Signal, Image and Video Processing (2018) 12:1551–1558
https://doi.org/10.1007/s11760-018-1311-z

ORIGINAL PAPER

Enhancing human action recognition via structural average curves analysis

Shichen Zeng[1] · Guoliang Lu[1] · Peng Yan[1]

Received: 2 May 2017 / Revised: 10 April 2018 / Accepted: 24 May 2018 / Published online: 29 May 2018

Abstract

Human action recognition typically requires a large amount of training samples, which is often expensive and time-consuming to create. In this paper, we present a novel approach for enhancing human actions with a limited number of samples via structural average curves analysis. Our approach first learns average sequences from each pair of video samples for every action class and then gather them with original video samples together to form a new training set. Action modeling and recognition are proposed to be performed with the resulting new set. Our technique was evaluated on four benchmarking datasets. Our classification results are superior to those obtained with the original training sets, which suggests that the proposed method can potentially be integrated with other approaches to further improve their recognition performances.

Keywords Action recognition · Limited training samples · Average sequences · Dynamic programming

1 Introduction

Recognition of single-person-oriented human actions is one of the central functions of modern computer systems which uses a camera tool for understanding humans with many applications such as surveillance, human–computer interaction (HCI) and motion retrieval.

1.1 Motivation

Over the last two decades, the majority of approaches (e.g., learning-based approaches including deep learning methods [1–5], instance matching-based methods [6–8] and sparse representation-based approaches [9,10]) focus on the classification of a query video after collecting a large number of (or at best a full/completed set of) labeled training samples. In other words, the underlying assumption of these methods is that a sufficient number of training samples must be available per class, which makes performances of these methods deteriorate when only a few training samples are available. But unfortunately, in some intelligent systems, the users often do not have sufficient training samples for action modeling. For instance, in vision-based surveillance applications such as safety protection and terrorism/crime deterrence, abnormal actions/activities are often defined as those rarely occurred in specific monitored sites, where the users cannot collect sufficient training samples for designing detectors [11]. To address this problem, some researchers attempted to collect extensive training samples by virtue of web data, e.g., [7,12]. This is, however, expensive and time-consuming to collect such volume of data in practical usages.

Another group of studies have taken a different way to perform action recognition only with a limited number of training samples. In particular, Seo and Milanfar [13] proposed a method of using a single example of an action as a query to find similar matches through measuring the likeness of a voxel to its surroundings, which is based on the computation of novel space-time descriptors from the query video; Rodriguez et al. [14] proposed a method based on a maximum average correlation height (MACH) filter which is capable of capturing intra-class variability by synthesizing a single-action MACH filter for a given action class; Neverova et al. [15] presented a training strategy to overcome

✉ Guoliang Lu
luguoliang@sdu.edu.cn

Shichen Zeng
shichenzeng@gmail.com

Peng Yan
yanpeng@sdu.edu.cn

[1] Key Laboratory of High-Efficiency and Clean Mechanical Manufacture of MOE, National Demonstration Center for Experimental Mechanical Engineering Education, School of Mechanical Engineering, Shandong University, Jinan 250061, China

the training problem when the number of labeled samples is not at *web-scale* like static image datasets by exploiting careful initialization of individual modalities and gradual fusion of modalities from the strongest to weakest cross-modality structure. These approaches mostly enhanced action recognition by improving the phase of classifiers' training, and thus their performances are still below those using a larger number of samples.

On the other hand, since various variations are often included in the training samples for many other applications, methods that apply structural analysis for the original data before processing have been popular, as seen from [16–18]. For example, Ahmadi et al. [19] proposed to recover accurate surgical workflow by averaging signals recorded in different operations of the same type taking advantages of an enhanced version of the dynamic time warp algorithm; Boudaoud et al. [20] presented a specific statistical tools for shape dispersion analysis based on a mean shape curve which is learned according to the degree of specific polynomial time functions; Morlini and Zani [21] proposed a new method to estimate the structural mean of a sample of curves by modifying the classical DTW, which has been demonstrated the priority on air pollutant data analysis; Xie et al. [22] introduced a method for clustering and averaging the tracks of people obtained in a multi-camera network using DTW and random sampling for optimizing the work cycles. In these works, the method of structural mean/average learning has been proven to be a promising strategy for enhancing model training/learning when handling training samples with varying amplitudes and phases/timings.

In the area of action recognition, Cherla et al. [18] have also proposed a fast and view-invariant average-template action model called "*action basis*" by the use of eigenanalysis from training sequences of different people, where the model shows great potentials to deal with action recognition with fewer training samples but it uses empirical eigenvalues to construct the average template that requires further quantitative investigation and experimental validation. Additionally, the action basis is only appropriate for unimodal classes where the samples are expected to gather around their class center. However, in complex action recognition tasks, unimodality is a very strong assumption that is not valid. Indeed, even the simplest action (e.g., walk) is rather different when performed by different persons, various views, scales, etc.

In this paper, in line with the methods of structural mean/average analysis, we focus on the further extension and validation of the average templates for action recognition when only a few training samples are available. Noticeably, different from [18] using PCA to generate the average template, the method proposed in this paper uses structural average curves analysis (SACA) to generate average templates by taking into account the variations of timing and amplitude between sample sequences per action class. Our method is complementary to those methods focusing on action recognition using limited samples, e.g., [13–15,18], and also could be potentially integrated with some of them for further improving their recognition performances.

1.2 Overview and contribution

As illustrated in Fig. 1, rather than directly using original training samples for action modeling and recognition, we propose to learn structural average samples by using SACA from these original samples, and then gather the resulting average samples with the original ones to form a new training set. Afterward, based on the new set, statistical distribution of human actions can be extracted using, e.g., bag-of-words (BoW). A query action can be finally recognized with conventional classification strategies such as ANN, SVM and *k*-NNC. The main contributions of this paper to the field are:

– SACA has been successfully applied to speech recognition. Here, we introduce SACA to the problem of action recognition. To the best of our knowledge, this is the first work that uses SACA to analyze human motions.
– Instead of using the original training samples for action modeling directly, we propose the average samples extracted by SACA together with the original ones to model human actions which takes into account the variations of timing and amplitude between video sequences in one action class.
– The proposed method of action modeling is successfully extended and validated on benchmarking datasets by comparing with the baselines relying on the original samples. In addition, it could potentially be integrated with the existing approaches for further improving their recognition performances.

The remainder of this paper is organized as follows. Section 2 details the SACA-based approach for the recognition of human actions. Experimental results are presented in Sect. 3, followed by discussions. Section 4 concludes this paper.

2 Methodology

2.1 Frame feature extraction

As the first step of video analysis, for a given query video F to be recognized which contains n frames, we first extract features in each frame and concatenate the resulting features to be a time-sequential set of features that can represent the video as, $F = \{f_i\}, i \in \{1, 2, \ldots, n\}$ where f_i corresponds to the features at ith frame. Here, it is worth mentioning that feature extraction plays an important role in video description

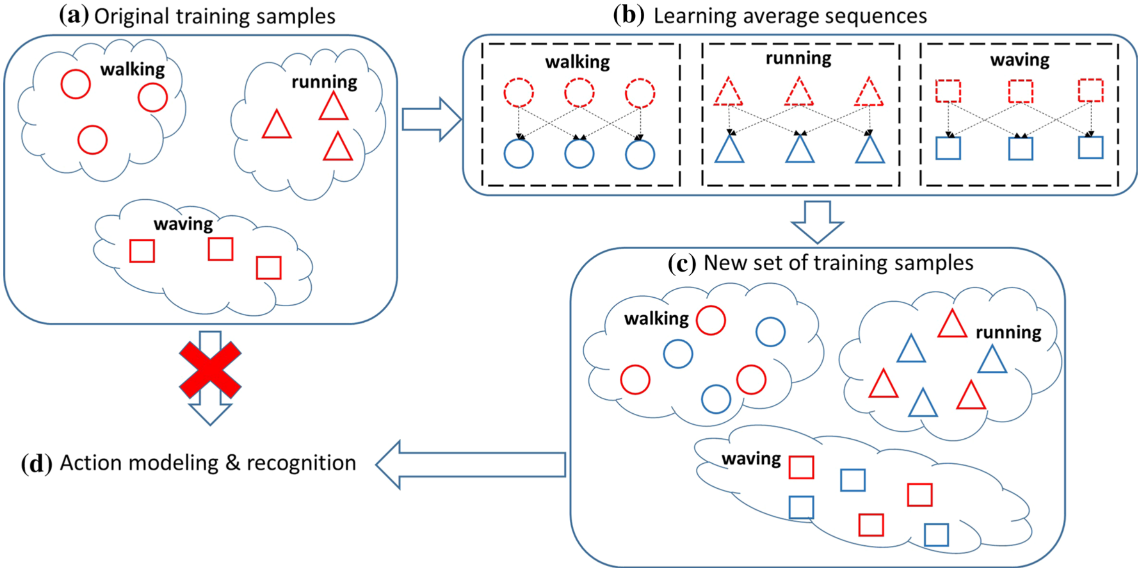

(a) Original training samples
walking
running
waving
(b) Learning average sequences
walking
running
waving
(c) New set of training samples
walking
running
waving
(d) Action modeling & recognition

Fig. 1 Given a training set containing a limited number of samples, rather than directly using them for action modeling and recognition, we propose to learn structural average sequences from each sample pairs in every class and then form a new set of training samples by taking together the original samples and the average sequences for action modeling and recognition

and thus takes a direct influence in next action recognition. While, further discussion on this procedure is beyond the scope of the paper because our focus here is to design an enhanced recognition framework using limited action samples. In other words, our expected framework does not rely on specific action features but would be workable for other features as long as they can describe the video effectively and informatively.

2.2 Structural average curves analysis for action modeling

2.2.1 Problem formulation

Let $\{F_i^c : i = 1, 2, \ldots, N\}$ be the collected training samples for action class c, where i in F_i^c indicates the index of ith action video and N is the number of video samples for this class. Suppose each observed frame $f_i(j)$ in a video F_i (i.e., $f_i(j) \subset F_i, 1 \le j \le n_i$ where n_i is the number of frames in F_i) fit the following model as,

$$f_i(j) = \mathcal{G}(t_{i,j}) + \varepsilon_{i,j}, \quad j = 1, 2, \ldots, n_i, \tag{1}$$

where $\mathcal{G}$ is a smoothing function, $t_{i,j} \in [0, 1]$ is the timings with any closed interval for the ith video sequence and $\{\varepsilon_{i,j}\}$ are the independent and identical distributed (*I.I.D.*) errors with zero mean, i.e., $\mathsf{E}[\varepsilon_{i,j}] = 0$.

The problem of learning averaging sequences is equivalent to estimating the smoothing function $\mathcal{G}$. When all video samples in the class have the same number of frames, i.e., $\forall i, n_i = n$, the expectation of $f_i(j)$ can be given by

$$\mathsf{E}[f_i(j)] = \mathcal{G}(t_{i,j}) + \mathsf{E}[\varepsilon_{i,j}] = \mathcal{G}(t_{i,j}). \tag{2}$$

Assuming ergodicity of i for all samples, i.e., $i \in [1, \ldots, m]$ in each frame, we can estimate each element $g(j)$ in $\mathcal{G}$ approximately by the law of large numbers as a sample mean as

$$g(j) \simeq \overline{f_i}(j) = \frac{1}{m} \sum_{i=1}^{m} f_i(j). \tag{3}$$

This approach, however, does not take into account for timing variations but only for amplitude variations. In real-life scenarios, action videos are often observed with a greatly different number of video frames because of different performing paces/intensities between individuals or sometimes even in the same individual. In fact, the timing variation is more common in automatic speech recognition where the processed speech sequences are often varying in time or speed [23,24]. To address this issue, an intuitive and natural alternative is to find the best match between every video sample f_i and an average sequence candidate $\mathcal{G} = \{g(j') : j' = 1, 2, \ldots, m\}$ by alignments $\mathcal{W}$ with respect to minimizing a cost function using an accumulated error, as

$$\inf_{\mathcal{W}} \sum_{i=1}^{n_i} \sum_{(j,j') \in \mathcal{W}} ||f_i(j) - g(j')||, \tag{4}$$

where $||\cdot||$ is a distance metric. Thanks to the dynamic programming, we can obtain $\mathcal{W} = \{(j, j')\}$ as a warping path connecting $(1, 1)$ and (n_i, m). Now, the problem addressed in this paper is how to learn structural average sequences from $\mathcal{W} = \{(j, j')\}$. The following section gives the procedures.

2.2.2 Averaging sequences

Sequential optimize Eq. (4) for each average sequence candidate is extremely time-consuming or even impossible. Therefore, some studies (see [19–22] for example) solve this problem on the basis of a structural averaging analysis. Motivated by these works, given two arbitrary action video samples $F = \{f(1), f(2), \ldots, f(n)\}$ and $F' = \{f'(1), f'(2), \ldots, f'(n')\}$, we learn the structural average sequences as follows:

- Step 1: Compute the distances for all frame pairs between F and F' (i.e., $\{(f(i), f'(j)) : i = 1, 2, \ldots, n; j = 1, 2, \ldots, n'\}$) to form a two-dimensional square lattice, and then take the optimal warping path $\mathcal{W} = \{w(k) \rightarrow (i(k), j(k)) : k = 1, 2, \ldots, K; i(1) = j(1) = 1; i(K) = n, j(K) = n'\}$ from the resulting square lattice using dynamic programming with respect to the cost function in Eq. (4);
- Step 2: The length K of obtained warping path $\mathcal{W}$ contains a different number of timings (or, in other words, sampling rates). We then normalize $\mathcal{W}$ to be a common timing $\overline{K}$ by the interpolation and averaging operations where the common timing is produced by averaging the timings of F and F', i.e., $\overline{K} = (n + n')/2$;
- Step 3: The normalized warping path $\mathcal{U} = \{u(k) : k = 1, 2, \ldots, \overline{K}\}$ indicates the best matching pairs between the two video sequences (as shown in Fig. 2).[1] We finally construct the average sequence $\mathcal{F}$ as

$$\begin{aligned} &\mathcal{F} = \{f^c(k) : k = 1, 2, \ldots, \overline{K}\}, \\ &f^c(k) = (f(\mathcal{U}^{-}(k) + f'(\mathcal{U}^{-}(k)))/2, \end{aligned} \tag{5}$$

  where $\mathcal{U}^{-}$ is an inverse of $\mathcal{U}$ since it is strictly increasing in temporal extent.

2.3 Practical issues

In the Step 1, the distances for all frame pairs between two compared video sequences F and F' (i.e., $\{(f(i), f'(j)) : i = 1, 2, \ldots, n; j = 1, 2, \ldots, n'\}$) have to be computed to synchronize these two sequences. Here, it is worth mentioning that for human actions studied in this paper, human actions are often or almost always represented by multiple features from different measurements, and furthermore each feature may provide different weights/cues for action discrimination. For this reason, the classical distance metric, typically the Euclidean distance, would be not suitable for coping with such multi-dimensional sequences. To address this problem, we employ the following procedures to compute the distance between each frame pair in implementation:

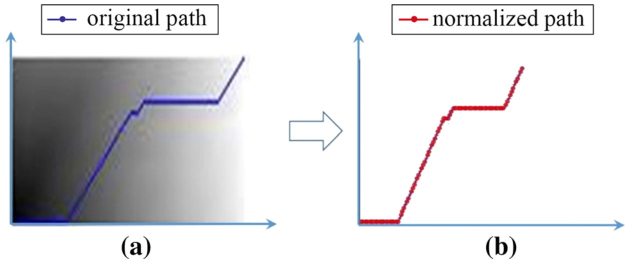


Fig. 2 Learning average sequences: **a** taking the optimal warping path from a two-dimensional square lattice resulted by computing the distances for all frame pairs between two compared video sequences; **b** normalizing the original warping path to be a common timing by the interpolation and averaging operations

- Normalize each dimension of F and F' separately to a zero mean and unit variance and smooth each dimension with a Gaussian filter;

- Compute the distance matrix $\mathcal{D}$ by:

$$\mathcal{D}(i, j) = \sum_{h=1}^{H} |f(i, h) - f'(j, h)| \tag{6}$$

  where $f(i, h)$, $f'(j, k)$ are the hth features, respectively, in $f(i)$ and $f'(j)$;

- Use $\mathcal{D}$ to find the optimal warping path with the Viterbi algorithm.

2.4 Action modeling and recognition

Assuming that we have learnt average sequences from every pair of action samples for each action class by the above-described procedures, we now have a set of average sequences $\{\mathcal{F}_i^c : i = 1, 2, \ldots, N^c\}$ for each class c, and apparently the number of this set is $N^c = C_N^2 = N(N-1)/2$. By collecting the two sets of average sequences and the original action samples together (as shown in Fig. 3), we can obtain a new set, i.e., $\mathcal{S}^c = \{F_i^c\} \cup \{\mathcal{F}_i^c\}$ for performing action modeling. We then use the bag-of-words (BoW) model to represent each sample in the new set $\mathcal{S}^c$ for modeling the action of class c as follows:

- The codebook (i.e., vocabulary of words) is first constructed by clustering $\{\mathcal{S}^c : c = 1, 2, \ldots, C\}$ (C is the

[1] Here, one notes that the normalized warping path do not need to be equally spaced.

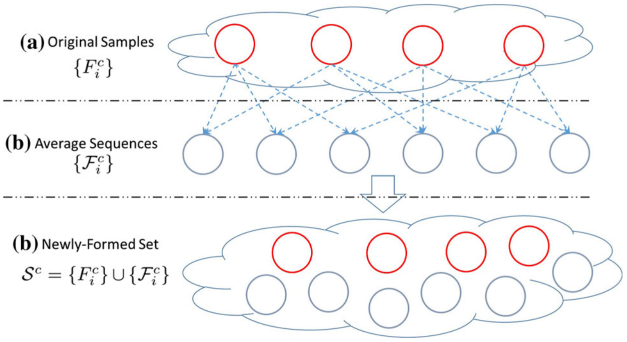


Fig. 3 Given an original set of action samples $\{F^c\}$ for class c, we learn average sequences $\{\mathcal{F}^c\}$ from every sample pair in the set. Then, action modeling is then performed with a newly formed set of $\mathcal{S}^c$ by taking together $\{F^c\}$ and $\{\mathcal{F}^c\}$

number of total classes) using k-means algorithm where codewords are defined by the centers of resulted clusters;

- Each frame in the video sample is assigned as one codeword by minimizing the Euclidean distance over all codewords in the codebook;
- Last, each video sample is described as a histogram of assigned codewords. The effect of codebook size K on action recognition was investigated in experiments (see Fig. 4).

Let us assume that we have a set of histograms of codewords with action labels $c \in \{1, 2, \ldots C\}$. For a newly arrived query action video F^* also represented by a histogram of codewords learnt already, we can classify it for example using k-nearest neighbors classifier (k-NNC) or a support vector machine (SVM).

3 Experimental validation

3.1 Dataset

Since our focus is on enhancing action recognition with less amount of action samples, we chose four small-scaled benchmarking datasets for our evaluation as follows:

The *Weizmann Dataset* consists of 90 video sequences including 10 categories of human action: bend, jack, jump, pjump, run, side, skip, walk, wave1 and wave2, performed by each of nine subjects.

The *UT-Tower Dataset*[2] consists of 108 video sequences from 9 types of actions: pointing, standing, digging, walking, carrying, running, wave1, wave2 and jumping. Each action is performed 12 times by 6 individuals.

The *UC-3D Motion Database*[3] consists of 11 different activities including 6 interactive actions and 5 single actions. In this paper, we mainly focus on individual actions, so we chose the 5 single actions in our investigation: bend, jumping, running, walking and sitting/standing cycle, performed 15 times by 5 individuals.

The *UTD Multimodal Human Action Dataset* (UTD MHAD) was released very recently [25]. In this dataset, each action is performed by 8 subjects. We tested 15 actions, i.e., swipe left, swipe right, wave, clap, throw, arm cross, basketball shoot, draw X, draw circle (clockwise), draw circle (counter clockwise), draw triangle, bowling, boxing, baseball swing, and tennis swing, in our investigation.

3.2 Experimental implementation

As stated previously, frame feature extraction is the first step for video analysis. In the experiments, we used local temporal self-similarities (LTSS) extracted from difference images for frame representation [26] due to its relative simpleness in implementation and its no-requirement of bounding-box annotation and subjection detection. We used the same parameter setting as described in [26] which brings the total number of features up to 240 in each frame. Here, one notes that, in the Weizmann dataset, the two actions of wave1 and wave2 have very similar flow and they are easily confused to each other by the flowed-based approaches, we thus in experiments only tested the action of wave1 as made in [26].

For all datasets, in the following experiments, we tested the codebook size K from 50 to 150 with a step of 5. We tested two widely used classification methods of k-NNC and SVM, for performing action recognition. They were operated, respectively, as follows:

k-NNC: we compared F^* with k nearest action samples in $\mathcal{S}^c$ for each action class c, i.e., $\{F_1^c, F_2^c, \ldots, F_k^c\} \subset \mathcal{S}^c$, by a distance metric *dist*, typically the Euclidean distance. Then the most similar class was chosen as

$$F^* \rightarrow \arg\min_c \sum_{i=1}^{k} dist(F^*, F_i^c). \tag{7}$$

SVM: we trained SVM with RBF kernel in a one-against-all framework to handle multi-class classification. LIBSVM library was used in MATLAB for implementing the SVM-based action classification.

We also compared these classification methods with the recently proposed deep learning (DL)-based method. We implemented the DL method based on convolutional neural network (CNN). More specifically, the Deep Learn Toolbox[4]

[2] http://cvrc.ece.utexas.edu/SDHA2010/Aerial_View_Activity.html.

[3] http://mrl.isr.uc.pt/experimentaldata/public/uc-3d/.

[4] https://github.com/rasmusbergpalm/DeepLearnToolbox.

was used in MATLAB for accomplishing this task where we trained a *6c-2s-12c-2s* CNN to deal with multi-class classification. In this method, we feed the extracted LTSS features directly to the DL classification.

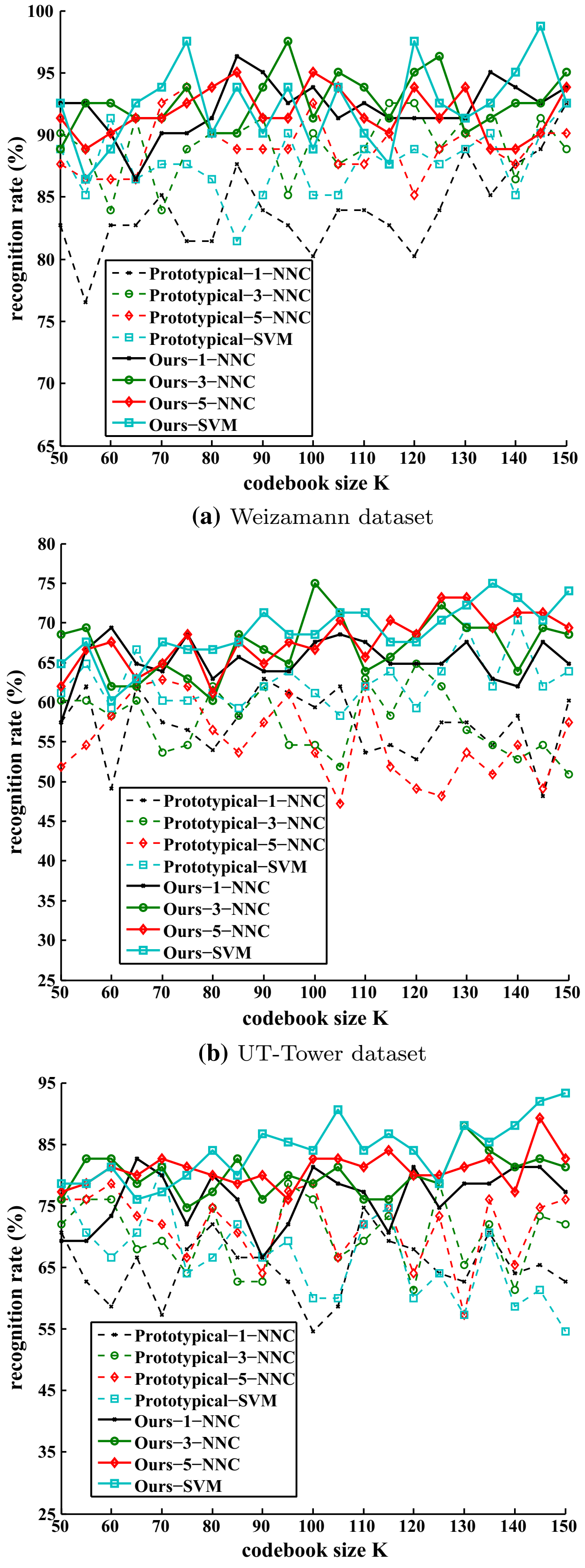
Additionally, for all the above classification methods, we compared the recognition performances of using original training samples with those obtained by the proposed scheme to investigate the effectiveness and priority. The leave-one-person-out cross-validation was used for classification evaluation.

3.3 Results and analysis

Figure 4 shows the recognition rates for tested values of codebook size K by using k-NNC or SVM classification. It can be seen that, for all datasets, the recognition performance has been improved significantly for almost all tested values of K with our proposed method than those obtained by original training samples. More specifically, in Fig. 5, we summarized the average recognition rates by using k-NNC and SVM classification as well as the recognition rate by DL. It can be seen that, for all datasets, the recognition rates by each classification method with using the extended samples are higher than those using the original prototypical samples.

More details are provided in Table 1 where we can find that, in Weizmann dataset, our method achieved a recognition rate of 98.77% (SVM, $K = 145$), while 93.83% was obtained by using the original samples (5-NNC, $K = 75$). UT-Tower dataset was 75% by our method (3-NNC, $K = 100$ and SVM, $K = 135$) and 70.37% (SVM, $K = 140$) with the original samples. In the UC-3D Motion dataset, it was 93.33% (SVM, $K = 150$) by our method, while it was 81.33% (SVM, $K = 70$) with the compared method. Last, in UTD MHAD dataset, our method achieved 91.67% (SVM, $K = 120$), while it was 84.17% (1-NNC, $K = 115$) with using the original samples. Here, one interesting observation is, for each testing dataset, the recognition rate of DL is lower than those by using k-NNC and SVM classification. It is not surprising because the performance of DL classification relies heavily on the number of training samples, while, the extended number of samples by our method is still somewhat limited on the testing datasets. In addition, there are some parameters that can significantly affect the performance of DL, for example, as reported in [27], the recognition rate on the Weizamann dataset can achieve 96.67% by using 3D CNN, that is higher than 88.89% reported in our experiment. In this regard, it is believed that the performance of DL method by integrating our proposed scheme would be further improved through optimizing appropriate settings. Further discussion is, however, beyond the scope of this paper as our focus in this paper is on the extension of training samples.

In the method, we propose to use the extended training set derived from SACA, instead of original training set, for

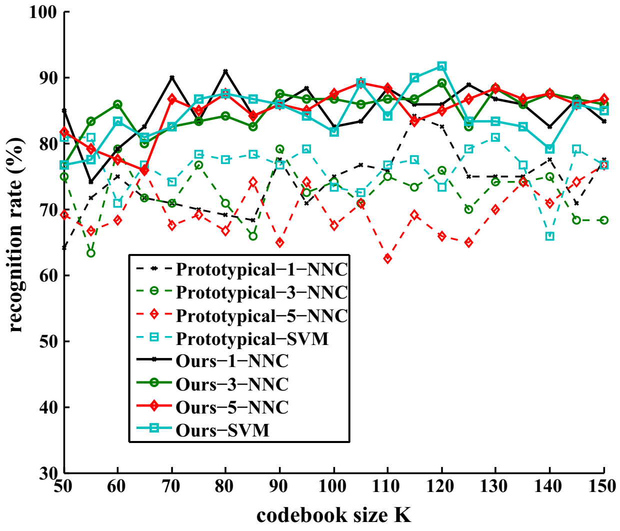


(a) Weizamann dataset

(b) UT-Tower dataset

(c) UC-3D Motion dataset

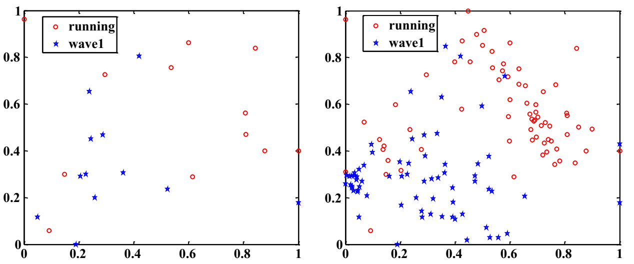
Fig. 4 Recognition rates in four datasets. **a** Weizamann dataset. **b** UT-Tower dataset. **c** UC-3D Motion dataset. **d** UTD MHAD dataset



(d) UTD MHAD dataset

Fig. 4 continued

action modeling and recognition. An example of two randomly selected actions in UT-Tower dataset is shown in Fig. 6. Intuitively, we can see that the samples are more *dense* within each action class and meanwhile these two actions have a more distinguishable classification boundary in the extended training set, compared with those in the original training set. These would be the main reasons why we can achieve better recognition performances with the proposed method.



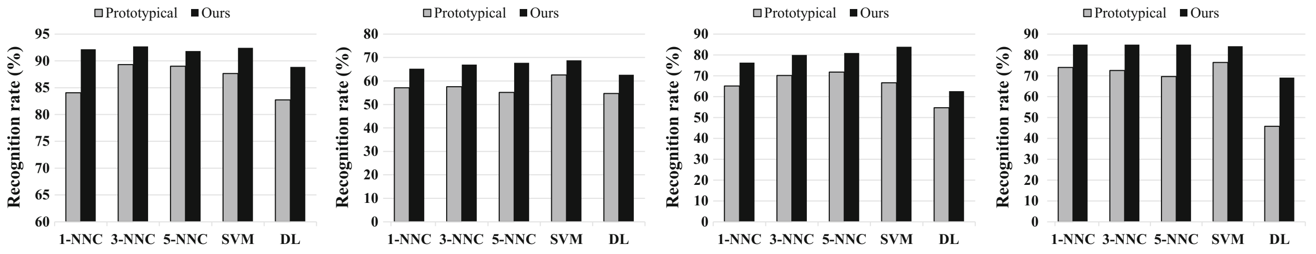
(a) Original samples **(b)** Extended training samples

Fig. 6 PCA-2D of two randomly selected actions in UT-Tower dataset

In addition, we derived structural average sequences by learning from each pair of video samples in every action class, which is conducted on the basis of features obtained previously. And we only chose two conventional classification methods and one deep learning method for comparison. In fact, since our proposed method is to extend the training samples prior to action modeling and recognition, other feature extraction methods and classification methods can also be integrated with our method to further improve their recognition performances, especially in the cases where there are a limited number of samples.

4 Conclusion

In this paper, we have proposed a new scheme for modeling human actions by virtue of SACA when only a limited number of training samples are available. Rather than directly



(a) Weizmann dataset **(b)** UT-Tower dataset **(c)** UC-3D Motion dataset **(d)** UTD MHAD dataset

Fig. 5 Average recognition rates by using *k*-NNC and SVM classification and recognition rates by DL method

Table 1 Summary of experimental results

Dataset	Original samples					Our method				
	1-NNC	3-NNC	5-NNC	SVM	DL	1-NNC	3-NNC	5-NNC	SVM	DL
Weizmann	92.59	92.59	**93.83**	92.59	82.72	96.3	97.53	95.06	**98.77**	88.89
UT-Tower	62.96	64.81	62.89	**70.37**	64.81	69.44	**75.00**	73.15	**75.00**	72.22
UC-3D Motion	74.67	78.67	78.67	**81.33**	54.67	82.67	88.00	89.33	**93.33**	62.67
UTD Multimodal	**84.17**	79.17	76.67	80.83	45.83	90.00	89.17	89.17	**91.67**	69.17

Bold values are for emerging the best recognition rate either using original samples or using our method using the original training set for action modeling, we derived structural average sequences by learning from each pair of video samples in every action class and then combined them with original video samples to generate a new training set. Extensive experiments and methodological analysis on the new training set were provided to demonstrate the advantages of the proposed method. In addition, the proposed method can potentially be integrated with other approaches to further improve their recognition performances.

Acknowledgements The work is supported by National Natural Science Foundation of China (61403232, 61327003), Natural Science Foundation of Shandong Province, China (ZR2014FQ025), and Young Scholars Program of Shandong University (YSPSDU, 2015WLJH30).

References

1. Mahbub, U., Imtiaz, H., Ahad, M.A.R.: Action recognition based on statistical analysis from clustered flow vectors. Signal Image Video Process. **8**(2), 243–253 (2014)
2. Shao, L., Zhen, X., Tao, D., Li, X.: Spatio-temporal laplacian pyramid coding for action recognition. IEEE Trans. Cybern. **44**(6), 817–827 (2014)
3. Pei, L., Ye, M., Zhao, X., Xiang, T., Li, T.: Learning spatio-temporal features for action recognition from the side of the video. Signal Image Video Process. **10**, 199–206 (2016)
4. Keçeli, A.S., Kaya, A., Can, A.B.: Combining 2d and 3d deep models for action recognition with depth information. Signal Image Video Process. (2018). https://doi.org/10.1007/s11760-018-1271-3
5. Wang, L., Qiao, Y., Tang, X.: Action recognition with trajectory-pooled deep-convolutional descriptors. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4305–4314. IEEE (2015)
6. Chaaraoui, A.A., Padilla-Lpez, J.R., Climent-Prez, P., Flrez-Revuelta, F.: Evolutionary joint selection to improve human action recognition with rgb-d devices. Expert Syst. Appl. **41**(3), 786–794 (2014)
7. Duan, L., Xu, D., Tsang, I.H., Luo, J.: Visual event recognition in videos by learning from web data. IEEE Trans. Pattern Anal. Mach. Intell. **34**(9), 1667–1680 (2012)
8. Zhou, F., De la Torre, F.: Generalized time warping for multi-modal alignment of human motion. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1282–1289. IEEE (2012)
9. Guha, T., Ward, R.K.: Learning sparse representations for human action recognition. IEEE Trans. Pattern Anal. Mach. Intell. **34**(8), 1576–1588 (2012)
10. Theodorakopoulos, I., Kastaniotis, D., Economou, G., Fotopoulos, S.: Pose-based human action recognition via sparse representation in dissimilarity space. J. Vis. Commun. Image Represent. **25**(1), 12–23 (2014)
11. Zhang, D., Gatica-Perez, D., Bengio, S., McCowan, I.: Semi-supervised adapted hmms for unusual event detection. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), pp. 611–618. IEEE (2005)
12. Laptev, I., Marszalek, M., Schmid, C., Rozenfeld, B.: Learning realistic human actions from movies. In: 2008 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1–8. IEEE (2008)
13. Seo, H.J., Milanfar, P.: Action recognition from one example. IEEE Trans. Pattern Anal. Mach. Intell. **33**(5), 867–882 (2011)
14. Rodriguez, M.D., Ahmed, J., Shah, M.: Action mach a spatio-temporal maximum average correlation height filter for action recognition. In: 2008 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1–8. IEEE (2008)
15. Neverova, N., Wolf, C., Taylor, G.W., Nebout, F.: Multi-scale deep learning for gesture detection and localization. In: Computer Vision-ECCV 2014 Workshops, pp. 474–490. Springer International Publishing (2014)
16. Schmid, M.F., Booth, C.R.: Methods for aligning and for averaging 3d volumes with missing data. J. Struct. Biol. **161**(3), 243–248 (2008)
17. Demsar, J.: Statistical comparisons of classifiers over multiple data sets. J. Mach. Learn. Res. **7**, 1–30 (2006)
18. Amit, K., Kaustubh, K., Srikanth, C.V.: Ramasubramanian: Towards fast, view-invariant human action recognition. In: 2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 1–8. IEEE (2008)
19. Ahmadi, S.A., Sielhorst, T., Stauder, R., Horn, M., Feussner, H., Navab, N.: Recovery of surgical workflow without explicit models. In: International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI), pp. 420–428. Springer, Berlin (2006)
20. Boudaoud, S., Rix, H., Meste, O.: Core shape modelling of a set of curves. Comput. Stat. Data Anal. **54**(2), 308–325 (2010)
21. Morlini, I., Zani, S.: Estimation of the structural mean of a sample of curves by dynamic time warping. Data Analysis, Classification and the Forward Search, pp. 39–48. Springer, Berlin (2006)
22. Xie, X., De Vylder, J., Van Cauwelaert, D., Veelaert, P., Philips, W., Aghajan, H.: Average track estimation of moving objects using ransac and dtw. In: Proceedings of the International Conference on Distributed Smart Cameras, Article No. 28. ACM (2014)
23. Seltzer, M.L., Yu, D., Wang, Y.: An investigation of deep neural networks for noise robust speech recognition. In: 2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 7398–7402. IEEE (2013)
24. Liu, Y., Shriberg, E., Stolcke, A., Hillard, D., Ostendorf, M., Harper, M.: Enriching speech recognition with automatic detection of sentence boundaries and disfluencies. IEEE Trans. Audio Speech Lang. Process. **14**(5), 1526–1540 (2006)
25. Chen, C., Jafari, R., Kehtarnavaz, N.: UTD-MHAD: a multimodal dataset for human action recognition utilizing a depth camera and a wearable inertial sensor. In: IEEE International Conference on Image Processing, pp. 168–172. IEEE (2015)
26. Lu, G., Kudo, M.: Learning action patterns in difference images for efficient action recognition. Neurocomputing **123**, 328–336 (2014)
27. Jung, M., Hwang, J., Tani, J.: Multiple spatio-temporal scales neural network for contextual visual recognition of human actions. In: Joint IEEE International Conferences on Development and Learning and Epigenetic Robotics, pp. 235–241. IEEE (2014)