Nearest Atom of Local Spatio-temporal Features for Action Recognition

Chunhai HU, Changde FAN*, Bin LIU

Hebei Key Laboratory of Measurement Technology and Instrumentation, Yanshan University, Qinhuangdao 066004, China

Abstract

Motivated by the success of dictionary learning based classification works, we propose a nearest neighbour like method to recognize an action. First, local spatio-temporal features are extracted. Then an over-completed discriminative dictionary is learned by K-SVD for each class. The query video is labeled by counting the number of nearest atom of local features on a sub dictionary. Different from the SRC (sparse representation based classification) methods, we compute the angle between the atoms and a local feature to find the nearest atom so as to recognize an action instead of a sophisticated sparse coding stage. Experimental results on Facial and Weizmann public datasets demonstrate the validity of the proposed framework.

Keywords: Local Spatio-temporal Features; Action Recognition; Dictionary Learning; Nearest Atom

1 Introduction

Action recognition has been researched for a long time in computer vision. Over the past few years, it has drawn much attention with many important applications including human-computer interaction, video surveillance, content-based video search and health care [1]. Action recognition starts with a powerful model of the video sequence. A typical model that has been used extensively is the BoW (bag of words) model [2-4], whose procedure is usually: feature detection and description, learning a codebook, action recognition. The codebook is learned by k-means. K-means will lead to information lose due to vector quantization. To address this problem, Zhu [5] introduced the sparse representation theory for action recognition. They learned an over-completed dictionary using local spatio-temporal features and adopted a max pooling strategy to represent the video. Other works [6, 7] also learn a dictionary for action recognition.

In this work, we develop a general framework for human action recognition. The method is simple and fast. After the extraction of the local spatio-temporal features, we learn a redundancy dictionary for each class and construct a concatenated dictionary for recognition. The main

*Project supported by the Nature Science Foundation of Hebei Province, China (No. F2011203117).
*Corresponding author.

Email address: changdefan@yeah.net (Changde FAN).
contribution of this paper lies in that we propose a nearest neighbour like method named nearest atom of local spatio-temporal features for action recognition. A nearest atom is defined as the atom who has the smallest angle with a local feature. Instead of using the sophisticated OMP method to find the sparse representation of the local spatio-temporal features, we recognize an action according to the distribution of the nearest atom of local features over the concatenated dictionary.

The remainder of the paper is organized as follows. Previous work is discussed in Section 2. Section 3 gives a detailed description of the proposed recognition framework. We present experimental validation in Section 4 and conclude the paper in Section 5.

2 Related Work

Action recognition is a hot and tough issue in computer vision. We can roughly divide these works into two groups: generative models and discriminative models. Generative models, like hidden markov model (HMM) [8] and dynamic bayesian network (DBN) [9], directly simulate the generation process of the video sequences. They specify a joint probability distribution over observation and label sequences. Discriminative models learn the posterior probability distribution of unobserved variable on an observed variable, such as support vector machines (SVMs) [3] and relevance vector machines (RVMs) [10]. For a simple classification task, discriminative models can yield superior performance [1]. In addition, there are some other classification algorithms like tensor subspace learning method [11], the K-NN algorithm [12]. Refer to [1] for a comprehensive review about computer vision based motion recognition.

Learning a discriminative dictionary is an important issue for our recognition framework. There are many methods to construct a dictionary. Engan et al. [13] proposed an dictionary training algorithm named MOD. It takes the derivative of the representation mean square error equation with respect to dictionary to update the dictionary. KSVD [14] is a mature approach to learn an over-completed dictionary. It iteratively performs sparse coding and dictionary update at every iteration. Mairal et. al [15] proposed an online dictionary learning method which can be perfectly suitable for large scale data. In this work, we use KSVD to learn the dictionary. We also test the online method and it achieves a similar result.

Note that our work is different from the SRC based method in that after learning a discriminative dictionary, we classify a video sequence by counting the number of nearest atom of local spatio-temporal features instead of computing the sparse representation complicatedly.

3 Nearest Atom Based Action Recognition System

3.1 Local spatial-temporal features

As discussed in [16], local features are more robust to noise and occlusion, and possibly invariant to rotation and scale. In this work, we use the method proposed in [16] to extract local spatio-temporal features of a video sequence. The response function is defined as:

\[ R = (I \ast g \ast h_{ev})^2 + (I \ast g \ast h_{od})^2 \]  

(1)
Where $g(x, y, \sigma)$ is the spatial domain gaussian smooth kernel. $h_{ev}$ and $h_{od}$ are a quadrature pair of gabor filters applied along temporal direction and defined as follows:

$$
\begin{align*}
    h_{ev}(t, \tau, \omega) &= -\cos(2 \pi t \omega) - t^2 / 2 \\
    h_{od}(t, \tau, \omega) &= -\sin(2 \pi t \omega) - t^2 / 2
\end{align*}
$$

Interest points are detected at the local maximum of the response function. Small video patches are extracted around the interest points to represent the video.

### 3.2 Over-completed dictionary learning

Assuming a set of local spatial temporal features as $T = [t_1, t_2, \cdots, t_n] \in \mathbb{R}^{m \times n}$, the problem of dictionary learning can be formulated as the following optimization problem:

$$
\min_{D_k, X} \| T - D_k X \|_F
$$

Given a tolerant error $\varepsilon$, the optimization problem can be rewritten as:

$$
\min_{D_k, X} \| x_i \|_0 \quad \text{subject to} \quad \| T - D_k X \|_F^2 \leq \varepsilon
$$

The $l_0$ term in the expression ensures the sparsity of the solution and the Frobenius norm serves as the fidelity term.

K-SVD is a more mature approach to learn a dictionary. It solves the problem by iteratively performing sparse coding and dictionary update. Refer to [14] for a detailed description of the K-SVD algorithm.

### 3.3 Nearest atom based action recognition method

In the query video recognition stage, we take the concatenated dictionary as $D = [D_1 D_2 \cdots D_k]$, where $D_i$ is the class specific dictionary, $k$ is the number of action classes. Assume the extracted local spatio-temporal features of the query video as $Y = [y_1 y_2 \cdots y_q] \in \mathbb{R}^{m \times q}$, where $q$ is the number of local features. Rewritten the concatenated dictionary as the atoms containing form:

$$
D = [d_1 d_2 \cdots d_a], \quad a \text{ is the number of atoms.}
$$

Assume the angle between a local feature $y_i$ and an atom $d_j$ is $\theta_j$, we can get,

$$
\| T_j \|_2 \times y_i = \| y_i \|_2 \times \cos \theta_j
$$

Where $j = 1, 2, \cdots a$, $i = 1, 2, \cdots q$, $q$ is the number of local features. The angle between a local feature and the atom is:

$$
\theta_j = \arccos \left( \frac{d_j^T \times y_i}{\| y_i \|_2^2} \right),
$$

the angle $\theta$ between a fixed $y_i$ and the nearest atom corresponding to it is:

$$
\theta_j = \begin{cases} 
\theta_j & 0 \leq \theta_j \leq \pi/2 \\
\pi - \theta_j & \pi/2 < \theta_j < \pi 
\end{cases}
$$

For every $y_i$, the coefficient $x_i$ is $\| y_i \|_2 \times \cos \theta_i$. 
From equation Eq. (5)-Eq. (7), we can get the coefficient matrix of all local spatio-temporal features

\[ X = [x_1, x_2, \cdots, x_n] \]  

(8)

Where \( x_i \) is of the form \([0, 0, \cdots, \|y_i\|_2 \times \cos \theta, 0, 0]\).

Local spatio-temporal features of the same action have more nonzero coefficients over its corresponding sub-dictionary. We define \( \varphi \) as an indicator function that counts the number of local spatio-temporal features that have nonzero coefficient. In this work, we also propose another classification criterion that the indicator function just counts the number of local features having positive coefficient. We name this indicator function as \( \varphi_+ \). Then the label of the query video will be classified as:

\[ \text{label} = \arg \max_c \varphi_+(X_c) \]  

(9)

4 Experimental Evaluation

In this section, we perform a series of experiments to validate the performance of the proposed method. We first compare our two classification criteria NA-PC and NA-AC. Then a detailed comparison with state of the art work on the two public datasets is given. All the experiments are implemented for many times and average results are presented. Note that, in our experiments, all local features are pre extracted to reduce the experiment running time.

4.1 Comparison of the two classification criteria

In this subsection, we test the effectiveness of the classification criterion that just counts the number of local spatio-temporal features that have positive coefficient. For notation convenience, we call the two classification criteria as NA-PC (positive coefficients) and NA-AC (all coefficients).

The performance of the two classification criteria is presented in Fig. 1 (tested on Facial public dataset). As can be seen, accuracies obtained by NA-PC has an improvement about 2% with respect to NA-AC. Hereafter, the result reported is achieved by the NA-PC criterion.

Fig. 1: Relative performance of the classification criterion
4.2 Facial dataset

The Facial expression dataset [16] contains six different emotions of two individuals under two different lighting setups. The actions are anger, disgust, fear, joy, sadness and surprise. Each action is performed eight times under different lighting, giving 192 video sequences of resolution 152*194. The dataset is divided into four subsets according to different conditions, i.e. illumination and person. One subset is used for training and the other three for testing.

A number of parameters need to be chosen properly during the experiment. One simple method is to use cross-validation. Following the suggestion of [16], we set \( \sigma = 2 \) and \( \tau = 3 \). In our experiment, the high dimensional local features are reduced to 100 dimensions by PCA. The size of the class specific dictionary is set to be 100*512 and the concatenated dictionary is 100*3072.

<table>
<thead>
<tr>
<th>Experimental setup</th>
<th>Methods</th>
<th>BoW <a href="%25">17</a></th>
<th>Guha <a href="%25">6</a></th>
<th>Our approaches(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>same subject, diff. illumination</td>
<td>BoW <a href="%25">17</a></td>
<td>89.6</td>
<td>93.7</td>
<td>99.17</td>
</tr>
<tr>
<td>diff. subject, same illumination</td>
<td>Guha <a href="%25">6</a></td>
<td>75.0</td>
<td>91.7</td>
<td>83.33</td>
</tr>
<tr>
<td>diff. subject, diff. illumination</td>
<td>Our approaches(%)</td>
<td>69.8</td>
<td>72.9</td>
<td>81.25</td>
</tr>
</tbody>
</table>

Table 1: Comparison over different experimental setup on Facial dataset

Table 1 compares our method with the other two under different experimental setups. We use one subset containing 48 videos for training. The three subset cases: different subject, different illumination, different subject and illumination is tested once at a time. From the table, we can see our approach showing significant improvement under different lighting setups (about 6% for same subject, 8% for different subject comparing to [6]). The confusion matrices of recognition results on the facial dataset are demonstrated in Fig. 2. As can be seen, the error mainly comes from anger, fear, sadness. The lack of structural information leads to the description of the video inefficiently. This is another issue of this paper to research next.

![Confusion matrices on the Facial dataset](image)

Fig. 2: Confusion matrices on the Facial dataset. From left to right, same subject and different illumination, different subject and same illumination, different subject and different illumination

In order to compare fairly with [7, 17, 18], we also use three subset for training (144 videos) and one subset for testing. In Table 2, we compare the best recognition results on this dataset achieved by our method with several state of the art methods. It can be seen that the recognition
accuracy is 97.40% for three training subsets, close to [7], higher than the other methods. When using one training subset, our method outperforms most of the examined methods.

<table>
<thead>
<tr>
<th>Methods</th>
<th>Accuracy(%)</th>
<th>Training subset</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dollar et al. [16]</td>
<td>78.13</td>
<td>1</td>
</tr>
<tr>
<td>Guha and Ward [6]</td>
<td>86.10</td>
<td>1</td>
</tr>
<tr>
<td>J.Wang et al. [7]</td>
<td>87.24</td>
<td>1</td>
</tr>
<tr>
<td><strong>Our approach</strong></td>
<td><strong>88.19</strong></td>
<td><strong>1</strong></td>
</tr>
<tr>
<td>Wong et al. [17]</td>
<td>87.50</td>
<td>3</td>
</tr>
<tr>
<td>Wong and Cipolla [18]</td>
<td>88.54</td>
<td>3</td>
</tr>
<tr>
<td>J.Wang et al. [7]</td>
<td>97.92</td>
<td>3</td>
</tr>
<tr>
<td><strong>Our approach</strong></td>
<td><strong>97.40</strong></td>
<td><strong>3</strong></td>
</tr>
</tbody>
</table>

### 4.3 Weizmann action dataset

The Weizmann action dataset is a benchmark dataset frequently used by many researchers. The later version of the dataset contains 10 classes. They are Bend, jumping jack (Jack), jump forward (Jump), jump in place (Pjump), Run, gallop sideways (Side), Skip, Walk, wave one hand (Wave1), wave two hands (Wave2). The dataset contains 90 video sequences of resolution 180*144. Each action is performed 10 times by 9 subjects.

We use leave-one-out validation protocol for this dataset. Each time we test one video sequence and use the remain for training. The result reported is the average of the 900 times action recognition. We empirically set $\sigma = 2$ and $\tau = 3$. The size of the extracted local features is $13^2*19$ and the size of the class specific dictionary is 100*200, the concatenated dictionary is 100*2000.

<table>
<thead>
<tr>
<th>Methods</th>
<th>No.of actions</th>
<th>Accuracy(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Thurau and Hlavac[23]</td>
<td>10</td>
<td>94.4</td>
</tr>
<tr>
<td>Junejo et al.[20]</td>
<td>9</td>
<td>95.3</td>
</tr>
<tr>
<td>Ali and Shah [22]</td>
<td>10</td>
<td>95.7</td>
</tr>
<tr>
<td>Guha and Ward [6]</td>
<td>10</td>
<td>98.9</td>
</tr>
<tr>
<td><strong>Our approach</strong></td>
<td><strong>10</strong></td>
<td><strong>99.1</strong></td>
</tr>
<tr>
<td>Wang and Mori [19]</td>
<td>9</td>
<td>100</td>
</tr>
<tr>
<td>Yeffet and Wolf [21]</td>
<td>9</td>
<td>100</td>
</tr>
</tbody>
</table>
Table 3 compares the best recognition results on the Weizmann dataset achieved by our method and several methods. From the table we can see that our method achieves a high recognition accuracy of 99.1% (8 misclassification out of 900). Our method outperforms most of the existing method. Note that, in the work of [19-21], they use an old version of the dataset that contains 9 classes.

The confusion matrices are presented in Fig. 3. As can be seen, our method makes a perfect classification for most of the action. The action bend and pjump are easier to be confused than the others. The reason may be that they both contain head movements, leading to some similarity of their local features. Given a specific recognition work, how to remove the similar features of different actions to improve recognition accuracy can be further researched.

Fig. 3: Confusion matrix on the Weizmann action dataset

5 Conclusion

In this paper, we propose a simple dictionary learning based action recognition method. We learn a class specific dictionary and concatenate them to form one dictionary. The test video is classified by counting the number of nonzero coefficients of local features on a sub-dictionary. Compared with BoW model, our method uses a dictionary framework instead of a codebook and performs better. Experiment results on Facial and Weizmann public dataset show that our algorithm outperforms most of the state-of-the-art results.

We use the K-SVD algorithm to learn a dictionary for one action in this paper. In order to improve classification performance, constructing an action classification oriented dictionary possibly be a future work. Ongoing work also includes using the spatio-temporal layout of the local features.

References


