Journal of Applied Sciences

ISSN 1812-5654

science alert

ANSInet
an open access publisher
http://ansinet.com

Journal of Applied Sciences 14 (20): 2578-2585, 2014
ISSN 1812-5654 / DOI: 10.3923/jas.2014.2578.2585

Action Recognition Based on Hierarchical Spatio-Temporal Motion Features

[1]Jiangfeng Yang, [1]Zheng Ma and [2]Guojun Lin
[1]School of Communication and Information Engineering,
[2]School of Electronic Engineering, University of Electronic Science and Technology of China,
Chengdu, 611731, China

Abstract: In this study, human action representation based on hierarchical optical flow motion feature was proposed, which can extract more informative motion information than recent proposed approaches. Dynamic Time Warping (DTW) algorithm is widely used in action sequence recognition for eliminating the negative effect caused by performing an identical action in different styles and/or different rates. To improve the temporal consistence on action sequence, automatic spatial and temporal alignment approach based on an improved DTW was proposed. Moreover, to reduce time-consumption and increase computational efficiency during action sequence matching, a novel matching algorithm based on multi-resolution DTW constraint was developed. Experimental results on the Weizmann and KTH databases show that the proposed system can achieve higher recognition performance.

Key words: Hierarchical spatio-temporal motion feature, action recognition, action representation, dynamic time wraping

INTRODUCTION

Algorithms for recognizing human actions in a video sequence are needed in applications such as video surveillance, where the goal is to look for typical and anomalous patterns of behavior and video search and retrieval in large, potentially distributed, video databases such as YouTube. Developing algorithms for action recognition in video that are not only accurate but also efficient in terms of computation and memory-utilization is challenging due to the complexity of the task and the sheer size of video.

In previous studies, two types of approaches have been proposed. One is to extract features from video sequence and compare with pre-classified features (Gorelick *et al*., 2007; Yilmaz and Shah, 2005). These methods used some voting mechanism to obtain better recognition performance and can adapt more variance by using large amount of training data. Another approach builds up a class of models from training set and computes, the recognition rate on test data related to these models (Ali *et al*., 2007; Oikonomopoulos *et al*., 2011). Because these models have unique features it may lose some characteristics of the feature. This approach is computationally efficient, but its accuracy is not as good as the first approach. In practical application, a large set of good training data is needed to obtain high recognition accuracy. As the result, we should balance the trade between accuracy and computational cost.

In this study, we focused on achieving a higher computational efficiency without sacrificing accuracy rate significantly and recognizing action in a real environment. Inspired by the observation of optical flow of human action proposed by Efros *et al.* (2003), who used optical flow as motion feature of actions (Poppe, 2010), we extracted shape information from training data and developed an improved Dynamic Time Warping (DTW) algorithm for measuring the similarity of two actions. Next, k Nearest Neighbor (k-NN) voting mechanism is combined with motion sequence pyramid to achieve a better recognition performance. Finally, considering computational efficiency, spatial enhancement on k-NN pyramid and Multi-Resolution (MR) DTW constraint are utilized.

The main contributions of this study are: (1) A hierarchical spatial-temporal optical flow capturing method which can extract more motion information, (2) A automatic spatial and temporal alignment method based on an improved DTW algorithm to increase temporal consistence of motion sequence and (3) MR-DTW constraint on motion features pyramids to speed up recognition process.

MATERIALS AND METHODS

System framework: The proposed action recognition is showed in Fig. 1. At stage 1, an input video is preprocessed and aligned with central Spatio-Temporal

Corresponding Author: Jiangfeng Yang, School of Communication and Information Engineering,
University of Electronic Science and Technology of China, Chengdu, 611731, China

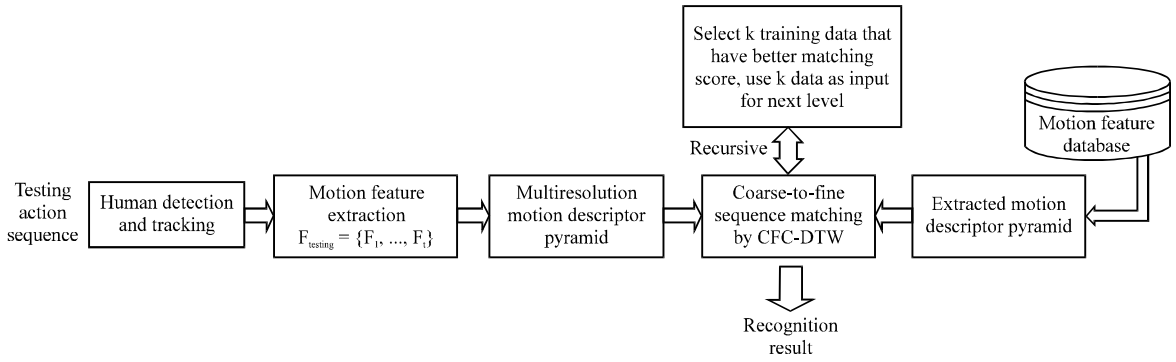

Fig. 1: Proposed system framework

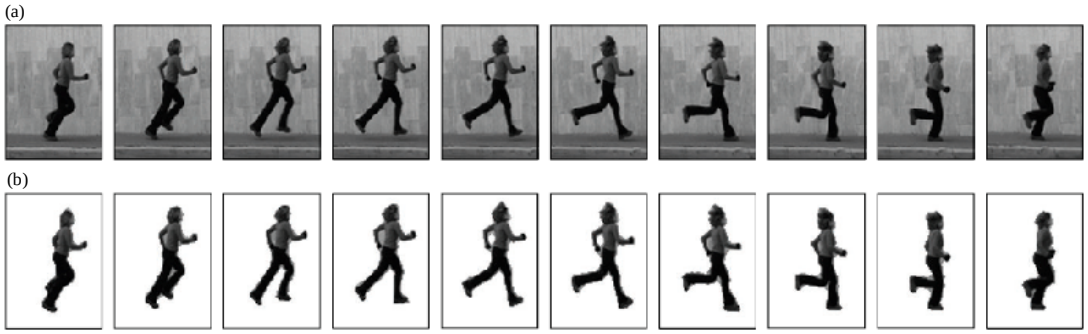

Fig. 2(a-b): (a) Original video sequence for testing data (Lena run1 from the Weizmann dataset) and (b) Background subtracted video sequence to calculate optical flow for training data

(ST) volume of each action. At stage 2, optical flow descriptors are calculated and formed into multi-resolution pyramid features. At stage 3, similarity matrix between actions is computed, an improved MR-DTW constraint algorithm is applied to calculate the similarity of the two actions for saving computation time. Finally, the test video is recognized.

Above all, a figure-centric ST volume is extracted from an input image sequence. This figure-centric volume can be obtained by tracking the human figure and then constructing a window in each frame centered at the figure. Any of a number of trackers are appropriate. The main requirement is that the tracking be consistent a person in a particular body configuration should always map to approximately the same stabilized image. The input to our recognition system should be stabilized to ensure the center of figures is aligned in space. In our system, background subtraction in the Weizmann action dataset and object tracking in KTH dataset are used as preprocessing. As shown in Fig. 2, in order to reduce the influence of noise and separate the moving human from background, background is subtracted from the original video sequence. Next, the resulting frames are sent to optical flow calculation.

Given a stabilized figure-centric sequence, we first computed optical flow at each frame using (Lucas and Kanade, 1981) algorithm. The optical flow vector field F is first split into two scalar fields corresponding to the horizontal and vertical components of the flow, F_x and F_y, each of which is then half-wave rectified into four non-negative channels $F_x^+, F_x^-, F_y^+, F_y^-$, so that $F_x = F_x^+ - F_x^-$ and $F_y = F_y^+ - F_y^-$. These are each blurred with a Gaussian and normalized to obtain the final four channels $\hat{F}_x^+, \hat{F}_x^-, \hat{F}_y^+, \hat{F}_y^-$, of the motion descriptor for each frame. Alternative implementation of the basic idea could use more than 4 motion channels the key aspect is that each channel be sparse and nonnegative. Results are shown in Fig. 3.

If the four motion channels for frame i of sequence A are a_1^i, a_2^i, a_3^i and a_4^i and the four motion channels for frame j of sequence B are b_1^i, b_2^i, b_3^i and b_4^i, then the similarity between frames i, j is defined as:

$$S(i,j) = \sum_{c=1}^{4} \sum_{x,y \in I} a_c^i(x,y) b_c^j(x,y) \quad (1)$$

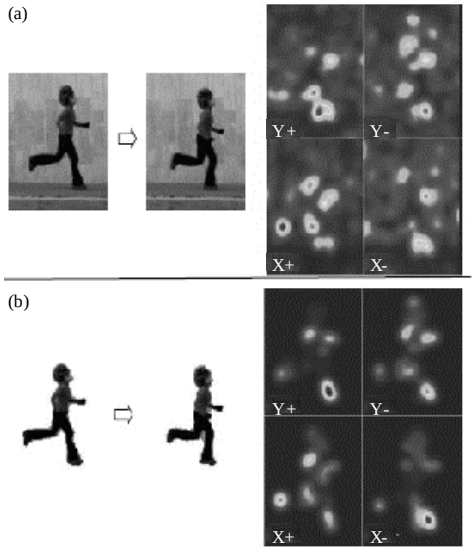

Fig. 3(a-b): Optical flow descriptors of "Lena run1" (a) Optical flow from original images and (b) Optical flow after background subtraction

To compare two sequences A and B, the similarity computation will need to be done for every frame of A and B.

In order to get smoother results of similarity matrix calculated by: (1) Convolution is performed as: (2) with identity matrix $I_T = diag(1, \ldots, 1) \in R^{T \times T}$, diag() denotes a diagonal matrix; T denotes how many frames to be smoothed which could improve the accuracy in dynamic time warping:

$$S^T(i,j) = S(i,j) \times I_T \quad (2)$$

When classifying a query action sequence it is compared to the training sequences in lower resolution level of the feature pyramid. The best matching is chosen by action width k-NN. And then this work is refined in a higher resolution level of the pyramid for selecting the best matching by k-NN.

Due to the complexity of action recognition problem, some actions are periodic while others are not. A novel framework is developed to handle the problems. Both a similarity based on DTW (Sakoe and Chiba, 1978) and an improved DTW algorithm were developed. Finally, similarity between query sequence and training sequence is calculated and the action with the best similarity score labeled the query action.

Improved DTW algorithm: The similarity between two sequences can be measured by their similarity matrix and

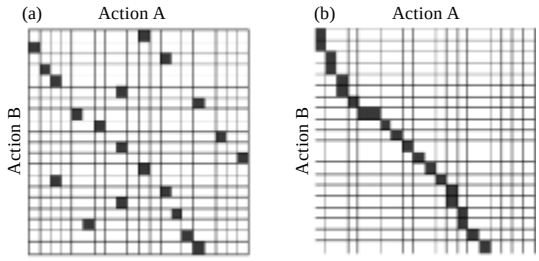

Fig. 4(a-b): Compare k-NN with DTW, (a) Frame-to-frame nearest neighbor and (b) Action-to-action DTW path

the matrix can represent how much these two input actions are like each other. Frame-to-frame voting mechanism was used to calculate the similarity between two sequences (Efros *et al*., 2003; Schindler and Van Gool, 2008). For each frame in test data, k frames with the best matching score in training data were selected by voting. Although this simple selection of the best matching score in all frames should result in a better recognition rate, noise in some frames will cause a negative matching. Moreover, due to the poor space alignment of action frames, actions in same class obtain a low similarity but actions with different class show higher similarity. This k-NN algorithm is lack of a self-corrective mechanism that can keep the frame match continuity in time domain.

Different from frame-to-frame k-NN algorithm (Efros *et al*., 2003), action-to-action similarity was adopted in our approach. This measurement calculates from frame-to-frame similarity matrix by summing up similarity values on the DTW path. This similarity measurement can be adaptive to speed variation of actions. Furthermore, it keeps the continuity of frames in time domain. One frame can be correctly matched to another even if it does not have the highest matching score and it just lays on a DTW path which will enhance the accuracy in action recognition. The demonstration of frame-to-frame similarity and action-to-action similarity is shown in Fig. 4 and the similarity is defined as follows:

$$M_{DTW} = \sum_{[i,j] \in path} \frac{S^T(i,j)}{Length} \quad (3)$$

where, Length denotes the length of DTW path and M_{DTW} denotes matching score.

When DTW used in speech recognition area, some constraints are added to the original algorithm for better

recognition accuracy. Sakoe and Chiba (1978) gave their Sakoe-Chiba Band and Itakura (1975) showed the Itakura Parallelogram and the latter was widely used in the speech recognition community. Since a sentence always has a start position and an end position, applying these two constraints will obtain better alignment and recognition results. Other approaches talk about processing of cyclic patterns on text matching or sequence matching (Wang *et al*., 2007). The DTW algorithm is also widely used in signal processing area, such as finding waveform pattern of ECG (Wang *et al*., 2007; Laguna *et al*., 1994). Recently, some studies in data mining used DTW as a sequence matching method and got inspiring achievement; they showed their new boundary constraints and got good experiment result on video retrieval, image retrieval, handwriting retrieval and text mining.

Previous study on DTW shows that proper constraints on DTW can achieve better recognition performance. Unlike general speech recognition, in action recognition, there are lots of periodical actions. Therefore it is desired to find a new constraint on the original DTW algorithm, so that adapting periodical actions and automatic aligning the start and end positions of actions. While matching two actions, traditional DTW leads to a matching path on similarity matrix as shown in Fig. 5a. It looks like an actual path segment and two straight lines from the start point and to the end point. Because the two straight lines are not required when calculating similarity value, a new method in Fig. 6 was developed in the present study to get an accurate matching path as shown in Fig. 5b.

In our enhanced DTW algorithm, a constraint called Multi-Resolution (MR) constraint is developed. It improves recognition speed by time complexity $O(n^2)$.

Multi-resolution matching: Similarity matrix calculation is a time-consuming problem, since for obtaining only one element in the action-to-action similarity matrix, multiplication frame by frame should be implemented and its time complexity is $O(n^2)$. Therefore, similarity matrix with n×n elements requires a total computation complex $O(n^4)$. When the training/test data becomes bigger, more similarity calculations are needed. For example, processing all 93 videos in the Weizmann dataset, the calculation of similarity matrix takes about 30 min in a 2.5 GHz Pentium E5200 Dual-Core computer. It takes about 20 sec for per recognition which is an unacceptable performance.

As presented in Fig. 1, the main idea of this study is comparing the similarity of two actions using

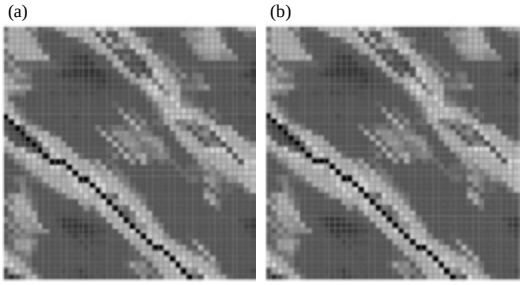

Fig. 5(a-b): DTW matching path of two actions, (a) Original DTW algorithm result and (b) Auto aligned DTW path shows matching start point and matching end point

multi-resolution motion feature pyramid. Firstly, similarities were measured in low resolution level, then in a higher resolution level, till the highest level. In each of multi-resolution comparison steps except for the highest level, actions that are selected by comparison only in lower resolution level are used as input of higher resolution level. That is, when comparing actions in low resolution level, actions that have the highest matching score in comparison results are selected by k-NN. These selected actions are used as the input for the higher resolution level in this multi-resolution motion features pyramid.

At lowest resolution level, all pre-classified action sequences are compared to the test action, but the scales of these actions are very small. Therefore, the required computational burden is less than action comparison in original scale. On the other hand, higher computation cost lies in higher resolution level in the pyramid but only a few pre-classified actions should be compared. The overall computational cost is decreased. This method is more than 10 times faster than calculating the similarities in original resolution.

For the purpose of performance, a new DTW constraint is applied to the multi-resolution motion feature pyramid. Each DTW matching path of similarity matrix is saved as a constraint for higher level .When calculating similarity matrix in a higher resolution level, the saved path is convoluted with a kernel of 5×5 defined as:

$$K = (k_{ij})_{5\times5}, \quad k_{ij} = 1 \qquad i, j = -2, -1, \cdots, 2 \tag{4}$$

The convoluted kernel will be used as a constraint in DTW algorithm. We name this constrained DTW as MR-DTW.

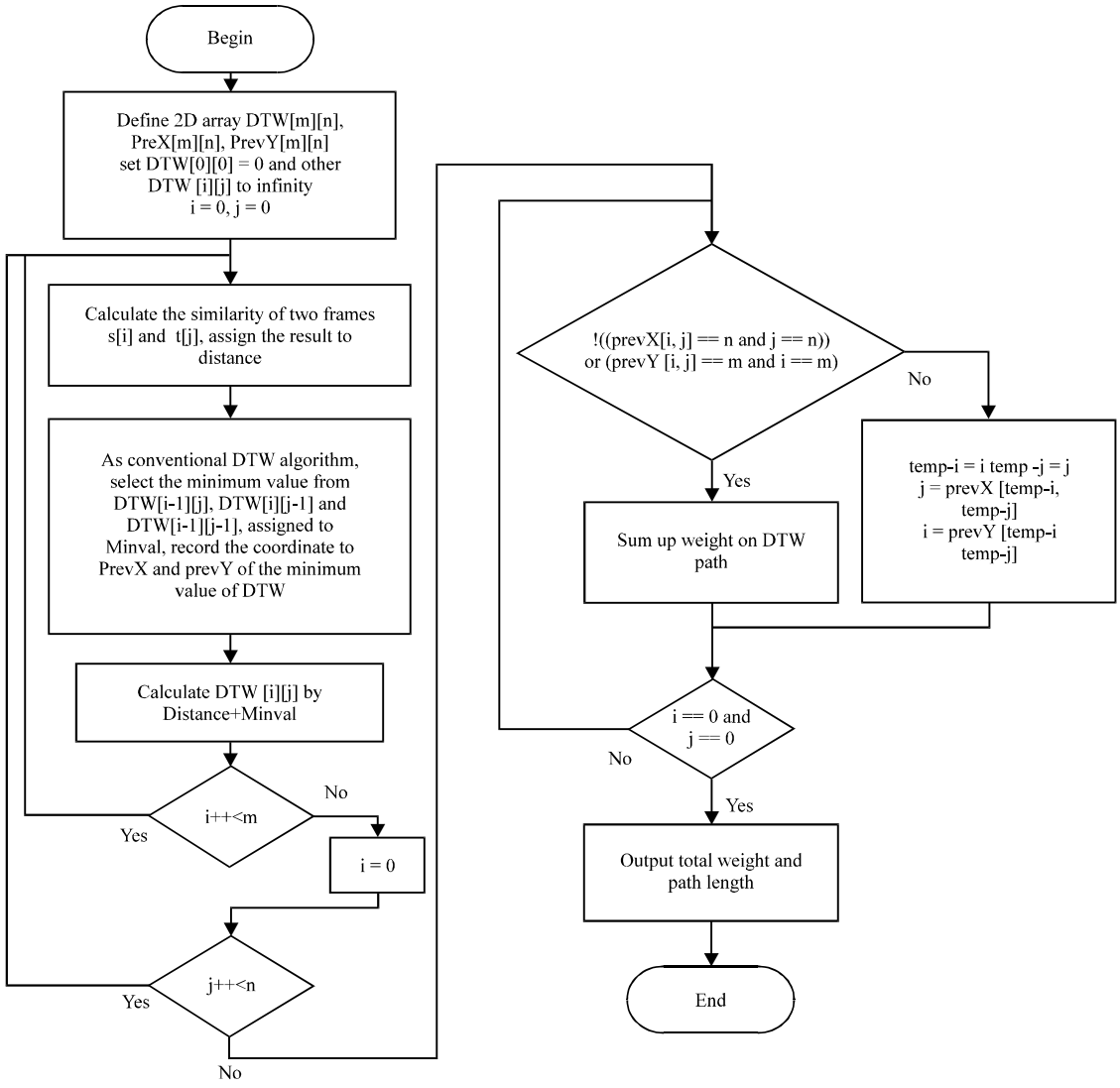

Fig. 6: Flowchart of auto-aligned DTW

Gaussian pyramid in multi-scale image: In image processing, Gaussian pyramid is widely used in image matching, image fusion, segmentation and texture synthesis. Using low-pass filtering on the images and followed by sub-sampling can generate Gaussian Pyramid shown in Fig. 7. Each pixel value at level L is computed as a weighted average of pixels in a 5×5 neighborhood at level L-1. Given an initial image $F_0(i, j)$ with size M×N, the level-to-level weighted average is calculated by Burt and Adelson (1983):

$$F_L(i,j) = \sum_{m=-2}^{2}\sum_{n=-2}^{2} r(m,n)F_{L-1}(2i+m,2j+n) \quad (5)$$

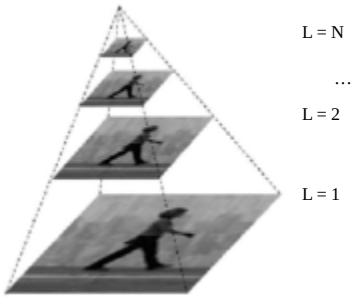

Fig. 7: Gaussian pyramid

where, r(m, n) is a separable 5×5 Gaussian low pass filter given by Sakoe and Chiba (1978):

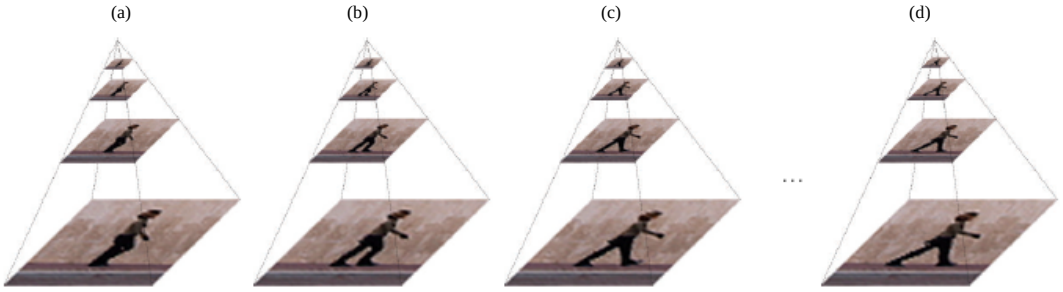

Fig. 8(a-d): Motion sequence pyramid F(i, j, t) at (a) $t = t_0$, (b) $t = t_1$, (c) $t = t_2$ and (d) $t = t_n$

$$\begin{aligned} &r(m,n) = r(m)r(n) \\ &r(0) = a \\ &r(1) = r(-1) = 1/4 \\ &r(2) = r(-2) = 1/4 - 2/a \end{aligned} \tag{6}$$

Parameter a is set from 0.3-0.6 based on experiment results. The separation of r(m, n) will reduce the computational complexity in generating multi-scale images.

Motion sequence pyramid: By extending Gaussian Pyramid, multi-resolution similarity algorithm was introduced for reducing computation complexity. Each pyramid has L levels and every level is constructed by scaling original frame. The lowest level in the motion sequence pyramid has motion feature images with original size, while the higher level image has smaller size.

In training, all L level pyramids of motion sequence descriptor in training database are stored as pre-classified actions database for similarity calculation. Consider:

$$F_L(i,j,t) = \sum_{m=-2}^{2} \sum_{n=-2}^{2} r(m,n) F_{L-1}(2i+m, 2j+n, t) \tag{7}$$

where, $F_L(i, j, t)$ denotes the image F(i, j) on level L at frame t, as Fig. 8.

It is obvious that computing similarity between two motion sequence pyramids at every level is not needed because L_0 has the most feature information. Performing Eq. 3 on level L_0 can obtain good recognition rate. But big frame size causes high computational burden.

Considering time consumption, a new strategy of multi-layer classification is proposed, whose basic idea is that sequence matching starts from the lowest resolution level L_n rather than level L_0. This strategy can obtain an acceptable classification result, in which actions with significant difference are separated apart like walking and waving hand, running and bending, jogging and boxing. On the meantime, time consumption is greatly reduced. This result can be used as the input of a higher resolution level. After obtaining the classification result at level L_s, selecting k actions according to M_{DTW} at a descent order and using them as the input of a higher resolution level L_{s-1} classification. This refinement is repeated until level L_0 is reached. Parameter s can be set as:

$$s = 1/(1+L)^2$$

or decided in cross-validation strategy.

Our results show that for the balance of time consumption and recognition accuracy, two levels of pyramid can achieve satisfactory results.

Multi-Resolution (MR) DTW: When searching for the best action matching in coarse-to-fine resolution motion sequence pyramid, MR-DTW is performed to accelerate calculation performance. First, in MR-DTW, similarity matrix $S^T_{L_n}$ of two sequences is calculated using Eq. 2. Performing algorithm on $S^T_{L_n}$, 2D matrix T_{L_n} denoting a DTW path is given as:

$$T_{L_n}(i,j) = \begin{cases} 1 & S^T_{L_n}(i,j) \in \text{path} \\ 0 & S^T_{L_n}(i,j) \notin \text{path} \end{cases} \tag{8}$$

Secondly, as shown in Fig. 9, convoluting $T_{L_n}(i, j)$ with kernel K leads to a MR constraint on the higher resolution level L_{n-1} and a 5×5 rectangle convolution kernel has been used in our study. Consider:

$$ctr_{L_{n-1}}(i,j) = T_{L_n}(i,j) \times K \tag{9}$$

Due to $ctr_{L_{n-1}}(i, j)$, the computation complexity of $S^T_{L_{n-1}}$ has been reduced. This MR constraint saves computation time from frames by frames multiplication.

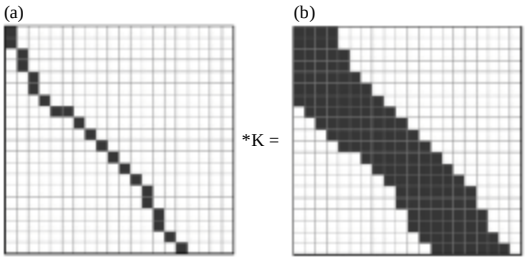

Fig. 9(a-b): Multi-Resolution Constraint DTW (MRC-DTW), (a) DTW path of lower level and (b) DTW constraint on higher level

RESULTS AND DISCUSSION

Experimental setting: We evaluated our approach on public benchmark datasets: The Weizmann human action dataset (Gorelick *et al*., 2007) and the KTH action dataset (Schuldt *et al*., 2004). The Weizmann dataset contains 93 low-resolution (188×144 pixels, 25 fps) video sequences showing 9 persons performing a set of 10 different actions: Bending down, jumping jack, waving one hand, waving two hands, in place jumping, jumping, siding, running and walking. Background subtraction is used to get shape information and optical flow features of actions.

The KTH dataset contains 25 persons acting 6 different actions: Boxing, hand-clapping, jogging, running, walking and hand-waving. These actions are recorded under 4 different scenarios: Outdoors (s1), outdoors with scale-variations (s2), outdoors with different appearance (s3) and indoors (s4). Each video sequence can be divided into 3 or 4 sub-sequences for different direction of jogging, running and walking. Human tracking method was used to get action centric volumes. Background subtraction was not applied in this case. Results were compared with previous studies.

Leave-one-out (LOO) mechanism was used in the experiments. Each testing action had been compared to other 92 actions in the dataset. A total recognition rate and an average recognition time of each algorithm were evaluated. All methods mentioned in this study were combined to construct the spatial-temporal motion feature.

The hardware environment is composed of a Windows 7 PC with 2.5 GHz Pentium E5200 Dual-Core CPU and 2G Bytes system memory.

Experimental results: Results of multi-resolution pyramid method on Weizmann dataset were shown in Table 1. For computation efficiency, a two-level pyramid was built and different resolution of each level were used. Recognition rate and average recognition time per action are shown in Table 1. The k denotes the number of input actions from lower resolution level to higher resolution level. The MR-DTW was used in this experiment at the same time.

Table 1: Experimental results on the Weizmann dataset

Methods	Recognition rate (%)	Recognitiontime(sec)
Improved DTW		
Original image size	100	5.33
Level 1:30% size		
Level 2:100% size	97.8	1.17
K = 3		
MRC-DTW		
Level 1:30% size		
Level 2:100% size	98.9	1.30
K = 5		
Level 1:20% size		
Level-250% size	98.9	0.55
K = 10		

Table 2: Comparison with other approaches on the Weizmann dataset

Method	Recognition rate (%)	Frame No.
Gorelick *et al.* (2007)	93.5	2
	96.1	3
	99.2	10
Schindler and Van Gool (2008)	93.8	2
	96.3	3
	99.6	10
Jhuang *et al.* (2007)	97.8	All
Fathi and Mori (2008)	100.0	All
Our approach	100.0	All

Table 3: Experimental result on the KTH dataset

	Schindler and Van Gool (2008)		
Scenarios	SNIPPET 1/SNIPPET 7(%)	Jhuang *et al.* (2007) (%)	Our approach (%)
S1	90.9/93.0	96.1	94.0
S2	78.1/81.1	86.7	88.3
S3	88.5/92.1	89.8	91.4
S4	92.4/96.4	94.8	93.8

Experimental result in Table 2 shows that our enhanced DTW algorithm achieves 100% recognition rate with all frames calculated. By MR-DTW acceleration, the recognition speed is 10 times faster than enhanced DTW and still gets acceptable recognition rate. In Table 1, 0.55 sec period means that the MR-DTW algorithm can be used in practical applications.

On KTH dataset, our approach obtained the best result in scenario s2 comparing to studies of Schindler and Van Gool (2008) and Jhuang *et al.* (2007) (Table 3). Videos in this scenario were captured from outdoors environments and with camera zoom-in and zoom-out, because the MR-DTW keeps the continuity of each frame in action sequence during matching. If one frame is not matched it had always been corrected by nearly frames. The average recognition time was near 3 sec in our test platform. This performance can be improved by multi-core technology and GPU computation for real-time purpose.

CONCLUSION

In the study, we presented a fast action recognition algorithm with enhanced MR-DTW. Although DTW is a time-consuming method, the proposed MR-DTW and motion pyramid significantly speed up the traditional DTW method. Therefore, real-time recognition becomes possible in practice. Because the DTW can align the continuity of action, even low resolution videos can achieve an acceptable recognition rate. Furthermore, the algorithm developed in this study can be applied to thin client communication environment, since the multi-resolution feature of MR-DTW can fit the requirement of action recognition in the environment (Wang *et al*., 2009, 2012a, b) and modal data can be transferred in different level based on requirements.

ACKNOWLEDGMENT

This study is supported by National Nature Science Foundation of China (Grant No. 61271288).

REFERENCES

Ali, S., A. Basharat and M. Shah, 2007. Chaotic invariants for human action recognition. Proceedings of the IEEE 11th International Conference on Computer Vision, October 14-21, 2007, Rio de Janeiro, Brazil, pp: 1-8.

Burt, P.J. and E.H. Adelson, 1983. The laplacian pyramid as a compact image code. IEEE Trans. Commun., 31: 532-540.

Efros, A.A., A.C. Berg, G. Mori and J. Malik, 2003. Recognizing action at a distance. Proceedings of the 9th International Conference on Computer Vision, Volume 2, October 13-16, 2003, Nice, France, pp: 726-733.

Fathi, A. and G. Mori, 2008. Action recognition by learning mid-level motion features. Proceedings of the 26th IEEE Conference on Computer Vision and Pattern Recognition, June 23-28, 2008, Anchorage, AK., pp: 1-8.

Gorelick, L., M. Blank, E. Shechtman, M. Irani and R. Basri, 2007. Actions as space-time shapes. IEEE Trans. Pattern Anal. Mach. Intell., 29: 2247-2253.

Itakura, F., 1975. Minimum prediction residual principle applied to speech recognition. IEEE Trans. Speech Signal Process., 23: 67-72.

Jhuang, H., T. Serre, L. Wolf and T. Poggio, 2007. A biologically inspired system for action recognition. Proceedings of the IEEE 11th International Conference on Computer Vision, October 14-21, 2007, Rio de Janeiro, pp: 1-8.

Laguna, P., R. Jane and P. Caminal, 1994. Automatic detectionofwave boundaries in multilead ECG signals: validation with the CSE database. Comput. Biomed. Res., 27: 45-60.

Lucas, B.D. and T. Kanade, 1981. An iterative image registration technique with an application to stereo vision. Proceedings of the 7th International Joint Conference on Artificial Intelligence, Volume 2, Vancouver, BC, Canada, August 24-28, 1981, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA., pp: 674-679.

Oikonomopoulos, A., I. Patras and M. Pantic, 2011. Spatiotemporal localization and categorization of human actions in unsegmented image sequences. IEEE Trans. Image Process., 20: 1126-1140.

Poppe, R., 2010. A survey on vision-based human action recognition. Image Vision Comput., 28: 976-990.

Sakoe, H. and S. Chiba, 1978. Dynamic programming algorithm optimization for spoken word recognition. IEEE Trans., Acoust. Speech Signal Process., 26: 43-49.

Schindler, K. and L. Van Gool, 2008. Action snippets: How many frames does human action recognition require? Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, June 23-28, 2008, Anchorage, AK., USA., pp: 1-8.

Schuldt, C., I. Laptev and B. Caputo, 2004. Recognizing human actions: A local SVM approach. Proceedings of the 17th International Conference on Pattern Recognition, Volume 3, August 23-26, 2004, Cambridge, UK., pp: 32-36.

Wang, F., T. Syeda-Mahmood and D. Beymer, 2007. Finding disease similarity by combining ECG with heart auscultation sound. Proceedings of the Computers in Cardiology, September 30-October 3, 2007, Durham, NC., pp: 261-264.

Wang, J.B., M. Chen, X. Wan and C. Wei, 2009. Ant-colony-optimization-based scheduling algorithm for uplink CDMA nonreal-time data. IEEE Trans. Veh. Technol., 58: 231-241.

Wang, J.B., Y. Jiao, X. Song and M. Chen, 2012a. Optimal training sequences for indoor wireless optical communications. J. Opt., Vol. 14. 10.1088/2040-8978/14/1/015401

Wang, J.B., X.X. Xie, Y. Jiao, X. Song, X. Zhao, M. Gu and M. Sheng, 2012b. Optimal odd-periodic complementary sequences for diffuse wireless optical communications. Opt. Eng., Vol. 51. 10.1117/1.OE.51.9.095002

Yilmaz, A. and M. Shah, 2005. Actions sketch: A novel action representation. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Volume 1, June 20-25, 2005, San Diego, CA., USA., pp: 984-989.