Sparse human movement representation and recognition

Nikolaos Gkalelis [#†], Anastasios Tefas [†], Ioannis Pitas [#†]

[#] *Informatics and Telematics Institute, CERTH, Greece*
[†] *Department of Informatics, Aristotle University of Thessaloniki, Greece*
{galelis,tefas,pitas}@aiia.csd.auth.gr

***Abstract*—In this paper a novel method for human movement representation and recognition is proposed. A movement type is regarded as a unique combination of basic movement patterns, the so-called dynemes. The fuzzy c-mean (FCM) algorithm is used to identify the dynemes in the input space and allow the expression of a posture in terms of these dynemes. In the so-called dyneme space, the sparse posture representations of a movement are combined to represent the movement as a single point in that space, and linear discriminant analysis (LDA) is further employed to increase movement type discrimination and compactness of representation. This method allows for simple Mahalanobis or cosine distance comparison of movements, taking implicitly into account time shifts and internal speed variations, and, thus, aiding the design of a real-time movement recognition algorithm.**

I. Introduction

There is a bulk of human motion analysis literature. A general review can be found in [1]. In particular, human motion recognition encompasses several levels of complexity. Among the many suggestions for motion categorization, here we adopt the three level taxonomy proposed in [2]. In the lower level, a dyneme, is described as the most constructive unit of motion, while one level above, a movement, is conceived as a sequence of dynemes, with clearly defined temporal boundaries as well as clear conceptual interpretation, e.g., one period of walk.

An important question in human movement recognition is what kind of information should be exploited in order to represent a movement. Most methods in the literature exploit either the local or the global motion information within a sequence of posture images to represent a movement. Local motion information is derived by observing the spatial variation of points in the human body over time, e.g., by feature tracking. Then position and velocity of these points may be used to represent the movement in the input space, e.g., [2], [3]. Global motion refers to the shape configurations that the human body receives through the course of a movement, without considering any point correspondences. Consequently, a movement is represented by a sequence of posture images extracted from the original video, e.g., [4]–[6].

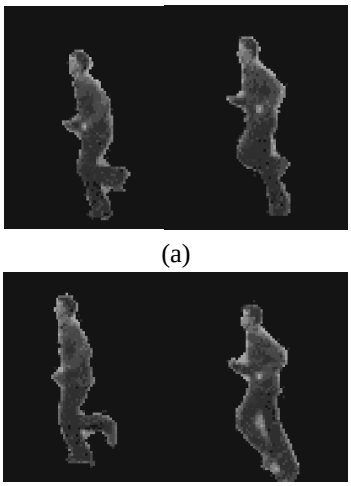
From the classification point of view, most methods mainly fall within two categories, template matching, e.g., [4], [5], and statistical techniques, e.g., [2]. Regardless the classification type, movements are often represented as manifolds in some space and compared using an expensive similarity metric, e.g., Hausdorff distance, to account for different length movements and internal speed variations during the execution of the same movement.

Recent studies in psychophysics, e.g., [7], [8], have suggested that perception of a walking figure may occur from the integration of the body shape over time, i.e., global motion information is mainly responsible for the recognition of the movement in the human visual system, while local motion information has only supportive role in this process, e.g., for motion detection, segmentation and tracking. Inspired from these studies, we represent a movement with a sequence of human posture binary masks, i.e., we exploit the global motion information of a posture image sequence. Moreover, we conceive a movement as a unique combination of dynemes, where each dyneme can be thought as the integration of the temporally neighboring postures of a movement. Using FCM, [9], we identify the dynemes in the input space, and project each posture, to the so-called dyneme space. The sparse movement representations in this space are combined to yield a single vector, encoding its similarity to the identified dynemes. Next, we project the movement vectors with LDA, to increase movement class discrimination and compactness of representation. This representation allows simple Mahalanobis- or cosine-based nearest centroid classification, of movements of variable length. Experimental results show the applicability of the method for human movement recognition.

II. Proposed method

A movement appears in a video as a sequence of human posture frames. We assume that binary masks of the postures are available. These masks are further preprocessed to create regions of interests (ROIs), which have the same dimension, contain as much foreground as possible and are centered in respect to the centroid of the posture. A posture mask ROI is scanned column-wise to produce the so-called *posture vector* $\mathbf{x} \in \Re^F$ and, thus, represent the movement with a spatiotemporal trajectory in the input space $\{\mathbf{x}_i\}$ which is called *movement sequence*.

Movement classes highly overlap, and therefore do not express the actual structure of the input space $\Re^F$. An example of such overlap is shown in Fig. 1, where a naive observer may mistakenly perceive the sequence of skip postures with run. One way to counteract this problem and recognize R different movement types, is to assume that there are $C > R$ postures,



(a)

(b)

Fig. 1. ***(a)*** *Two postures of skip which can be easily confused with postures of run.* ***(b)*** *Two postures of "run".*

the so-called dynemes, which when combined, uniquely characterize the R different movements. Based on this assumption, we apply the FCM algorithm to discover the dynemes, and then project the individual postures to the identified dynemes. In this space, the sparse posture representations of a movement are combined to uniquely characterize each movement. The number of dynemes, the dyneme postures themselves and the fuzzification parameter are identified with the leave-one-out-cross-validation (LOOCV) procedure.

A. Computation of dynemes by FCM

Considering unlabelled posture vectors $\{\mathbf{x}_1, \ldots, \mathbf{x}_P\}$, we apply the FCM algorithm [9] to discover the intrinsic structure of the input space. FCM algorithm is based on the minimization of the following objective function:

$$\mathbf{J}_{FCM}(\mathbf{\Phi}, \mathbf{V}) = \sum_{c=1}^{C}\sum_{\imath=1}^{P}(\phi_{c,\imath})^m(\parallel \mathbf{x}_\imath - \mathbf{v}_c \parallel_2)^2 \ , \quad (1)$$

where, P, C are the number of samples and centroids respectively, $\mathbf{x}_\imath \in \Re^F$ is the $\imath$-th sample in the training data set, $\mathbf{V} = [v_{\jmath,c}] = [\mathbf{v}_1, \ldots, \mathbf{v}_C] \in \Re^{F\times C}$ is the matrix of cluster prototypes, in our case the dyneme representations, $\mathbf{\Phi} = [\phi_{c,\imath}] \in \Re^{C\times P}$ is the partition matrix with $\phi_{c,\imath} \in [0,1]$ the degree that the $\imath$-th sample belongs to the c-th cluster, $m > 1$ is the fuzzification parameter and $\parallel \cdot \parallel_p$ is the p-th vector norm. The FCM criterion (1) is subjected on producing non-degenerate fuzzy partition of the training data at each iteration of the optimization, $\{\mathbf{\Phi} \in \Re^{C\times P} \mid \forall c, \imath : \sum_{c=1}^{C}\phi_{c,\imath} = 1;\ 0 < \sum_{\imath=1}^{P}\phi_{c,\imath} < P;\ 0 \le \phi_{c,\imath} \le 1\}$.

The computation of the cluster centers and partition matrix is carried out through iterative optimization of (1), with the update of membership matrix and cluster centers at each step given by:

$$\phi_{c,\imath} = \frac{(\parallel \mathbf{x}_\imath - \mathbf{v}_c \parallel_2)^{\frac{2}{1-m}}}{\sum_{\jmath=1}^{P}(\parallel \mathbf{x}_\jmath - \mathbf{v}_c \parallel_2))^{\frac{2}{1-m}}}, \quad (2)$$

$$\mathbf{v}_c = \frac{\sum_{\imath=1}^{P}\phi_{c,\imath}^m\mathbf{x}_\imath}{\sum_{\imath=1}^{P}\phi_{c,\imath}^m} \ . \quad (3)$$

The iteration is initialized with an initial estimate of matrix $\mathbf{V}$ or $\mathbf{\Phi}$ and terminates when the difference of the estimated matrix between two iterations is smaller than a specified tolerance ϵ.

B. Movement classification

Let $\mathcal{U}$ be database of movement sequences, where each sequence belongs to one of R different movement classes. Ignoring the sequential information, we represent the n-th sequence of the r-th class with length L_n as a set of posture vectors $\{\mathbf{x}_{n,1}^{(r)}, \ldots, \mathbf{x}_{n,L_n}^{(r)}\}$. The total number of movement sequences in the database is $N = \sum_{r=1}^{R} N_r$, where N_r is the number of sequences of the r-th class.

Given the dyneme vectors and the fuzzification parameter, we can express the posture vectors of a movement in respect to the dynemes, take the linear combination of the respective vectors, and get the so-called *movement vector* $\mathbf{s} \in \Re^C$. Therefore, each movement sequence is represented by the respective movement vector, $\{\mathbf{s}_1^{(1)}, \ldots, \mathbf{s}_{N_1}^{(1)}, \ldots, \mathbf{s}_{N_R}^{(R)}\}$.

If the dimension of the dyneme space is larger than the number of movement classes, i.e., $C > R$, we may exploit the labelling information to further project the movement vectors using a subspace method, and further improve class discrimination and representation compactness. A convenient method for this is linear discriminant analysis (LDA). Most LDA algorithms, e.g., [10], seek for the linear projection $\mathbf{\Psi}_{opt} \in \Re^{C\times R-1}$, that maximizes the criterion $J_{LDA}(\mathbf{\Psi})$

$$J_{LDA}(\mathbf{\Psi}) = \frac{\mid \mathbf{\Psi}^T\boldsymbol{S}_b\mathbf{\Psi} \mid}{\mid \mathbf{\Psi}^T\boldsymbol{S}_w\mathbf{\Psi} \mid} \ ,$$

$$\mathbf{\Psi}_{opt} = \underset{\mathbf{\Psi}}{\operatorname{argmax}}(J_{LDA}(\mathbf{\Psi})) \ . \quad (4)$$

The matrix $\mathbf{\Psi}$ represents a linear transformation, and $\mathbf{S}_w$, $\mathbf{S}_b \in \Re^{C\times C}$, are the within and between scatter matrices respectively.

The rank of $\mathbf{S}_w$ is at most $N - C$, and thus, is invertible if the number of training videos N is adequately larger than the dimension of the dyneme space C. Then, the optimum matrix in (4) is formed by the generalized eigenvectors of $\mathbf{S_w}^{-1}\mathbf{S_b}$, and the projection of the n-th movement vector is given by $\mathbf{y}_n^{(r)} = \mathbf{\Psi}_{opt}^T\mathbf{s}_n^{(r)}$.

Assuming that the movement classes in the dyneme space are derived from unimodal Gaussian distributions with the same covariance matrix $\mathbf{\Sigma}$ but different means $\boldsymbol{\zeta}^{(r)}$, $r = 1, \ldots, R$, we can use the maximum likelihood technique to

estimate them,

$$\boldsymbol{\zeta}^{(r)} = \frac{1}{N_r}\sum_{n=1}^{N_r}\mathbf{y}_n^{(r)}\,, \quad (5)$$

$$\boldsymbol{\Sigma} = \frac{1}{N}\sum_{r=1}^{R}\sum_{n=1}^{N_r}(\mathbf{y}_n^{(r)}-\boldsymbol{\zeta})(\mathbf{y}_n^{(r)}-\boldsymbol{\zeta})^T\,, \quad (6)$$

where $\boldsymbol{\zeta}$ is the total mean ($\boldsymbol{\zeta} = \frac{1}{N}\sum_{r=1}^{R}\sum_{n=1}^{N_r}\mathbf{y}_n^{(r)}$).

If $\mathbf{S}_w$ is not invertible an appropriate LDA variant can be used [11]–[13] . We should also note that before applying LDA, the movement vectors are standardized using the mean and the standard deviation along the training set.

A novel movement sequence is projected in the dyneme space, and the respective movement vector, is standardized and projected with LDA. The unknown movement vector is classified according to the minimum Mahalanobis or maximum cosine distance from the movement class prototypes.

C. Algorithm optimization

LOOCV procedure is utilized to determine the dyneme vectors and the fuzzification parameter. The database may contain more than one instances of the same person performing the same movement. Therefore, at each validation cycle the testing set is formed by all the movement sequences that refer to a specific person and a specific movement class, while the rest of the movement sequences form the training set and used to compute the movement prototypes. The input to the LOOCV procedure is the number of dynemes C and the fuzzification parameter m. To determine the optimum parameters, we follow the global-to-local search strategy similar to [11], [13]. After globally searching over a wide range of the parameter space, we find a candidate interval where the optimal parameters might exist. Then we try to find the optimal parameters within this interval. Application of this procedure is shown in section III-B.

III. Experimental results

The classification database reported in [5] is used to assess the performance of the proposed method. Each video in the database depicts a person executing one or more instances of one particular movement. The binary mask sequences of the videos are provided as well. The mask sequences are imperfect as they were produced from median background substraction and simple thresholding in color space. A few masks of walk, run, skip, side and jump are shown in Fig. 2. We see that the silhouette of jump is highly corrupted. This happens because the color of the trousers is similar to the color of the background.

A. Preprocessing

For non-stationary movements a mask sequence is transformed to show persons moving in the same direction, either left or right. This is done by first deciding the direction, and then mirroring the frames of the videos that show a person moving to an opposite direction from the chosen one.

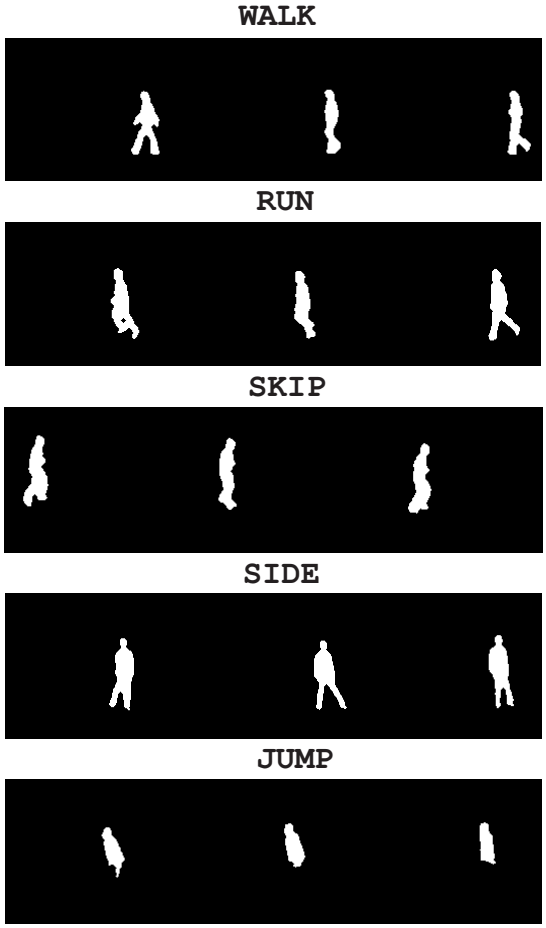


Fig. 2. *Three binary masks for each of the movements walk, run, skip, side and jump.*

From each binary mask, the rectangular region containing the posture is extracted, to form a posture image. All posture mask ROIs are centered according to the center of mass of the posture mask. The resulted images are scaled to the same size, here 64×48, with bicubic interpolation (as in [4]), and are scanned column-wise to form 3072-dimensional posture vectors.

B. Algorithm optimization

The classification database contains nine persons performing ten movements. We select five similar movements to test our method, namely, walk (wk), run (rn), skip (sp), galloping sideways (sd) and jump (jp). The proposed algorithm assumes that the training set contains videos depicting a human executing only a single instance of a movement. In contrary, some videos show a person executing several cycles of a periodic movement. We brake such videos to several videos that show a person executing only one cycle of the movement, the so-called *movement videos*, and thus, we produce a database of 150 such videos. The movement videos comprise variable inter- and intra-class length, for instance, the smallest video of run consists of 10 frames, while the largest video of side, consists of 15 frames respectively.

The LOOCV procedure is used to assess the performance of the algorithm as described in section II-C. The number of corrected classified movement sequences at each LOOCV cycle are summed to compute the classification rate. Extensive

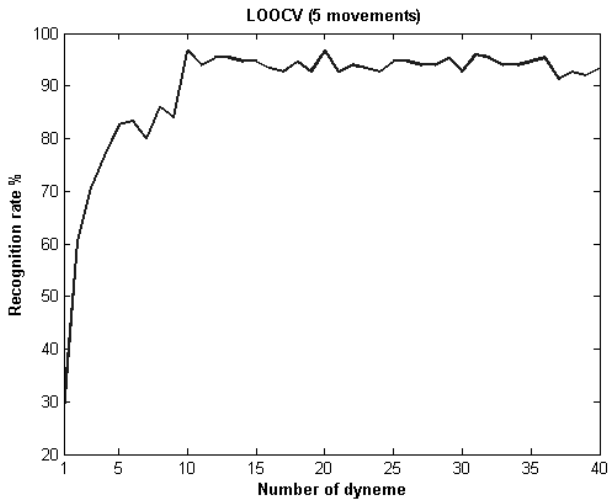


Fig. 3. *Recognition performance in accordance to the number of dynemes, where $m = 1.13$, for five movement classes.*

experiments have been performed to identify the number of clusters C, and fuzzification parameter m. The plot in Fig. 3 depicts the recognition rate of the proposed algorithm, over different number of dynemes C, while the fuzzification parameter is $m = 1.13$. For $C = 10$ dynemes a classification accuracy of 96.7% is achieved, i.e., only 5 movements were misclassified. The respective confusion matrix is given in Table I. We also see that for dynemes $C > 10$, the recognition rate is always above 94%. This shows the representation compactness of our algorithm, as 10 dynemes can already discriminate well the specific 5 different movement classes.

	jp	rn	sd	sp	wk
jp	26			3	
rn		27			1
sd			22		
sp	1			28	
wk					42

TABLE I
Confusion between five movements. A row represents the actual movement and the column the name of the movement recognized by the algorithm during the LOOCV procedure.

IV. Conclusions

A novel human movement representation and recognition method has been proposed. A movement of any length is compactly expressed in terms of its comprising dynemes, as a single vector in a low dimensional space. This representation allows simple cosine- or Mahalanobis-comparison of different movements, avoiding expensive comparison metrics, and thus, offering higher speed and storage efficiency as well as comparable recognition rates with other state of the art methods in the field, e.g., [3]–[5].

A real time algorithm has been also reported in [6]. In this work, the number of clusters are identified using dominant sets. Although the clusters identified there may well represent the intrinsic structure of the input space, it is not guaranteed that they provide the cluster centers that optimally discriminate different movements, as we pursue in our method. For this reason, the recognition rate achieved here outperforms the rate attained in [6].

Currently we are working on the extension of the method for view-independent continuous human movement recognition exploiting a multi-camera infrastructure. Initial results on this direction are promising, showing the applicability of the method for multi-view continuous human movement recognition.

Acknowledgment

The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 211471 (i3DPost).

References

[1] T. B. Moeslund, A. Hilton, and V. Kruger, "A survey of advances in vision-based human motion capture and analysis," *Comput. Vis. Image Understand.*, vol. 104, no. 2, pp. 90–126, Nov. 2006.

[2] R. Green and L. Guan, "Quantifying and recognizing human movement patterns from monocular video images-part I: A new framework for modeling human motion," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 14, no. 2, pp. 179–190, Feb. 2004.

[3] J. Ben-Arie, Z. Wang, P. Pandit, and S. Rajaram, "Human activity recognition using multidimensional indexing," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 8, pp. 1091–1104, Aug. 2002.

[4] L. Wang and D. Suter, "Learning and matching of dynamic shape manifolds for human action recognition," *IEEE Trans. on Image Proc.*, vol. 16, no. 6, pp. 1646–1661, Jun. 2007.

[5] L. Gorelick, M. Blank, E. Shechtman, M. Irani, and R. Basri, "Actions as space-time shapes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 12, pp. 2247–2253, Dec. 2007.

[6] Q. Wei, W. Hu, X. Zhang, and G. Luo, "Dominant sets-based action recognition using image sequence matching," in *Proc. IEEE Int. Conf. Im. Proc. 2007*, vol. 6, San Antonio, TX, USA, Sept. 2007, pp. 1522–4880.

[7] J. A. Beintema and M. Lappe, "Perception of biological motion without local image motion," *Proceedings National Academy of Science*, vol. 99, no. 8, pp. 5661–5663, Apr. 2002.

[8] J. Lange, K. Georg, and M. Lappe, "Visual perception of biological motion by form: A template-matching analysis," *Journal of Vision*, vol. 6, no. 8, pp. 836–849, Jul. 2006.

[9] J. C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*. New York: Plenum, 1981.

[10] K. Fukunaga, *Introduction to Statistical Pattern Recognition, 2nd ed.* New York: Academic Press, 1990.

[11] K. R. Muller, S. Mika, G. Ratsch, K. Tsuda, and B. Scholkopf, "An introduction to kernel-based learning algorithms," *IEEE Trans. Neural Netw.*, vol. 12, no. 2, pp. 181–201, Mar. 2001.

[12] G. Goudelis, S. Zafeiriou, A. Tefas, and I. Pitas, "Class-specific kernel-discriminant analysis for face verification," *IEEE Trans. Inf. Foren. and Secur.*, vol. 2, no. 3, pp. 570–587, Sep. 2007.

[13] J. Yang, A. Frangi, J. Y. Yang, D. Zhang, and Z. Jin, "KPCA plus LDA: a complete kernel fisher discriminant framework for feature extraction and recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 2, pp. 230–244, Feb. 2005.