Mining Motion Atoms and Phrases for Complex Action Recognition [*]

LiMin Wang[1,2], Yu Qiao[2] [†], and Xiaoou Tang[1,2]
[1]Department of Information Engineering, The Chinese University of Hong Kong
[2]Shenzhen key lab of Comp. Vis. & Pat. Rec., Shenzhen Institutes of Advanced Technology, CAS, China
07wanglimin@gmail.com, yu.qiao@siat.ac.cn, xtang@ie.cuhk.edu.hk

Abstract

This paper proposes motion atom and phrase as a mid-level temporal "part" for representing and classifying complex action. Motion atom is defined as an atomic part of action, and captures the motion information of action video in a short temporal scale. Motion phrase is a temporal composite of multiple motion atoms with an AND/OR structure, which further enhances the discriminative ability of motion atoms by incorporating temporal constraints in a longer scale. Specifically, given a set of weakly labeled action videos, we firstly design a discriminative clustering method to automatically discover a set of representative motion atoms. Then, based on these motion atoms, we mine effective motion phrases with high discriminative and representative power. We introduce a bottom-up phrase construction algorithm and a greedy selection method for this mining task. We examine the classification performance of the motion atom and phrase based representation on two complex action datasets: Olympic Sports and UCF50. Experimental results show that our method achieves superior performance over recent published methods on both datasets.

1. Introduction

Human action recognition is an important problem in computer vision and has gained extensive research interests recently [1] due to its wide applications in surveillance, human-computer interface, sports video analysis, and content based video retrieval. State-of-the-art methods [24, 17] have performed well on simple actions recorded in constrained environment such as walking, running (e.g. KTH datast [21], Weizmann dataset [9]) . However, classification of complex actions (e.g. Olympic Sports dataset [15], UCF50 dataset [18]) in unconstrained environment is still challenging. Firstly, due to background clutter, viewpoint changes, and motion speed variation, there exist always large intra-class appearance and motion variations within the same class of action. Secondly, different from simple actions, a complex action always exhibits richer temporal structures and is composed of a sequence of atomic actions.

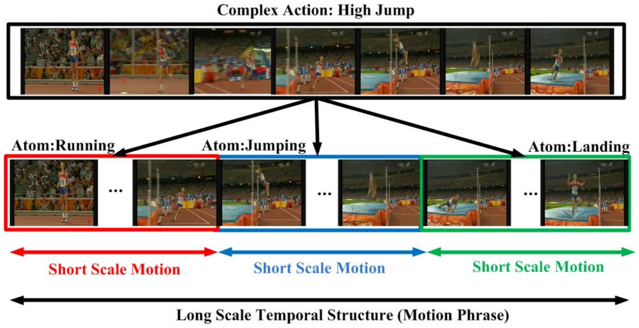

Figure 1. Complex actions usually can be decomposed into several motion atoms. For example of high jump, it contains atoms: running, jumping, and landing. For each atom, it describes a short temporal scale motion information, which can be shared by different complex action classes. For each complex action, there exist temporal structures of multiple atoms in a long temporal scale.

Recently, researches show that the temporal structures of complex action yield effective cues for action classification [8, 15, 23, 26]. As shown in Figure 1, from a long temporal scale, a complex action can be decomposed into a sequence of atomic motions. For instance, the sport action of high-jump can be decomposed into running, jumping, and landing. There exist different temporal configurations among these atomic motions for different action classes. From a short temporal scale, each atomic motion corresponds to a simple pattern and these atomic motions may be shared by different complex action classes. For example, both actions of long-jump and triple-jump include running. These observations offer us insights to complex action recognition:

- Unsupervised discovery of motion atoms. Motion atoms describe simple motion patterns, and can be seen as mid-level units to bridge the gap between low-level features and high-level complex actions. However, it is not straightforward to define and obtain mo-

[*]This work is partly supported by National Natural Science Foundation of China (61002042), Shenzhen Basic Research Program (JC201005270350A, JCYJ20120903092050890, JCYJ20120617114614438), 100 Talents Programme of Chinese Academy of Sciences, and Guangdong Innovative Research Team Program (No.201001D0104648280).

[†]Corresponding Author.

tion atoms from current public action datasets. In most action datasets, we only have class labels and do not have detailed annotations for the type and duration of each motion atom. Furthermore, it is heavily time-consuming to manually label atomic actions in each video. We need to design an unsupervised method to discover a set of motion atoms automatically from video dataset.

- Mining temporal composite of motion atoms. A single motion atom describes motion information in a short temporal scale. The discriminative power of motion atom is limited by its temporal duration. We note that the temporal structure (i.e. sequential composition of motion atoms), captures motion information in a longer scale and provides important cue to discriminate different action classes. However, the number of candidate combinations is exponential with the number of atomic actions and much of the combinations are not discriminative for classification. We need to mine an effective subset of temporal composites of motion atoms.

Based on the above insights, this paper proposes *motion atom and phrase*, a mid-level representation of action video, which jointly encodes the motion, appearance, and temporal structure of multiple atomic actions. Firstly, we discover a set of *motion atoms* from training samples in an unsupervised manner. These training action videos are only equipped with class labels. We transform this task into a discriminative clustering problem. Specifically, we develop an iterative algorithm, which alternates between clustering segments and training classifier for each cluster. Each cluster corresponds to a motion atom. These atoms act as building units for video representation. Then, we construct *motion phrase* as a temporal composite of multiple atoms. It not only captures short-scale motion information of each atom, but also models the temporal structure of multiple atoms in a longer temporal scale. We resort to an AND/OR structure to define motion phrase, which allows us to deal with temporal displacement effectively. We propose a bottom-up mining algorithm and greedy selection method to obtain a set of motion phrases with high discriminative and representative power. Finally, we represent each video by the *activation vector* of motion atoms and phrases by max pooling the response score of each atom and phrase. We conduct experiments on two complex action datasets: Olympic Sports dataset [15] and UCF50 dataset [18]. The experimental results show that the proposed methods outperform recent published methods.

2. Related Work

Action recognition has been studied extensively in recent years and readers can refer to [1] for good surveys. Here, we only cover the works related to complex action recognition.

Complex actions refer to those that contain several atomic actions such as Olymipc Sports actions [15], and Cooking Composite actions [19]. Many researches use state-observation sequential models, such as Hidden Markov Models (HMMs) [16], Hidden Conditional Random Fields (HCRFs) [27], and Dynamic Bayesian Networks (DBNs) [13], to model the temporal structure of action. Niebles *et al.* [15], Tang *et al.* [23], and Wang *et al.* [26] propose to use latent variables to model the temporal decomposition of complex actions and resort to Latent SVM [6] to learn the model parameters in an iterative approach. Gaidon *et al.* [8] annotate each atomic action for each video data and propose Actom Sequence Model (ASM) for action detection. Different from these approaches, we focus on learning a set of feature units, i.e. motion atoms and phrase, to represent video of complex action, and our representation is flexible with the classifier used for recognition. Besides, previous studies usually train a single model for each action class, but our method can discover a set of motion atoms and phrases. Thus, they are more effective to handle large intra-class variations than a single model.

Attribute and part based representations originated from object recognition [4, 22] and have been introduced to action recognition [14, 19, 25]. Liu *et al.* [14] define a set of action attributes and map the video data into attribute space. To deal with intra-class variation, they propose to use latent variables to indicate the presence of attributes given a video. Rohrbach *et al.* [19] propose to use simple cooking actions as attributes to recognize composite cooking activities. They use a script data approach to obtain the temporal information for cooking composite actions. Wang *et al.* [25] discover a set of mid-level cuboids called motionlet to represent video. Their cuboids are limited in temporal duration and not suitable for complex action recognition Our motion atom has the similar role as these motion attributes and parts in essence. However, our motion atoms are obtained through an unsupervised manner from training data, and we model temporal structure of multiple motion atoms to enhance their descriptive power.

AND/OR structure have been successfully used in object and action classification tasks [29, 3]. Amer *et al.* [3] propose an AND/OR model to unify multi-scale action detection and recognition in a principle framework. However our goal is different from theirs and we aim to mine a set of mid-level units to represent video, partially inspired by grouplet [29].

3. Unsupervised Discovery of Motion Atoms

To construct effective representations for complex actions, we first discover a set of motion atoms that capture the motion patterns in a short temporal scale. These atoms act

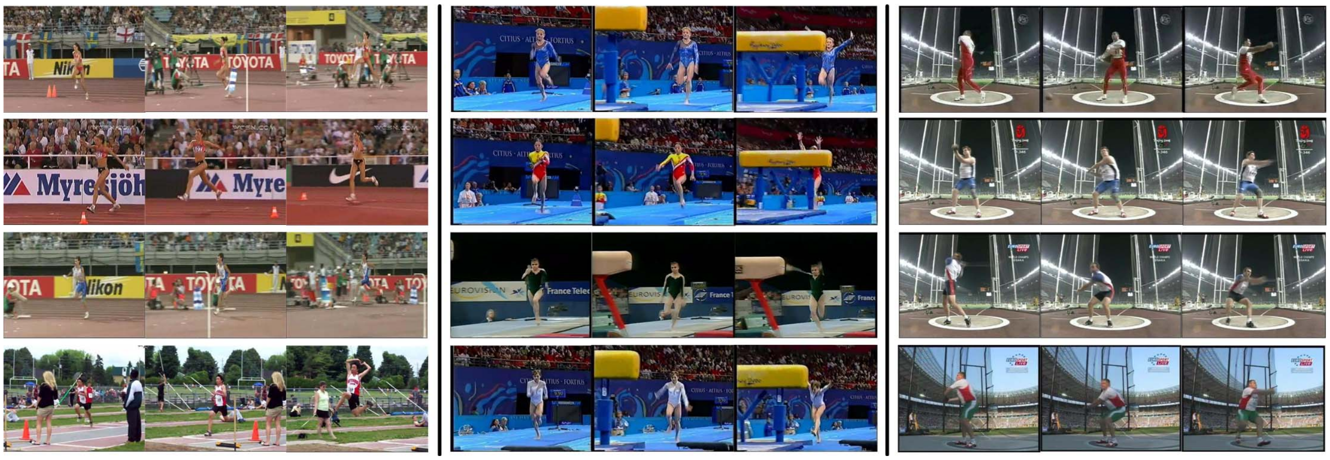

Figure 2. Some examples of clustering result, left: motion atom corresponds to running in complex action high-jump; middle: motion atom corresponds to running and opening arms for complex action gym-vault; right: motion atom corresponds to rolling in circles for complex action hammer throw.

as basic units for constructing more discriminative motion phrase in a longer scale. Given a set of training videos, our objective is to automatically discover a set of common motion patterns as motion atoms. The challenge comes from the facts that the number of possible segments extracted from videos is huge and the variation is large.

Motivated by a recent work on finding discriminative patches in images [22], we propose a discriminative clustering method for obtaining motion atoms. The whole process is shown in Algorithm 1. Note that our goal is different from [22]. They try to identify discriminative patches that are against the patches of the rest visual world. Thus, they need to use a set of natural world images to extract discriminative patches. Our main goal is to determine a large set of simple motion patterns, which are shared by many complex actions and can be used as basic units to represent complex actions. We need to make sure that the obtained atom set can cover different motion patterns occurring in various actions.

Initialization: The input to Algorithm 1 is a set of training videos $\mathcal{V} = \{V_i\}_{i=1}^N$. Note that we ignore the class label of each video. Firstly, we densely extract segments from each video. Due to the fact that each training video is a short clip that is approximately aligned in temporal dimension, we divide each video clip into k segments of equal duration with 50% overlaps. Then, we extract dense trajectory features [24] for each segment with default parameter settings. We resort to Bag of Visual Words (BoVW) method [28] to represent each video segments. Specifically, we choose four types of descriptors: HOG, HOF, MBHX, and MBHY [24]. For each type of descriptor, we construct a codebook of size K, and a histogram representation is obtained for each segment. In order to group segments, we need to define a similarity measure between segments. Given two segments $S_i = \{\mathbf{h}_i^m\}_{m=1}^4$ and $S_j = \{\mathbf{h}_j^m\}_{m=1}^4$, we define their similarity as follows:

Algorithm 1: Discovery of motion atoms.

```
Data: Discovery samples: $\mathcal{V} = \{V_i\}_{i=1}^N$.
Result: Motion atoms: $\mathcal{A} = \{A_i\}_{i=1}^M$.
- $\mathcal{S} \leftarrow$ DenseSampling($\mathcal{V}$).
- $T \leftarrow$ CalculateSimilarity($\mathcal{S}$).
- $\mathcal{A} \leftarrow$ APCluster($T$).
while $t \leq MAX$ do
    foreach cluster $A_i$ with size($A_i$) $> \tau$ do
        TrainSVM($A_i, \mathcal{V}$).
        FindTop($A_i, \mathcal{V}$).
    end
    CoverageCheck($\mathcal{A}, \mathcal{V}$).
    $t \leftarrow t + 1$.
end
- Return motion atoms: $\mathcal{A} = \{A_i\}_{i=1}^M$.
```

$$\mathrm{Sim}(S_i, S_j) = \sum_{m=1}^{4} \exp(-\mathcal{D}(\mathbf{h}_i^m, \mathbf{h}_j^m)), \tag{1}$$

where $\mathcal{D}(\mathbf{h}_i^m, \mathbf{h}_j^m)$ is the normalized χ^2 distance between two histograms:

$$\mathcal{D}(\mathbf{h}_i^m, \mathbf{h}_j^m) = \frac{1}{2M_m} \sum_{k=1}^{D} \frac{(\mathbf{h}_{i,k}^m - \mathbf{h}_{j,k}^m)^2}{\mathbf{h}_{i,k}^m + \mathbf{h}_{j,k}^m}, \tag{2}$$

where $\mathbf{h}_i^m$ denotes the histogram feature vector for the m-th feature channel of segment S_i, M_m is the mean distance for feature channel m over training samples.

With this similarity measure, we use Affinity Propagation (AP) to cluster segments [7]. AP is an exemplar based cluster algorithm whose input is a similarity matrix between samples. The only parameter in AP is the preference value. Due to large variance of action data, the preference parameter is set to be larger than the median similarity to ensure that the training segment is tightly clustered. We set a threshold to eliminate the small clusters with the number of segments less than τ.

Iterative Approach: Given clustering results, we firstly train a SVM for each cluster. The segments within the cluster are chosen as positive examples. We mine *hard negative examples* from other clusters. An segment is identified as hard negative example if its similarity with the current cluster is less than a threshold. Then, we run the SVM classifier on the training samples and detect its top m segments with the highest SVM scores. Finally, we check coverage percentage of current detection results and make sure that each training sample has at least $\frac{1}{2}k$ segments detected in the current result. Otherwise, we will randomly extract $\frac{1}{2}k$ segments from this training sample and cluster the new segments. The newly formed cluster will be added into the current results. We call this step as *Coverage Check* which ensures that the current results are sufficient to represent the whole training dataset. The whole process is running for a fixed number of iterations and some examples are shown in Figure 2.

Implementation Details: We divide each video into 5 segments with equal duration and we make sure each video has at least 3 segments contained in the selected clustering results. The codebook size for each descriptor is set to 1000. During the iterative process, the cluster is kept if it has at least 4 segments. We train SVM with χ^2-RBF kernels [30] for each cluster ($C = 0.01$), and detect top 10 segments from training samples. The iteration number is set as 5.

4. Mining Motion Phrases

Motion atoms are obtained by clustering short video segments. One atom usually corresponds to a simple motion pattern within a short temporal scale, and may occur in different classes of complex actions. These facts limit the discriminative ability of motion atoms in classifying complex actions. To circumvent this problem, we make use of these atoms as basic units to construct motion phrase with a longer scale.

For action classification task, motion phrases are expected to have the following properties:

- *Descriptive property*: Each phrase should be a temporal composite of highly related motion atoms. It can capture both the motion and temporal structure information of these atoms. Meanwhile, to deal with motion speed variations, motion phrase needs to allow temporal displacement among its composite atoms.

- *Discriminative property*: To be effective in classification, motion phrases should exhibit different distributions among different classes. It is desirable that a motion phrase is highly related to a certain class of action. Thus it can discriminate one complex action class from others.

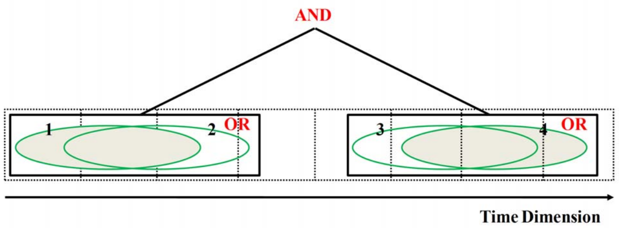

Figure 3. Illustration for motion phrase: motion phrase is an AND/OR structure over a set of atom units, which are indicated by ellipsoids.

- *Representative property*: Due to large variations among complex action videos, each motion phrase can only cover part of the action videos. Thus, we need to take account of the correlations between different phrases, and we wish to determine a set of motion phrases which convey enough motion patterns to handle the variations of complex actions.

Motion Phrase Definition: Based on the analysis above, we define motion phrase as an AND/OR structure on a set of motion atom units as shown in Figure 3. Similar AND/OR structures have been successfully used for image classification [29].

To begin with, we introduce some notations as follows. Each atom unit, denoted as $\Pi = (A, t, \sigma)$, refer to a motion atom A detected in the neighborhood of temporal anchor point t. The temporal extent of A in the neighborhood of t is expressed as a Gaussian distribution $\mathcal{N}(t'|t, \sigma)$. The response value v of an atom unit Π with respect to a given video V is defined as follows:

$$v(V, \Pi) = \max_{t' \in \Omega(t)} \text{Score}(\Phi(V, t'), A) \cdot \mathcal{N}(t'|t, \sigma), \tag{3}$$

where $\Phi(V, t')$ is the BoVW representation extracted from video V at location t', $\text{Score}(\Phi(V, t'), A)$ denotes the SVM output score of motion atom A, and $\Omega(t)$ is the neighborhood extent over t.

Based on these atom units, we construct motion phrases by AND/OR structure. We first apply *OR operation* over several atom units that have the same atom label and are located nearby (e.g. 1 and 2, 3 and 4 in Figure 3). The atom unit that has the strongest response is selected (e.g. 1 and 4 are selected in Figure 3). Then, we conduct *AND operation* over the selected atom units and choose the smallest response as motion phrase response. Thus the response value r of an motion phrase P with respect to a given video V:

$$r(V, P) = \min_{OR_i \in P} \max_{\Pi_j \in OR_i} v(V, \Pi_j), \tag{4}$$

where OR_i denote the OR operations in motion phrase P. The size of motion phrase is defined as the number of OR operations it includes (e.g. the size of atom phrase in Figure 3 is 2).

In essence, motion phrase representation is the temporal composite of multiple atomic motion units. The OR operation allows us to search for the best location for current motion atom, and makes it flexible to deal with the temporal displacement caused by motion speed variations. The AND operation incorporates temporal constraints among several motion atoms. Above all, motion phrase not only delivers motion information of each atom, but also encodes temporal structure among them. This structure representation can enhance the descriptive power and make it more discriminative for complex action classification.

Evaluation of Discriminative Ability: A motion phrase P is discriminative for c-th class of complex action if it is highly related with this class, but appears sparely in other action classes. We define its discriminative ability as follows:

$$\mathrm{Dis}(P,c) = \mathrm{Rep}(P,c) - \max_{c_i \in C-c} \mathrm{Rep}(P,c_i), \qquad (5)$$

where C represents all the classes and $\mathrm{Rep}(P,c)$ denotes the representative ability of P with respect to class c, whose high value indicates strong correlation with the class c:

$$\mathrm{Rep}(P,c) = \frac{\sum_{i \in S(P,c)} r(V_i, P)}{|S(P,c)|}, \qquad (6)$$

where $r(V_i, P)$ denotes the response value of motion phrase P in video V_i (Equation (4)), $S(P,c)$ is a set of videos defined as:

$$S(P,c) = \{i | Class(V_i) = c \wedge V_i \in top(P)\}, \qquad (7)$$

where $Class(V_i)$ is the class label of video V_i and $top(P)$ represents a set of videos that have the highest response values for motion phrase P. Due to the large variance among action videos, a single motion phrase could obtain strong value only on part of the videos of certain class. Thus, we evaluate its representative ability using the subset of videos of this class.

Mining Motion Phrase: Given a training video set $\mathcal{V} = \{V_i\}_{i=1}^N$ with class label $\mathcal{Y} = \{y_i\}_{i=1}^N$ and a set of motion atoms $\mathcal{A} = \{A_i\}_{i=1}^M$, our goal is to find a set of motion phrases $\mathcal{P} = \{P_i\}_{i=1}^K$for complex action classes. Given the class c, for each individual motion phrase, we want each motion phrase to have high discriminative and representative ability with current class c. Meanwhile, for a set of motion phrases $\mathcal{P} = \{P_i\}_{i=1}^K$, we need consider the correlation among them and define its set representative power with respect to class c as follows:

$$\mathrm{RepSet}(\mathcal{P}, c) = \frac{1}{T_c} | \cup_{P_i \in \mathcal{P}} S(P_i, c)|, \qquad (8)$$

where T_c is the total number of training samples for class c, $S(P_i, c)$ is the video set defined in Equation (7). Intuitively, considering the correlations of different motion phrases, we can eliminate the redundance and ensure the diversity of mining results. Thus, the set of motion phrases is able to cover the complexity of action videos.

Algorithm 2: Mining motion phrases

```
Data: videos: $\mathcal{V} = \{V_i, y_i\}_{i=1}^N$, motion atoms:
      $\mathcal{A} = \{A_i\}_{i=1}^M$.
Result: Motion phrases: $\mathcal{P} = \{P_i\}_{i=1}^K$.
- Compute response value for each atom unit on all videos
$v(V, \Pi)$ defined by Equation (3).
foreach class $c$ do
    1. Select a subset of atom units (see Algorithm 3).
    2. Merge continuous atom units into 1-motion phrase
    $\mathcal{P}_1^c$.
    while $maxsize < MAX$ do
        a. Generate candidate $s$-motion phrase based on
        $(s-1)$-motion phrase.
        b. Select a subset of motion phrases $\mathcal{P}_s^c$ (see
        Algorithm 3).
    end
    3. Remove the motion phrase whose $\mathrm{Dis}(P, c) < \tau$.
end
- Return motion phrases: $\mathcal{P} = \cup_{c,s} \mathcal{P}_s^c$.
```

The main challenge comes from the fact that the possible combination atom units that form a motion phrase is huge. Assuming a video with k segments and the size of motion atoms is M, there are $M \times k$ possible atom units. The total number of possible motion phrase is $\mathcal{O}(2^{M \times k})$. However, it is impossible to evaluate all possible configurations for motion phrase. We develop an efficient phrase mining method, inspired by Apriori algorithm [2]. If a phrase of size s has a high representative ability for action class c (Equation (6)), then any $(s-1)$-atom phrase by eliminating one motion atom should also have a high representative ability as well. This observation allows us to obtain a representative phrase of size s from a set of phrases of size $s-1$. The mining algorithm is shown in Algorithm 2. Finally, we eliminate some motion phrase of low discriminative ability with a threshold τ.

During each iteration of Algorithm 2, due to the huge number of possible motion phrases, we need to identify a subset of motion phrases. Ideally, both the individual and set representative ability should be as high as possible. We design a method to select effective phrases in a greedy way. Details are shown in Algorithm 3. In each iteration, we determine a motion phrase with high individual representative power, that meanwhile increases the set representative power the most.

Implementation Details: In current implementation, we fix the parameter σ_i for each atom unit as 0.5. During

Algorithm 3: Selecting a subset of motion phrases.

```
Data: motion phrases candidates $\mathcal{P} = \{P_i\}_{i=1}^{L}$, class: $c$,
      number: $K_c$.
Result: selected motion phrases: $\mathcal{P}^* = \{P_i\}_{i=1}^{K_c}$.
- Compute the representative ability of each motion phrase
$\mathrm{Rep}(P, c)$ defined in Equation (6).
- Initialization: $n \leftarrow 0, \mathcal{P}^* \leftarrow \emptyset$.
while $n < K_c$ do
    1. For each remaining motion phrase $P$, compute:
    $\triangle\mathrm{RepSet}(P, c) = \mathrm{RepSet}(\mathcal{P} \cup P, c) - \mathrm{RepSet}(\mathcal{P}, c)$,
    where $\mathrm{RepSet}(\mathcal{P}, c)$ is defined in Equation (8).
    2. Choose the motion phrase:
    $P^* \leftarrow \arg\max_P[\mathrm{Rep}(P, c) + \triangle\mathrm{RepSet}(P, c)]$.
    3. Update: $n \leftarrow n + 1, \mathcal{P}^* \leftarrow \mathcal{P}^* \cup \{P^*\}$
end
- Return motion phrases: $\mathcal{P}^*$.
```

the mining process, for each motion phrase, we consider top 40 videos with highest response value (i.e. $|top(P)| = 40$). For each class, we mine nearly the same number of motion phrases.

5. Recognition with Motion Atoms and Phrases

Motion atoms and phrases can be regarded as mid-level units for representing complex action. In this section, we make use of them to construct a mid-level representation of input action video, and develop a method to classify complex actions with this new representation.

Specifically, for each motion atom A, we define a special motion phrase, in which there is only one atom unit $(A, 0, +\infty)$. We call this special motion phrase as 0-*motion phrase*. Then, with a set motion phrase $\mathcal{P} = \{P_i\}_{i=1}^{K}$ whose sizes range from 0 to MAX, we represent each video V by an *activation vector* $f = [r_1, \cdots, r_K]$, where r_i is the response value of motion phrase P_i with respect to video V. We use this activation vector as a representation for video data. For classifier, we resort to linear SVM implemented by LIBSVM [5], and adopt one-vs-all scheme for multi-class classification.

6. Experiments

We evaluate the effectiveness of motion atom and phrase on two complex action datasets: Olympic Sports dataset [15] and UCF50 dataset [18]. The Olympic sports dataset has 16 sports classes, and there are 649 videos for training and 134 for testing. We conduct experiments according to the settings released on its website[1]. The final performance is evaluated by computing the average precision (AP) for each action class and the mean AP over all the classes (mAP). UCF50 [18] is a large dataset for complex action recognition. It includes 50 action classes with 6,618 videos, and each action class is divided into 25 groups with at least 100 videos for each class[2]. We adopt the Leave-One-Group-Out-Cross-Validation scheme and report the average accuracy for multi-class classification.

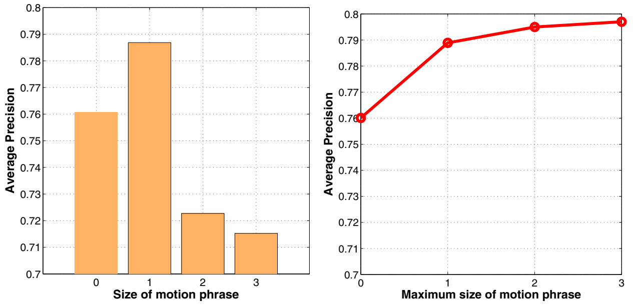

Figure 4. Left: performance of motion phrase for different sizes on the Olympic Sports dataset. Right: performance trend of varying maximum size for motion phrase on the Olympic Sports dataset.

Size of Motion Phrases: We examine the performance of motion phrases with different sizes on the Olympic Sports dataset and the results are shown in Figure 4. Firstly, we observe that 1-motion phrase achieves better results than other phrases. 1-motion phrases are mined for high discriminative and representative power, and thus their performance is better than 0-motion phrases (motion atoms), whose discriminative power is relatively low. Secondly, we notice that the mAPs of 2-motion phrases and 3-motion phrases are lower than the mAPs of 1-motion phrases and 0-motion phrases. This may be due to the large variations of video data, and the number of mined 2-motion phrases and 3-motion phrases is much smaller than the other two. Although motion phrases of large size are more discriminative than others, they only cover a small part of the video data. Thus their representative power may be relatively low. Besides, the information conveyed by large motion phrases has been partly contained in the motion phrases of smaller size.

We combine the representation of motion phrases with different sizes and the performance is shown in the right of Figure 4. We see that the performance increases apparently in using motion phrases of size from 0 to 2. But there is only slight improvement when including motion phrases of size 3. These results indicate that the maximum size 2 may contain enough information for complex action recognition. Therefore, in the remaining discussions, we fix the maximum size of motion phrases as 2.

Effectiveness of Representation: In this part, we examine the effectiveness of motion atom and phrase as mid-level representation. We compare the mid-level representation with low-level features. Specifically, we use the same descriptor (i.e. HOG, HOF, MBHX, MBHY) and codebook size (i.e. 1000) to construct a bag of visual words repre-

[1] http://vision.stanford.edu/Datasets/OlympicSports/

[2] http://crcv.ucf.edu/data/UCF50.php

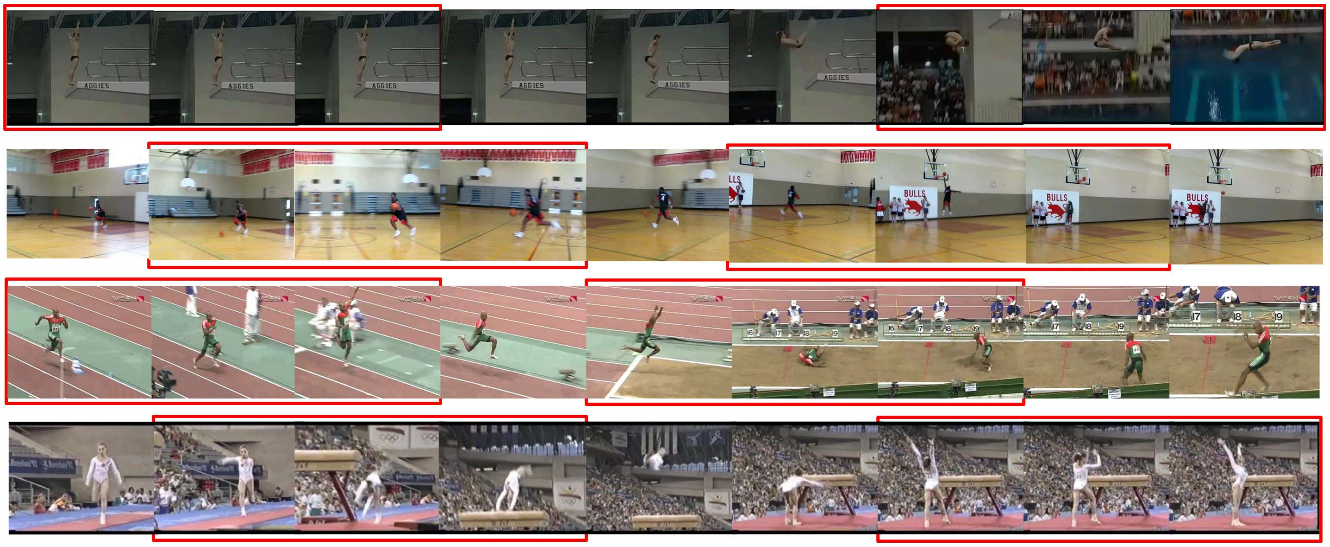

Figure 5. Some examples of 2-motion phrase for complex actions: diving platform, basketball-layup, triple-jump and gym-vault. Motion phrase can automatically locate temporal composites of multiple motion atoms (indicated by red boxes) in complex actions.

Dataset	Olympic Sports	UCF50
Low-level Features (linear)	58.1 %	66.6 %
Low-level Features (kernel)	70.1 %	77.4 %
Motion Atoms	76.1%	82.5%
Motion Atoms and Phrases	79.5%	84.0%
Combine All	**84.9%**	**85.7%**

Table 1. Performance comparison of motion atom and phrase with low-level features on the Olympic Sports dataset and the UCF50 dataset. Combine all indicates the combination of low-level features with motion atoms and phrases, with which we obtain state-of-the-art performances on two datasets.

Methods	Performance (mAP)
Laptev *et al.* [12]	58.2%
Niebles *et al.* [15] (from [23])	62.5%
Tang *et al.* [23]	66.8%
Liu *et al.* [14]	74.4%
Wang *et al.* [24]	77.2%
Our best result	**84.9%**

Table 2. Comparison of our methods with others on the Olympic Sports Dataset.

Methods	Performance (accuracy)
Laptev *et al.* [12] (from [20])	47.9%
Sadanand *et al.* [20]	57.9%
Kliper-Gross *et al.* [11]	72.6%
Reddy *et al.* [18]	76.9%
Wang *et al.* [25]	78.4%
Wang *et al.* [24]	85.6%
Our best result	**85.7%**

Table 3. Comparison of our methods with others on the UCF50 Dataset.

sentation for low-level features. We choose two kinds of classifier for these low level features: linear SVM and χ^2-RBF kernel SVM. Note that for motion atoms and phrases, we only use linear SVM. The results are summarized in Table 1. We can find that the motion atom based mid-level representations achieve better performance than low-level features on both datasets. We also notice that the performance of low-level features largely depends on the classifiers used, and χ^2-RBF kernel SVM performs better than linear SVM. However, motion atoms can achieve good results just with linear SVM. The combination of motion atoms and phrases can further improve the recognition results. These results partly verify the importance of incorporating temporal structure information among motion atoms. Finally, we combine motion atoms and phrases with low-level features, and obtain the state-of-the-art performances on both datasets. The results show that our mid-level representations are also complementary to low-level features.

Comparison with Other Methods: We compare motion atoms and phrases with other methods on both datasets, and the results are shown in Table 2 and Table 3. On the Olympic Sports dataset, [15, 23] use latent variable to model decomposition of atomic actions. They usually train a single model for each complex action. Our mid-level representation aims to find multiple motion atoms and phrases, and each representation covers a subset of videos. These results indicate the effectiveness of multiple representations (mixture representations) to handle large variance in video data, which has been verified in object detection [6, 10]. In [14], the authors use attribute representation, where the attributes are specified in advance, and we find motion atoms and phrases learned from training data are more flexible and effective.

For the UCF50 dataset, [12, 11] focus on designing new low-level features: STIP+HOG/HOF, Motion Interchange

Pattern. From the results of Table 3, we see that our motion atoms and phrases outperform these low-level features on UCF50 dataset. This can be ascribed to the fact that these low-level features only capture motion and appearance information in a local region which limits their descriptive power. Action Bank [20] is a global template for complex action, which cannot deal with large variance of video data well. Unlike action bank, our motion atom and phrase correspond to middle-level "parts" of the action, similar to the mid-level motionlet [25]. They make good tradeoff between low-level features and global template, and is more effective for representing complex action videos. However, motionlets are limited in temporal domain and lack descriptive power for longer temporal structure.

Compared with the latest paper [24], our motion atom and phrase use less descriptor and smaller codebook size. Besides, we only use linear SVM and do not incorporate structure information with spatial-temporal pyramids. Our results outperform on the Olympic Sports dataset and are comparable on the UCF50 dataset. This indicates that motion atom and phrase is effective for action classification, especially for complex action classes with longer temporal scale.

Visualization: We show some examples of motion atoms and phrases in Figure 2 and Figure 5 respectively. From the results, one can see that the proposed discriminative clustering method can group segments with similar motion and appearance. Each cluster usually corresponds to one atomic action such as running, rolling, and landing. Motion phrase consists of a sequence of motion atoms. As shown in the examples of Figure 5, motion phrase can discover waiting and diving for diving-platform, running and layup for basketball-layup, running and jumping for triple jumping, and running and landing for vault.

7. Conclusion

We propose motion atom and phrase for representing and recognizing complex actions. Motion atom describes simple motion pattern in a short temporal scale, and motion phrase encodes temporal structure of multiple atoms in a longer scale. Both of them can be seen as mid-level "parts" of complex actions. We evaluate the performance of the proposed method on two complex action datasets: Olympic Sports dataset and UCF50 dataset. From the experimental results, we see that motion atoms and phrases are effective representations and outperform several recently published low-level features and complex models.

References

[1] J. K. Aggarwal and M. S. Ryoo. Human activity analysis: A review. *ACM Comput. Surv.*, 43(3):16, 2011.

[2] R. Agrawal and R. Srikant. Fast algorithms for mining association rules in large databases. In *VLDB*, 1994.

[3] M. R. Amer, D. Xie, M. Zhao, S. Todorovic, and S. C. Zhu. Cost-sensitive top-down/bottom-up inference for multiscale activity recognition. In *ECCV*, pages 187–200, 2012.

[4] T. L. Berg, A. C. Berg, and J. Shih. Automatic attribute discovery and characterization from noisy web data. In *ECCV*, 2010.

[5] C.-C. Chang and C.-J. Lin. Libsvm: A library for support vector machines. *ACM TIST*, 2(3):27, 2011.

[6] P. F. Felzenszwalb, R. B. Girshick, D. A. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *TPAMI*, 32(9):1627–1645, 2010.

[7] B. J. Frey and D. Dueck. Clustering by passing messages between data points. *Science*, 315:972–976, 2007.

[8] A. Gaidon, Z. Harchaoui, and C. Schmid. Actom sequence models for efficient action detection. In *CVPR*, 2011.

[9] L. Gorelick, M. Blank, E. Shechtman, M. Irani, and R. Basri. Actions as space-time shapes. *TPAMI*, 29(12):2247–2253, 2007.

[10] C. Gu, P. A. Arbeláez, Y. Lin, K. Yu, and J. Malik. Multi-component models for object detection. In *ECCV*, 2012.

[11] O. Kliper-Gross, Y. Gurovich, T. Hassner, and L. Wolf. Motion interchange patterns for action recognition in unconstrained videos. In *ECCV*, 2012.

[12] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld. Learning realistic human actions from movies. In *CVPR*, 2008.

[13] B. Laxton, J. Lim, and D. J. Kriegman. Leveraging temporal, contextual and ordering constraints for recognizing complex activities in video. In *CVPR*, 2007.

[14] J. Liu, B. Kuipers, and S. Savarese. Recognizing human actions by attributes. In *CVPR*, 2011.

[15] J. C. Niebles, C.-W. Chen, and F.-F. Li. Modeling temporal structure of decomposable motion segments for activity classification. In *ECCV*, 2010.

[16] N. Oliver, B. Rosario, and A. Pentland. A bayesian computer vision system for modeling human interactions. *TPAMI*, 2000.

[17] X. Peng, Y. Qiao, Q. Peng, and X. Qi. Exploring motion boundary based sampling and spatial-temporal context descriptors for action recognition. In *BMVC*, 2013.

[18] K. K. Reddy and M. Shah. Recognizing 50 human action categories of web videos. *MVAP*, 2012.

[19] M. Rohrbach, M. Regneri, M. Andriluka, S. Amin, M. Pinkal, and B. Schiele. Script data for attribute-based recognition of composite activities. In *ECCV*, 2012.

[20] S. Sadanand and J. J. Corso. Action bank: A high-level representation of activity in video. In *CVPR*, 2012.

[21] C. Schüldt, I. Laptev, and B. Caputo. Recognizing human actions: A local svm approach. In *ICPR*, 2004.

[22] S. Singh, A. Gupta, and A. A. Efros. Unsupervised discovery of mid-level discriminative patches. In *ECCV*, 2012.

[23] K. Tang, F.-F. Li, and D. Koller. Learning latent temporal structure for complex event detection. In *CVPR*, 2012.

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

[25] L. Wang, Y. Qiao, and X. Tang. Motionlets: Mid-level 3d parts for human motion recognition. In *CVPR*, pages 2674–2681, 2013.

[26] L. Wang, Y. Qiao, and X. Tang. Latent hierarchical model of temporal structure for complex activity classification. *TIP*, to appear.

[27] S. B. Wang, A. Quattoni, L.-P. Morency, D. Demirdjian, and T. Darrell. Hidden conditional random fields for gesture recognition. In *CVPR*, 2006.

[28] X. Wang, L. Wang, and Y. Qiao. A comparative study of encoding, pooling and normalization methods for action recognition. In *ACCV*, pages 572–585, 2012.

[29] B. Yao and F.-F. Li. Grouplet: A structured image representation for recognizing human and object interactions. In *CVPR*, 2010.

[30] J. Zhang, M. Marszalek, S. Lazebnik, and C. Schmid. Local features and kernels for classification of texture and object categories: A comprehensive study. *IJCV*, 73(2):213–238, 2007.