21st International Conference on Pattern Recognition (ICPR 2012)
November 11-15, 2012. Tsukuba, Japan

Human Actions Recognition from Streamed Motion Capture

Mathieu Barnachon* Saïda Bouakaz* Boubakeur Boufama‡ Erwan Guillou*

*Université de Lyon, CNRS
Université Lyon 1, LIRIS, UMR5205, F-69622 France
{firstname}.{lastname}@liris.cnrs.fr

‡School of Computer Science
University of Windsor, Windsor, ON, Canada N9B 3P4
boufama@uwindsor.ca

Abstract

This paper introduces a new method for streamed action recognition using Motion Capture (MoCap) data. First, the histograms of action poses, extracted from MoCap data, are computed according to Hausdorf distance. Then, using a dynamic programming algorithm and an incremental histogram computation, our proposed solution recognizes actions in real time from streams of poses. The comparison of histograms for recognition was achieved using Bhattacharyya distance. Furthermore, the learning phase has remained very efficient with respect to both time and complexity. We have shown the effectiveness of our solution by testing it on large datasets, obtained from animation databases. In particular, we were able to achieve excellent recognition rates that have outperformed the existing methods.

1 Introduction

Human action recognition is a challenging topic that, if solved, would enhance numerous applications in areas ranging from Human Computer Interface (HCI) to entertainment. In particular, new acquisition devices, like Microsoft Kinect and its realtime low cost Motion Capture system [13], can be used to enhance the user's experience with games, serious games, presentation softwares, *etc.* This problem has attracted a great deal of research works the last decades. Fujiyoshi *et al.* [3], is among the pioneering works to identify human actions through human body skeletonization. They have particularly targeted walking, running and even gait analysis. Their solution has the advantage of being easy to use but it is mainly useful for simple interpretations. Bobick and Davis [2] have proposed a method using spatio-temporal templates for human activity recognition. Their method runs in real-time and uses a database of actions they have previously extracted. Although their recognition process is real-time, adding new actions to their database is time consuming and not very flexible. Xiong and Liu [14] have used Hidden Markov Model on extracted silhouettes, but their target was limited to simple human behaviors. Ryoo [11] has proposed a method dealing with integral histograms and bag of words for early action recognition. The author has adapted 2D features to spatio-temporal action, and was able to recognize actions at 50% from their completion with a 50% confidence. As experiments were made on short activities, it would be interesting to perform tests on more challenging datasets, like the well known CMU database [7]. In the work by Lv and Nevatia [6], actions were modeled as sets of virtual key-poses to be used in the matching process. This method is however limited by the high computation cost and by the number of available virtual key poses. Parameswaran and Chellappa [10] use MoCap with markers in an invariant motion space. As mentioned by the authors, there is no 3D invariance in motion space, and therefore all actions have to be described independently. In [4] for example, the authors propose a real-time motion interpretation using simplified MoCap data. However, as they are using a subset of the whole MoCap data, their solution requires a large database of similar actions for correct interpretation. Moreover, a computationally complex and time-consuming preprocessing stage is needed to cluster similar actions. Yao *et al.* [15] have showed that methods based on motion capture data give better results than appearance based solutions when dealing with complex human activities. In [16], they use a stochastic process to find a latent space that discriminates complex human activities. Even if the obtained results seem promising, the computation time, from 0.5 sec to more than 3 sec, makes it far from real-time.

In this paper, we present a new solution, dealing with stream of poses extracted from MoCap data, allowing the recognition of actions in real time. Thanks to dynamic programming optimisation and integral his-

tograms representation, our proposed method yields excellent recognition results, outperforming existing action recognition methods.

2 Histogram-based comparison

In this section, we present our histogram-based method to classify action from Motion Capture. Broadly speaking, let $\mathcal{P}$ be the set of all poses coming from video stream. Let $A = (p_0, \ldots, p_N)$ be an action consisting of a time-ordered sequence of poses, assuming A starts at time t_0 and ends at time t_N. To keep the notations simple and without loss of generality, let's assume that $t_0 = 0$. Geometrically, a pose is represented by a simple human skeleton that consists of a set of 3D joints, with hierarchical relations. Note that because of noise perturbations and speed variation of movements, two different actions may contain some identical poses and instances of the same action may be more or less different. To overcome this problem, all poses composing an action are grouped, based on similarity criteria of their appearances, into a set of clusters. Then, each of these clusters is defined by a representative element (for instance the median element), denoted by $\widetilde{p_j}$. To quantify the similarity between two poses, p_1 and p_2, we use the well know Hausdorff distance [5], denoted D_P hereafter, that provides an elegant way to compare two poses. In order to achieve the clustering mentioned above, we define the following ε-equivalence between two poses.

Definition 1. *Let D_P be a distance between two poses, the ε-equivalence between p_1 and p_2 is given by:*

$$p_1 \sim p_2 \quad \Leftrightarrow \quad D_P(p_1, p_2) \leq \varepsilon \tag{1}$$

where, p_1 and p_2 are in $\mathcal{P}$.

Using the above definition, we introduce the cumulative frequency occurrences of a representative pose $\widetilde{p_j}$ of an action $A = (p_0, \ldots, p_N)$ of length t_N.

$$H_A^t(\widetilde{p_j}) \;=\; |\{p_i\}|, \text{ with } i = 0 \ldots t \text{ and } p_i \sim \widetilde{p_j} \tag{2}$$

where, $t \leq t_N$.

Note that when $t < t_N$, we are considering a restriction of action A to the time lapse $[0, t]$. Such a restriction is useful to us as we are interested in recognizing actions even before they are completed. To do so, we need to evaluate the likelihood over time of the ongoing MoCap data to be one of our previously learned actions. Inspired by [11], we have defined our own integral histogram of actions that we have used to compute this likelihood for our recognition decision.

Definition 2. *A pose-based integral histogram, $\mathcal{IH}$ of action A, is a histogram given by:*

$$\mathcal{IH}^t(A, \mathcal{P}) = \bigcup_{\widetilde{p_j} \in \mathcal{P}} H_A^t(\widetilde{p_j}) \tag{3}$$

To quantify the similarity between two histograms, we use the Bhattacharyya distance [1] to our pose-based integral histograms. This histogram distance, denoted D_H, between two actions A and B is given by:

$$D_H(\mathcal{IH}^{t_A}(A, \mathcal{P}), \mathcal{IH}^{t_B}(B, \mathcal{P})) = \sqrt{1 - \sum_{\widetilde{p_j}} M} \tag{4}$$

With

$$M = \frac{\sqrt{H_A^{t_A}(\widetilde{p_j}) \cdot H_B^{t_B}(\widetilde{p_j})}}{\sqrt{\sum_{\widetilde{p_j}} H_A^{t_A}(\widetilde{p_j}) \cdot \sum_{\widetilde{p_j}} H_B^{t_B}(\widetilde{p_j})}} \tag{5}$$

where, t_A and t_B are the upper-bound times defining the restrictions for action A and action B, respectively.

Using relation 4, we can introduce a cost function to evaluate similarity between two actions A and B.

$$Cost\,(A, B) = D_H\left(\mathcal{IH}^{t_N}(A, \mathcal{P}), \mathcal{IH}^{t_M}(B, \mathcal{P})\right) \tag{6}$$

where t_N and t_M represent the end times (or lengths) for A and B, respectively.

3 Online recognition

Because integral histograms lack the temporal information about poses, we propose to decompose them into multiple smaller size integral histograms, called sub-histograms.

Definition 3. *A sub-histogram of a time lapse $[t_1, t_2]$, denoted by $\mathcal{IH}^{[t_1,t_2]}$, for action A, is given by:*

$$\mathcal{IH}^{[t_1,t_2]}(A, \mathcal{P}) \;=\; \bigcup_{\widetilde{p_j} \in \mathcal{P}} H_A^{[t_1,t_2]}(\widetilde{p_j}) \tag{7}$$

$H^{[t_1,t_2]}$ is the restriction of the cumulative frequency occurrences function to the time lapse $[t_1, t_2]$ with $0 \leq t_1 < t_2 \leq t_N$. Therefore, an action could be considered as a time series of sub-integral histograms, represented by a vector $(h_0, \ldots, h_N)$. Note that before providing a recognition score for an ongoing action, we need to align it, time-wise, with the learned actions. The objective is to obtain the alignment between two time-dependent actions, $A = (ha_0, \cdots, ha_p)$ and $B = (hb_0, \cdots, hb_q)$, respectively. Evaluating the cost

function measure for each pair of the sequences A and B, leads to a $p \times q$ cost matrix. We are looking for the alignment that minimizes the cost between actions A and B. To achieve this, we have been inspired by a well-known Dynamic Time Warping technique, proposed in [12], where the optimal alignment is provided by the following recursive relations.

$$\begin{aligned} Cost^{\star}(ha_0, hb_0) &= Cost(ha_0, hb_0) \\ Cost^{\star}(ha_i, hb_j) &= Cost(ha_i, hb_j) \\ &+ \min \left\{ \begin{array}{l} Cost(ha_{i-1}, hb_j), \\ Cost(ha_i, hb_{j-1}), \\ Cost(ha_{i-1}, hb_{j-1}) \end{array} \right. \end{aligned} \quad (8)$$

Because of the natural variation due to different body proportions and movements style, instances of the same actions can be different. Using a single instance per action as a training set will not usually yield a good recognition rate. To cope with this problem, we translate multiple instances of the same action into multiple histograms. Rather than finding an unified representation of an action among a large dataset, that is a challenging task, we propose to gather similar instances of an action into a single histogram. First, using the cost defined in 6, we calculate the standard deviation with respect to the median histogram. This allows us to define a distance threshold, for example equal to this standard deviation. Then, the Bhattacharyya distances between all histograms are calculated. If the distance between two histograms is below our threshold, only one of them is kept. In other words, given that "very similar" histograms are not better than a single one, we only keep those histograms that represent "different" instances of the same action. The set of these "different" histograms, $\mathcal{A}_H$, represents multiple hypotheses for the same action in the recognition stage. Hence, our recognition score is computed according to the equation given below:

$$Cost_{multi}(A, B) = \min_{A_h \in \mathcal{A}_H} \{Cost^{\star}(A_h, B)\} \quad (9)$$

where, $\mathcal{A}_H$ is the set of multiple hypotheses of action A and B is the ongoing observed action to be recognized.

4 Results

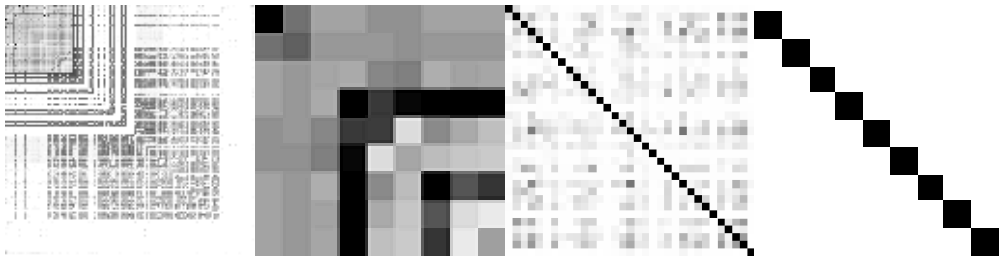
We have tested our system on the HDM dataset of actions from [9]. This dataset consists of 130 classes, obtained from 2337 actions (cuts of longer capture sets), made by 5 different actors. We have considered the two scenarios, single-hypothesis and muli-hypotheses. In the former case, we randomly selected one action instance from each class (random execution, random actor) and used it as training set. Whereas in the latter case, we randomly selected a few action instances from each class and have kept only three to be the training set.

Dataset	Accuracy
HDM(single-hypothesis)	67.89%
HDM(multi-hypothesis)	96.67%
CMU(single-hypothesis)	86.63%
CMU(multi-hypothesis)	90.92%

Table 1: Recognition rates for all datasets.

We have also constructed another dataset consisting of 9 classes out of 53 actions, from the very large CMU dataset [7]. The CMU dataset is more complex as it involves many similar actions. Compared to the results published in [8], where like in this paper, a subset of CMU dataset has been used, our solution has performed much better. Their recognition rate was around 75% whereas, ours are 86.63% and 90.92% for the single-hypothesis and multi-hypotheses, respectively. Our results are also very good compared to the 81.5% recognition rate obtained in [15], which uses 2D features and MoCap information from the TUM dataset.

For both scenarios, all the remaining instances were used for recognition. The single hypothesis results are shown on figures 1a and 1b. In particular, the confusion matrices, computed with $\varepsilon = 1.0$, show that the proposed method highly discriminates between different actions. It also highlights similar actions, such as "Boxing", "Drink" and "eat", as all of them involve hand activity. The multi-hypothesis solution, shown on figures 1c and 1d, is clearly more discriminant, as it allows more intra-class variation. Note that with the multi-hypothesis, the 130 initial classes are reduced to 33 different classes (for example, starting walking with a left foot or with the right one will end up in the same class). More quantitative scores are shown in Table 1, where the proposed method clearly yields excellent results. Although the single-hypothesis recognition rate is outperformed by most previous works, like [8] that yields an average recognition rate of 80% on the same dataset, our fully automatic multi-hypotheses solution performs a lot better at a rate of 96.67%. Note also that in [8], keyframes used in the queries were manually-selected.



(a) HDM (130 classes) (b) CMU (c) HDM multi (33 classes) (d) CMU multi

Figure 1: Confusion matrix for dataset (note that image colors are logarithmic to enforce contrast).

5 Conclusion

This paper proposed a new technique for human action recognition using MoCap data. The method has many advantages over previous ones. Thanks to a dynamic programming algorithm and an incremental histogram computation, our proposed solution recognizes actions in real time from streams of poses, without the need for a pre-segmentation of the input videos. The training phase is also very efficient with close to real-time speed. By expressing the MoCap poses as histograms, we have turned the action recognition into easy distance computations between histograms. Although pose histograms have been used in the two-dimensional case, this paper is the first, to the best of our knowledge, to use them successfully with 3D MoCap data. The obtained results have supported our claim as we have obtained excellent recognition rates, clearly outperforming many recent and well known methods.

References

[1] A. Bhattacharyya. On a measure of divergence between two statistical populations defined by their probability distributions. *Bulletin of the Calcutta Mathematical Society*, 35:99–109, 1943.

[2] A. F. Bobick and J. W. Davis. The recognition of human movement using temporal templates. *IEEE Trans. Pattern Anal. Mach. Intell.*, 23(3):257–267, 2001.

[3] H. Fujiyoshi and A. J. Lipton. Real-time human motion analysis by image skeletonization. *Applications of Computer Vision, IEEE Workshop on*, 0:15, 1998.

[4] L. Han, X. Wu, W. Liang, G. Hou, and Y. Jia. Discriminative human action recognition in the learned hierarchical manifold space. *Image and Vision Computing*, pages 836 – 849, 2010.

[5] F. Hausdorff. *Grundzge der Mengeniehre*. Von Veit, Leipzig, 1914.

[6] F. Lv and R. Nevatia. Single view human action recognition using key pose matching and viterbi path searching. In *CVPR*, 2007.

[7] MoCap CMU. The data used in this project was obtained from mocap.cs.cmu.edu. the database was created with funding from nsf eia-0196217. *http://mocap.cs.cmu.edu/*.

[8] M. Müller, A. Baak, and H.-P. Seidel. Efficient and robust annotation of motion capture data. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 17–26, 2009.

[9] M. Müller, T. Röder, M. Clausen, B. Eberhardt, B. Krüger, and A. Weber. Documentation mocap database hdm05. Technical Report CG-2007-2, Universität Bonn, June 2007.

[10] V. Parameswaran and R. Chellappa. View invariants for human action recognition. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, 2:613, 2003.

[11] M. Ryoo. Human activity prediction: Early recognition of ongoing activities from streaming videos. In *International Conference on Computer Vision*, 2007.

[12] H. Sakoe and S. Chiba. Dynamic programming algorithm optimization for spoken word recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 26:43–49, 1978.

[13] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake. Real-time human pose recognition in parts from a single depth image. In *CVPR*, 2011.

[14] J. Xiong and Z. Liu. Human motion recognition based on hidden markov models. In *Advances in Computation and Intelligence*, pages 464–471, 2007.

[15] A. Yao, J. Gall, G. Fanelli, and L. V. Gool. Does human action recognition benefit from pose estimation? In *Proceedings of the British Machine Vision Conference*, pages 67.1–67.11, 2011.

[16] A. Yao, J. Gall, L. V. Gool, and R. Urtasun. Learning probabilistic non-linear latent variable models for tracking complex activities. In *Neural Information Processing Systems (NIPS)*, 2011.