101 Dataset

4.1.1 Dataset

We first test our algorithm on the publicly available dataset UCF101 (Soomro et al. 2012), which is a large dataset for human action recognition. UCF101 consists of 101 action classes, 13K clips and 27 hours of video data, which makes it much more diverse than other datasets for action recognition. The videos in UCF101 were downloaded from YouTube, containing poor lighting, cluttered background, and severe camera motion. Frames of example videos are shown in Fig. 8. These videos have also been divided into five types: *human-object interaction, body motion only, human-human interaction, playing musical instruments* and *sports*. The reasons that we choose UCF101 as experimental dataset are as follows:

- As it is collected for YouTube, it contains real actions and poses significant challenges on action recognition.
- It contains nearly complete action classes in other action recognition datasets.
- It can be divided into different action types, which is suitable for our large-scale zero-shot learning task.

4.1.2 Feature Representation

Improved Dense Trajectory (IDT) features have been proved to be the most reliable features for action recognition and multimedia event recognition, which consist of different descriptors (HOG, HOF, MBHx and MBHy) to capture the shape and temporal motion information of videos. We adopt the improved trajectories proposed by Wang and Schmid (2013) to extract low-level features for each video in the UCF101 dataset with default parameters, that is, frames of length 15 for each trajectory on a dense grid with 5 pixel spacing. Inspired by recent success in image segmentation (Carreira et al. 2012), we pool the local features as Sect. 3.2 to form the global features. The pooled feature is normalized by subtracting the average value of the whole trajectories. To better evaluate the role of different features, we also separately apply the second-order pooling to the four types of trajectory descriptors.

4.1.3 Experimental Setting and Results

Our goal is action recognition that no positive exemplars are available. We divide the UCF101 dataset into two disjoint sets to make the problem more challenging. One set contains ten action classes for testing. The other set contains 91 classes for training. We apply Average Precision (AP) and mean Average Precision (mAP) as evaluation criteria. To visualize

Table 1 The mean Average Precision (mAP) comparisons with using few positive exemplars for 5 trials on UCF101 dataset

MAP	Trial 1	Trial 2	Trial 3	Trial 4	Trial 5
0 shot (ours)	**0.7204**	**0.7124**	**0.8764**	**0.7724**	**0.8346**
0 shot (Rohrbach et al. 2011)	0.5314	0.5138	0.6121	0.5968	0.7033
1 shot (Wang and Schmid 2013)	0.3118	0.3215	0.4062	0.3052	0.3674
2 shots (Wang and Schmid 2013)	0.4652	0.4321	0.5218	0.3987	0.4896
3 shots (Wang and Schmid 2013)	0.5013	0.4895	0.6053	0.4265	0.5517
4 shots (Wang and Schmid 2013)	0.5678	0.5496	0.6620	0.4958	0.6270
5 shots (Wang and Schmid 2013)	0.6078	0.5772	0.7432	0.5367	0.7759
1 shot (Wang et al. 2012)	0.3557	0.3834	0.4896	0.3576	0.4476
2 shots (Wang et al. 2012)	0.5576	0.4759	0.5766	0.5552	0.5872
3 shots (Wang et al. 2012)	0.6145	0.5464	0.6490	0.6014	0.6665
4 shots (Wang et al. 2012)	0.6478	0.6149	0.7765	0.6847	0.7365
5 shots (Wang et al. 2012)	0.6942	0.6742	0.8386	0.7437	0.8143

The best results are highlighted in bold

the performance, we also show the highest ranking results for each test class in the UCF101 dataset in Fig. 8.

We set three baselines to evaluate the effectiveness of the proposed approach. (1) Direct similarity based approach (Rohrbach et al. 2011, 2010; Liu et al. 2013); These approaches consider the known classes as attributes. (2) Fully supervised action recognition approach with few positive exemplars (Wang and Schmid 2013), (3) Semi-supervised approach with few positive exemplars (Wang et al. 2012).

In the direct similarity based zero-shot experiment, we implement the approach on our own, since it is quite straightforward. We first split the action classes into 91 training and 10 testing classes. For each training class, we use the videos belonging to that class as positive data and the videos belonging to other training classes as negative data, to train a binary classifier. In the testing phase, we select five related action classes in the training set based on semantic similarity scores with each testing classes as described in Rohrbach et al. (2011, 2010), and then combine them to recognize the testing videos. For n-shot (n positive exemplars are available) experiments, we utilize the randomly selected 1, 2, 3, 4 and 5 positive videos from the target action and other category videos to train the target action classifier. In our AMRM zero-shot experiments, we use 20 related action classes (no positive) to train the target action class. The top three action classes in analogy pool, we assign them with $a_i = 1$. The top four to ten action classes, we assign them with $a_i = 0$. The remaining 10 action classes, we assign them $a_i = -1$. We perform six trial experiments by randomly splitting the training classes and testing classes. One trial is used for tuning the free parameters μ and λ. We keep the testing classes of the first trial have no overlap with other five trials. The search ranges of parameters are $\lambda \in \{0.1, 1, 10, 100\}$ and $\mu \in \{0.1, 1, 10, 100, 1000\}$. To be noted, the five positive data used in the 5-shot experiment are excluded from the testing set in all experiments, so the testing data are the same in each trial. We then report the remaining five trial experimental results by setting optimal value $\lambda = 1$ and $\mu = 10$. The mAP scores for each trail are reported in Table 1. We can see that the mAP scores of our method are between 0.8764 and 0.7124, not only significantly beat the attribute representation approaches, but also outperform the state-of-art supervised and semi-supervised action recognition approaches that use 1, 2, 3, 4 and 5 positive exemplars.

For the consistent evaluation of zero-shot action recognition, we have selected ten testing classes for public comparison:[1] *apply lipstick, boxing punching bag, floor gymnastics, front crawl, horse riding, playing violin, soccer penalty, throw discus, trampoline jumping* and *volleyball spiking*. Thus our testing set consists of 1400 videos of those class actions, while the 12,000 videos of the remaining 91 classes can be used for training. Additionally, we also encourage the use of the dataset for the regular complex large-scale zero-shot action recognition setting. In particular, we expect the splits of the UCF101 dataset to be suitable to test the performances of zero-shot action recognition, because the choice of testing classes covers different action types and some classes also look visual similar, which makes the action recognition difficult.

We firstly show the compared results with the direct similarity based approach. From Table 2, we can find that the proposed approach achieves better performances for all the action classes than the direct similarity based approach. It further validates our claim that the proposed approach can address the inaccuracy of the attribute classifiers and then is more suitable for the zero-shot action recognition. We then show the compared results for each action class using five positive exemplars in Table 2. It can be found that the proposed method achieves the highest accuracies for seven

[1] Trial 5 in Table 1.

Table 2 Comparisons with baseline approaches on UCF101 dataset (trial 5)

Action name	0 shot (%) (Rohrbach et al. 2011)	0 shot (ours) (%)	5 shots (%) Wang and Schmid (2013)	5 shots (%) (Wang et al. 2012)
Apply lipstick	79.63	**95.73**	82.09	86.32
Boxing punching bag	68.32	72.92	**83.93**	76.94
Floor gymnastics	76.33	**86.83**	81.35	83.71
Front crawl	87.26	**99.88**	92.57	94.22
Horse riding	67.22	**74.85**	66.07	70.34
Playing violin	58.55	61.69	84.75	**86.21**
Soccer penalty	83.22	**96.45**	86.36	87.65
Throw discus	65.89	**76.47**	64.58	74.86
Trampoline jumping	74.48	82.94	82.46	**88.35**
Volleyball spiking	42.47	**86.88**	51.71	65.69
mAP	70.33	**83.46**	77.59	81.43

The best results are highlighted in bold
Average Precision (AP) per class and mean Average Precision (mAP) over all classes

Table 3 Action recognition with different descriptors on UCF 101 dataset (trial 5)

Target action	MM_{hog} (%)	MM_{hof} (%)	MM_{mbhx} (%)	MM_{mbhy} (%)
Apply lipstick	**81.03**	79.88	74.55	67.75
Boxing punching bag	**70.74**	35.34	47.10	43.84
Floor gymnastics	34.97	30.68	**49.96**	36.90
Front crawl	**97.18**	92.11	84.78	89.09
Horse ride	34.13	40.61	**67.95**	26.84
Play violin	26.23	18.40	32.47	**40.24**
Soccer penalty	29.88	**35.25**	33.56	25.34
Throw discus	37.29	39.91	**43.77**	37.46
Trampoline jump	**21.15**	9.26	15.32	11.23
Volleyball spike	**28.56**	25.97	16.91	21.77
mAP	46.12	40.74	**46.64**	40.05

The best results are highlighted in bold
Average Precision (AP) per class and mean Average Precision (mAP) over all classes

testing classes among the whole ten classes when the number of the positive exemplars increases to be 5. To further validate the effectiveness of adaptive multi-model rank-preserving mapping (AMRM) approaches, we also compare the average precision (AP) results for each testing class with the vector-based Kernel Ridge Regression (KR) (Vovk 2013) and the multi-model mapping (MM) method in Fig. 6. To better evaluate the proposed method, we also report the results on different features in Table 3 and multiple models in Fig. 4 for the zero-shot action recognition task. To test the framework of our ranking approach, we also conduct an experiment by replacing the trajectory features to the C3D (Tran et al. 2014) features, and use the kernel ridge regression as ranking function. Experiment results on Table 4 shows that multi-model mapping (MM) approach by using trajectory feature is better than the Kernel Ridge Regression approach by using C3D features. Nevertheless, simple late fusing the decision score of multi-model mapping (MM) which use trajectory features and the Kernel Ridge Regression approach that takes C3D feature as input will further improve the zero-shot action recognition performances.

4.1.4 Matrix Mapping Versus Vector Mapping

As discussed in the previous section, matrix form may be a more natural representation of images and video frames to reflect their structures. However, most existing classification algorithms require that an image or video is represented by a vector, which is usually obtained by concatenating all rows (or columns) of an image matrix. Aiming to preserve the second-order spatial structures within images while reducing the computational complexity, we propose the second-order mapping-based methods. In our experiment setting, we also compare the mAP value between the multi-model map-

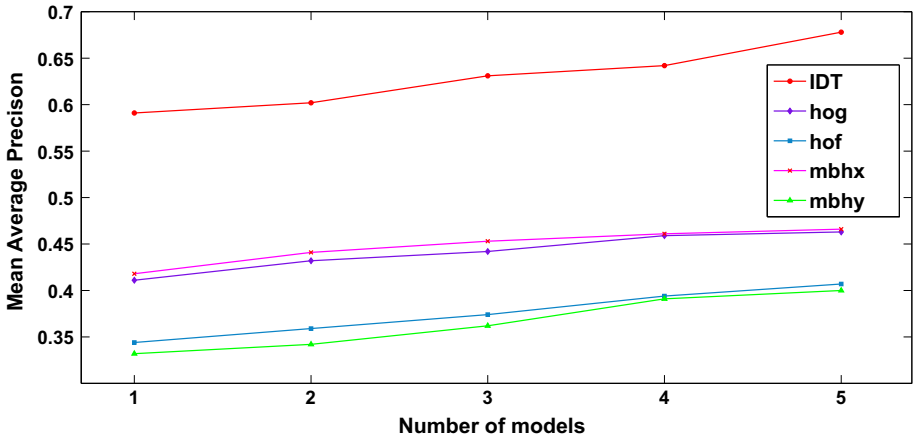

Fig. 4 Mean average precision (mAP) value *w.r.t.* number of models (*m* in Eq. (5)) on UCF101 dataset (trial 5)

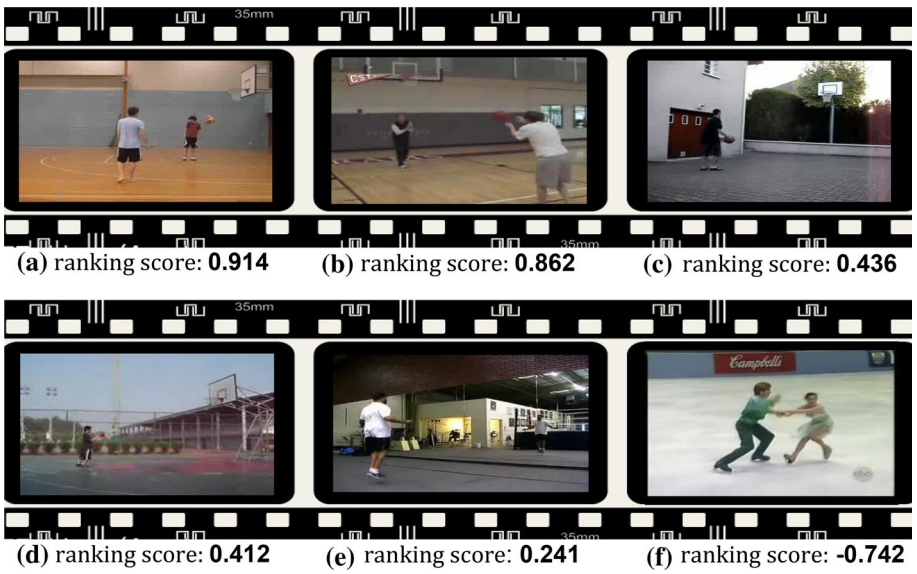

Fig. 5 Derived ranking scores of source videos **a**–**d** *basketball shooting*, **e** *jumping rope*, and **f** *ice dancing* for recognizing the target action *volleyball spiking*

ping (MM) with the vector-based Kernel Ridge Regression method (KR) (Vovk 2013) in Fig. 6. It can be seen that the matrix representation form can achieve better recognition results.

4.1.5 Different Descriptors

In this section, we analyze the role of different descriptors for the zero-shot action recognition using multi-model mapping approach, namely HOG (MM_{hog}), HOF (MM_{hof}), MBHx (MM_{mbhx}), and MBHy (MM_{mbhy}), as shown in Table 3. From the experimental results, we can easily find that action recognition for some new classes may rely more on the similarities of the objects and the scenes, such as *apply lipstick* that often occurs with human face, and *front crawl* that often occurs in the water. And some classes may rely more on the similarities of motion, such as *boxing punching box*, that is very similar to *boxing speed bag* and *punch*, and *volleyball spiking* that is similar to *jumping* and *basketball*.

4.1.6 Multiple Models Versus Single Model

We also demonstrate the relationship between the recognition performances and the number of models in Fig. 4. It can be observed that the performances are improved with the increment of model numbers. In other words, using multiple models is better than a single model to map the second-order representation to semantic labels.

4.1.7 With Adaptive Labels Versus Without Adaptive Labels

We first use some examples to show how the proposed AMRM method adaptively assigns ranking scores to related exemplars. As Fig. 5 shows, the highly related class to the target action *volleyball spiking* is *basketball shooting* Fig. 5a–d, the related action class is *jumping rope* Fig. 5e, and the unrelated action is *ice dancing* Fig. 5d. Their semantic correlations computed from *wikipedia* is 0.166, 0.113 and 0.014, respectively. However, the learned ranking scores are different. It can be viewed that the proposed approach can better

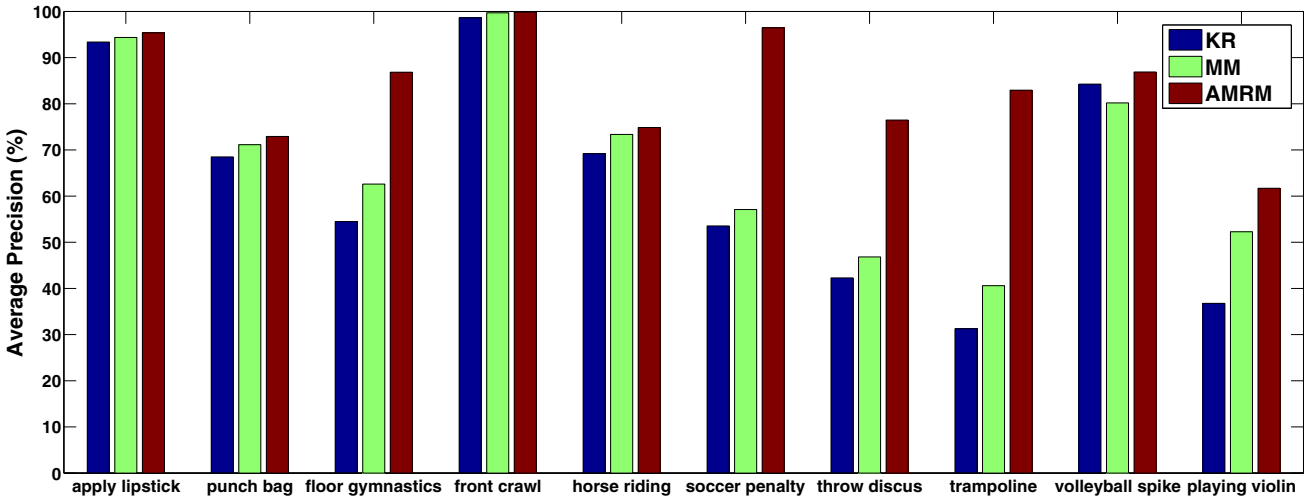

Fig. 6 Average Precision (AP) comparisons between kernel ridge regression (KR), multi-model mapping (MM) and adaptive multi-model rank-preserving mapping (AMRM) on UCF101 dataset (trial 5). The higher scores indicate the better results

separate different classes and adaptively utilize the training exemplars on a per video basis. For example, videos Fig. 5a–d are all named as *basketball shooting*, but (a) and (b) in Fig. 5 are more related to the target action. Therefore, they are assigned the higher ranking scores for training. In Fig. 6, we can see that our AMRM method using the optimal ranking scores for training improves the performance significantly.

4.1.8 Known Classes Versus Unknown Classes

In this section, we test the performance of our approach when testing data contains videos of highly related action classes in the analogy pool. The objective is to evaluate whether the proposed ranking-based approach can also distinguish the testing classes from related training classes. In this experiment, we intentionally add videos belonging to the three highly related action classes that used for training into the testing set. Therefore, there are five types of videos: (1) target action: the action class that we want to recognize; (2) known action-1: the most highly related action class used for training which are selected from the analog pool by the method discussed in Sect. 3.1; (3) known action-2: the second highly related action class used for training; (4) known action-3: the third highly related action class used for training; (5) others: the remaining nine unknown testing classes. We use the original ranking model to rank all the videos in the testing set. We use the number of top N returned videos to evaluate if our approach is able to discriminate the highly related action and the target action.

We first report the experiment results when N is 10, 20 and 50 in Table 5. We also report the mean class classification accuracy for top 50 returned videos (Macc@50) in Table 6. It can be observed that the videos of target action are not only ranked higher than the unknown testing classes, but also ranked higher than highly related known action which are used for training. This observation indicates that our approach is able to discriminate target action from highly related known classes. However, we also observe that highly related known actions are ranked higher than other videos, but it is reasonable because they are more similar to the target action than other videos.

Table 4 The mean Average Precision (mAP) of action recognition using different features and ranking approach on UCF 101 dataset (trial 5)

KR_{C3D}	MM_{traj}	$KR_{C3D} + MM_{traj}$
64.54 %	67.83 %	**72.28 %**

The best result is highlighted in bold

4.1.9 How Many Background Actions Are Needed?

For the attribute-based approaches, nearly all the classes should be used to train attribute models, which costs much memory and training time. However, in the experiments, we find that not all the source action classes are useful for our second-order mapping-based zero-shot action recognition method. After the most related action classes are selected, the variety and number of other action classes that are used as negative data, does not influence the results at all. Therefore our approach is more robust and efficient for large-scale zero-shot learning task. More action classes used can improve the experiment results slightly, but will lose the efficiency. Therefore, to balance the action recognition performance and the algorithm efficiency, we use 20 action classes to train a classifier for recognizing a novel action class. It takes about 1 min for training a novel action classifier and less than 10 s to predict 1400 videos using a PC with 3.4 GHz CPU and 32G memory in MATLAB 2012.

Table 5 Number of videos in the top *N* ranking list. Known-1, known-2 and known-3 mean the 3 highly related action classes that we used to train the ranking model for each target action class

Action class	Metric	target class	Known-1	Known-2	Known-3	Others
Apply lipstick	N = 10	7	2	1	0	0
	N = 20	13	4	2	0	1
	N = 50	34	8	4	1	3
Boxing punching bag	N=10	6	2	1	0	1
	N = 20	11	3	2	1	3
	N = 50	31	7	4	3	5
Floor gymnastics	N=10	8	1	1	0	0
	N = 20	14	3	2	0	1
	N = 50	36	6	4	2	2
Front crawl	N=10	9	1	0	0	0
	N = 20	18	2	0	0	0
	N = 50	42	6	1	0	1
Horse ride	N=10	6	3	0	0	1
	N = 20	11	7	0	0	2
	N = 50	30	13	2	0	5
Play violin	N = 10	5	2	1	1	1
	N = 20	10	3	2	2	3
	N = 50	27	6	6	5	6
Soccer penalty	N=10	9	1	0	0	0
	N = 20	16	2	1	0	1
	N = 50	44	3	2	0	1
Throw discus	N=10	6	1	1	1	1
	N = 20	12	3	2	1	2
	N = 50	28	6	6	5	5
Trampoline jump	N=10	6	1	1	1	1
	N = 20	13	2	2	1	2
	N = 50	33	5	4	4	4
Volleyball spike	N=10	7	1	0	0	2
	N = 20	14	2	1	0	3
	N = 50	35	5	3	2	5

Others means the original 9 action classes used for testing

Table 6 Mean class classification accuracy for top 50 returned videos (Macc@50)

Setting	Original	Mix
Macc@50	74.83 %	68.00 %

Mix/original means whether the test set is mixed with the training videos or not

4.2 Experiment on Multimedia Event Detection

4.2.1 Dataset

To further evaluate the effectiveness of the proposed algorithm, we conduct experiments on a more challenging dataset, i.e., the TRECVID MED 2011 development dataset.[2] This dataset contains 9746 unconstrained web videos with large variations in duration, quality and resolution. Following Lan et al. (2012) and Tang et al. (2013), we use the ten official testing events (E006−E015) outlined by the National Institute of Standards and Technology (NIST) to evaluate the performance against the baseline approaches. For each event, NIST also releases a text description called the event kit, which includes an event name and key evidences that are expected to be observed in the videos. The event names are listed in Table 7.

4.2.2 Experiment Settings

In all experiments, we followed the split up defined in Lan et al. (2012), and use all the videos in the test set to compare

[2] http://www.nist.gov/itl/iad/mig/med11.cfm.

Table 7 The video event detection results compared with the baseline approaches on MED 2011 dataset

Event name	0-shot (Rohrbach et al. 2011)	0-shot (ours)	5-shot (SVM)	5-shot (KR)
Birthday party	0.074	**0.149**	0.087	0.094
Changing a vehicle tire	0.015	0.038	0.043	**0.059**
Flash mob gathering	0.193	**0.302**	0.264	0.279
Getting a vehicle unstuck	0.033	0.055	0.091	**0.117**
Groom an animal	0.096	**0.197**	0.132	0.146
Making a sandwich	0.031	**0.091**	0.065	0.071
Parade	0.226	**0.343**	0.267	0.281
Parkour	0.213	**0.399**	0.337	0.344
Repairing an appliance	0.066	**0.167**	0.146	0.159
Work on a sewing project	0.071	**0.147**	0.115	0.123
mAP	0.102	**0.189**	0.155	0.167

The best results are highlighted in bold

the performance. We compare the baseline of direct similarity based zero-shot approach (Rohrbach et al. 2011, 2010; Liu et al. 2013), and the fully supervised n-shots approaches using improved dense trajectory (Wang and Schmid 2013) with Fisher Vector encoding (Oneata et al. 2013), which is reported as the best single feature for video event detection (Oneata et al. 2013). For all the zero-shot experiments, we take the textual event name and key evidence described in the event kit as the input and use the videos in the UCF101 dataset as training data. No positive training videos in the MED 2011 development dataset contained.

Direct similarity based zero-shot event detection For direct similarity based approach, we firstly train action detectors for the 101 action classes defined in the UCF101 dataset. For each action class, we use all the videos belonging to this class as positive data, and randomly sample 5000 videos from other action classes as negative data to train a binary classifier. For all the action classifiers training, where linear SVM are used, we employ fivefold cross validations for parameter tuning. The search ranges of this parameter are {0.01, 0.1, 1, 10, 100}. Finally, in order to get the action detection score, we directly apply the action detectors to the testing videos. Thus each video in the testing set is represented as a 101 dimensional vector, and each dimension corresponds to the detection score of a known action class. In the testing time, given an event name, we use the top five related action classes to retrieve videos.

N-shots approaches In the n-shots experiment, we use n positive videos which are randomly selected from the training set as positive data and all the null videos as negative data to train a binary classifier for each event. Following Lan et al. (2013), we use Support Vector Machines (SVMs) and Kernel Ridge Regression (KR) as classifiers. To alleviate the influence of variances between positive videos, we repeat experiments on ten groups of randomly-generated training data. The average mAP are then reported in Table 7.

AMRM approach We use 20 related action classes to train the classifier for each event. We assign $a_i = 1$ for the top three action classes in analogy pool. We assign $a_i = 0$ for the top four to ten action classes. The remaining ten action classes are assigned by $a_i = -1$. We also use the same parameters ($\lambda = 1$ and $\mu = 10$) obtained in Sect. 4.1.3 to train the event detector.

4.2.3 Experiment Result and Discussion

From Table 7, we can see that our approach outperforms the direct similarity based approach for all events, and outperform 5-shots supervised approaches for 8 events out of the ten events. This result confirms that AMRM is a general approach for the zero-shot video analysis. A possible explanation that our algorithm fails in two events is due to the lack of related action classes in the UCF101 dataset. We show the relationship between the recognition precision and the mean Semantic Correlation (mSC) of the target event and the top three highly related source actions in Table 8. It can be found that the recognition performance will degrade, if the semantic correlations between the target action and all source actions are all relatively low. For example, there are no *vehicle* related concepts in the UCF101 dataset, which make the performance of certain events, such as *Changing a vehicle tire* and *Getting a vehicle unstuck* rather low. We argue that this is still reasonable, since the zero-shot recognition is considerable difficult and can hardly be solved without transferring knowledge from potentially related actions as described in Rohrbach et al. (2011). However, as the number of source action classes increases, we believe that the performance will be improved.

Table 8 The relationship between semantic relatedness (mSC) and recognition precision (AP) on MED 2011 dataset

Event name	mSC	AP	Event name	mSC	AP
Birthday party	0.68	0.149	Making a sandwich	0.61	0.091
Changing a vehicle tire	0.16	0.038	Parade	0.93	0.343
Flash mob gathering	0.85	0.302	Parkour	0.84	0.399
Getting a vehicle unstuck	0.18	0.055	Repairing an appliance	0.53	0.167
Groom an animal	0.59	0.197	Work on a sewing project	0.49	0.147

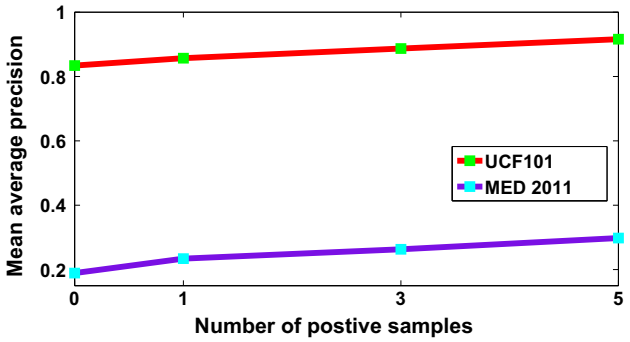

Fig. 7 Mean average precision (mAP) value *w.r.t.* number of positive samples on UCF101 and MED 2011 datasets

4.3 Beyond Zero-Shot Learning

In this section, we report results when learning with few examples (few-shots). The goal is to show that, the proposed AMRM approach could also incorporate few-labels and benefit few-shots action recognition.

In this experiment, we assume that we have few (e.g 1, 3, 5) positive samples for each action class plus the samples from other known action classes. We simply set the label y_i of the positive exemplar x_i as 1 and its degree of relatedness a_i is set to 1.5, which indicates that it is more relevant than the exemplars belonging to other related action classes. We show results in Fig. 7. It can be seen that our proposed AMRM framework can easily incorporate the positive exemplars and consistently improve the recognition performance both on the UCF101 and TRECVID MED 2011 datasets, which further validates the principle of our framework. In addition, compared with the results of N-shot approaches in Tables 1 and 7, we can find the proposed AMRM is also an effective approach to improve N-shot action recognition performances.

Fig. 8 **a–j** represent the highest ranking results for testing class *apply lipstick*, *boxing punching bag*, *floor gymnastics*, *front crawl*, *playing violin*, *horse riding*, *soccer penalty*, *throw discus*, *trampoline jumping* and *volleyball spiking* in the UCF101 dataset. Uniquely characterized classes are well identified, e.g. *apply lipstick* and *front crawl*. Confusions occur between visually similar classes, e.g. *floor gymnastics* and *trampoline jump*

5 Conclusion

In this paper, we proposed an approach that merely takes an action name as the input to recognize the action of interest without any attribute classifiers and positive exemplars. Given an action name, we first built an analogy pool consisting of a series of related actions that share certain common elements to the target action. According to the correlations inferred from the external ontology *wikipedia*, we then applied an adaptive multi-model rank-preserving mapping (AMRM) algorithm to train a classifier for action recognition. We showed that manual supervision can be fully replaced by tapping into linguistic sources in principle (Fig. 8). Extensive experiments have been carried out to validate our claims and confirmed our intuition that transferring knowledge from external knowledge bases and using the related source videos is an efficient and effective approach to perform action recognition. In future, we will explore the potentials of our approach to other visual recognition tasks, such as object recognition.

Acknowledgments This work was partially supported by the 973 Program (No. 2012CB316400), partially supported by the National Natural Science Foundation of China Grant 61033001, 61361136003, and partially supported by the ARC DECRA (DE130101311), the ACR DP (DP150103008). This work was done when Chuang Gan was a visiting student at Zhejiang University.

References

Akata, Z., Perronnin, F., Harchaoui, Z., & Schmid, C., et al. (2013). Label-embedding for attribute-based classification. In *CVPR* (pp. 819–826).

Blank, M., Gorelick, L., Shechtman, E., Irani, M., & Basri, R. (2005). Actions as space-time shapes. In *ICCV* (Vol. 2, pp. 1395–1402).

Cai, J., Zha, Z. J., Zhou, W., & Tian, Q. (2012). Attribute-assisted reranking for web image retrieval. In *Multimedia* (pp. 873–876). ACM.

Carreira, J., Caseiro, R., Batista, J., & Sminchisescu, C. (2012). Semantic segmentation with second-order pooling. In *ECCV* (pp. 430–443).

Chen, M. Y., & Hauptmann, A. (2009). *Mosift: Recognizing human actions in surveillance videos*.

Duan, L., Xu, D., Tsang, I. H., & Luo, J. (2012). Visual event recognition in videos by learning from web data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *34*(9), 1667–1680.

Farhadi, A., Endres, I., Hoiem, D., & Forsyth, D. (2009). Describing objects by their attributes. In *CVPR*, (pp. 1778–1785).

Frome, A., Corrado, G. S., Shlens, J., Bengio, S., Dean, J., & Mikolov, T., et al. (2013). Devise: A deep visual-semantic embedding model. In *NIPS*, (pp. 2121–2129).

Fu, Y., Hospedales, T. M., Xiang, T., Fu, Z., & Gong, S. (2014). Transductive multi-view embedding for zero-shot recognition and annotation. In *ECCV* (pp. 584–599).

Fu, Y., Hospedales, T. M., Xiang, T., & Gong, S. (2015). Transductive multi-view zero-shot learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *37*(11), 2332–2345.

Gabrilovich, E., & Markovitch, S. (2007). Computing semantic relatedness using wikipedia-based explicit semantic analysis. In *IJCAI* (pp. 1606–1611).

Guadarrama, S., Krishnamoorthy, N., Malkarnenkar, G., Venugopalan, S., Mooney, R., Darrell, T., & Saenko, K. (2013). Youtube2text: Recognizing and describing arbitrary activities using semantic hierarchies and zero-shot recognition. In *ICCV* (pp. 2712–2719).

Hauptmann, A., Yan, R., Lin, W. H., Christel, M., & Wactlar, H. (2007). Can high-level concepts fill the semantic gap in video retrieval? a case study with broadcast news. *IEEE Transactions on Multimedia*, *9*(5), 958–966.

Jiang, Y. G., Bhattacharya, S., Chang, S. F., & Shah, M. (2013). High-level event recognition in unconstrained videos. *International Journal of Multimedia Information Retrieval*, *2*(2), 73–101.

Kankuekul, P., Kawewong, A., Tangruamsub, S., & Hasegawa, O. (2012). Online incremental attribute-based zero-shot learning. In *CVPR* (pp. 3657–3664).

Lampert, C.H., Nickisch, H., & Harmeling, S. (2009). Learning to detect unseen object classes by between-class attribute transfer. In *CVPR* (pp. 951–958).

Lan, Z. Z., Bao, L., Yu, S. I., Liu, W., & Hauptmann, A. G. (2012). *Double fusion for multimedia event detection*.

Lan, Z.Z., Jiang, L., Yu, S.I., Rawat, S., Cai, Y., Gao, C., Xu, S., Shen, H., Li, X., & Wang, Y., et al. (2013). Cmu-informedia at trecvid 2013 multimedia event detection. In *TRECVID 2013 Workshop* (Vol. 1, p. 5).

Laptev, I. (2005). On space-time interest points. *International Journal of Computer Vision*, *64*(2–3), 107–123.

Laptev, I., Marszalek, M., Schmid, C., & Rozenfeld, B. (2008). Learning realistic human actions from movies. In *CVPR* (pp. 1–8).

Liu, J., Kuipers, B., & Savarese, S. (2011). Recognizing human actions by attributes. In *CVPR* (pp. 3337–3344).

Liu, J., Yu, Q., Javed, O., Ali, S., Tamrakar, A., Divakaran, A., Cheng, H., & Sawhney, H.S. (2013). Video event recognition using concept attributes. In *WACV* (pp. 339–346).

Ma, Z., Yang, Y., Nie, F., Sebe, N., Yan, S., & Hauptmann, A. G. (2014). Harnessing lab knowledge for real-world action recognition. *International Journal of Computer Vision*, *109*(1–2), 60–73.

Ma, Z., Yang, Y., Sebe, N., Zheng, K., & Hauptmann, A. G. (2013). Multimedia event detection using a classifier-specific intermediate representation. *IEEE Transactions on Multimedia*, *15*(7), 1628–1637.

Ma, Z., Yang, Y., Xu, Z., Sebe, N., & Hauptmann, A.G. (2013). We are not equally negative: Fine-grained labeling for multimedia event detection. In *ACM Multimedia* (pp. 293–302).

Oneata, D., Verbeek, J., & Schmid, C., et al. (2013). Action and event recognition with fisher vectors on a compact feature set. In *ICCV*.

Reddy, K. K., & Shah, M. (2013). Recognizing 50 human action categories of web videos. *Machine Vision and Applications*, *24*(5), 971–981.

Rohrbach, M., Ebert, S., & Schiele, B. (2013). Transfer learning in a transductive setting. In *NIPS* (pp. 46–54).

Rohrbach, M., Regneri, M., Andriluka, M., Amin, S., Pinkal, M., & Schiele, B. (2012). Script data for attribute-based recognition of composite activities. In *ECCV* (pp. 144–157).

Rohrbach, M., Stark, M., & Schiele, B. (2011). Evaluating knowledge transfer and zero-shot learning in a large-scale setting. In *CVPR* (pp. 1641–1648).

Rohrbach, M., Stark, M., Szarvas, G., Gurevych, I., & Schiele, B. (2010). What helps where-and why? Semantic relatedness for knowledge transfer. In *CVPR* (pp. 910–917).

Sadanand, S., & Corso, J. J. (2012). Action bank: A high-level representation of activity in video. In *CVPR* (pp. 1234–1241).

Sánchez, J., Perronnin, F., Mensink, T., & Verbeek, J. (2013). Image classification with the fisher vector: Theory and practice. *International Journal of Computer Vision*, *105*(3), 222–245.

Socher, R., Ganjoo, M., Manning, C.D., & Ng, A. (2013). Zero-shot learning through cross-modal transfer. In *NIPS* (pp. 935–943).

Soomro, K., Zamir, A. R., & Shah, M. (2012). UCF101: A dataset of 101 human actions classes from videos in the wild. arXiv preprint arXiv:1212.0402.

Sun, C., Gan, C., & Nevatia, R. (2015). Automatic concept discovery from parallel text and visual corpora. In *ICCV*.

Tang, K., Yao, B., Fei-Fei, L., & Koller, D. (2013). Combining the right features for complex event recognition. In *ICCV* (pp. 2696–2703).

Torresani, L., Szummer, M., & Fitzgibbon, A. (2010). Efficient object category recognition using classemes. In *ECCV* (pp. 776–789).

Tran, D., Bourdev, L., Fergus, R., Torresani, L., & Paluri, M. (2014). C3D: generic features for video analysis. arXiv preprint arXiv:1412.0767.

Vovk, V. (2013). Kernel ridge regression. In *Empirical inference* (pp. 105–116).

Wang, H., Klaser, A., Schmid, C., & Liu, C. L. (2011). Action recognition by dense trajectories. In *CVPR* (pp. 3169–3176).

Wang, H., Kläser, A., Schmid, C., & Liu, C. L. (2013). Dense trajectories and motion boundary descriptors for action recognition. *International Journal of Computer Vision*, *103*(1), 60–79.

Wang, H., & Schmid, C. (2013). Action recognition with improved trajectories. In *ICCV* (pp. 3551–3558).

Wang, S., Yang, Y., Ma, Z., Li, X., Pang, C., & Hauptmann, A. G. (2012). Action recognition by exploring data distribution and feature correlation. In *CVPR* (pp. 1370–1377).

Yang, Y., Ma, Z., Nie, F., Chang, X., & Hauptmann, A. G. (2014). Multi-class active learning by uncertainty sampling with diversity maximization. *International Journal of Computer Vision*, *113*(2), 113–127.

Yu, F. X., Cao, L., Feris, R. S., Smith, J. R., & Chang, S. F. (2013). Designing category-level attributes for discriminative visual recognition. In *CVPR* (pp. 771–778).