Temporal Segmentation Boundaries in Motion-Captured Exercise and Dance Computed with an Online Algorithm

Juan Ignacio Mendoza

Abstract—This article describes an online algorithm for finding segmentation boundaries in motion-captured data of human motion. The algorithm is efficient enough to be suitable for real-time applications. Its main component is a distance matrix of kinematic features extracted from a window of observations. The distance matrix is correlated with a checkerboard kernel, resulting in a novelty score whose peaks indicate change-points. The algorithm has three variables that can be manipulated to obtain suitable segments. Experiments were conducted with five motion-capture recordings comprising physical exercises and dance, having a range of perceived complexity. The results were qualitatively analyzed, showing that depending on conditions of data, the number of free variables may be reduced. This algorithm may be used as a pre-computing stage for systems that perform further classification or clustering of the found segments.

Index Terms—Temporal segmentation, human movement analysis, motion capture.

1 Introduction

This article presents a study concerned with temporal segmentation of motion capture data. Motion capture refers to any kind of system and process for measuring the movement of the human body. Such systems have a wide range of applications. In the audiovisual industry, data recorded from motion capture systems is applied to animate computer-generated characters; in biomechanics research, the systems are used for clinical diagnosis and rehabilitation; and in the performing arts, the systems are used to control video and audio in real time. These systems sense movement of the human body, and encode the movement as time-series data. This is achieved by the use of one or more of a variety of sensing technologies such as laser-scanning, electromagnetic sensors, mechanic sensors including rotation sensors, inertial measurement units, acoustic sensors, stereoscopic video cameras, two-dimension-plus-depth video cameras, and two-dimension video cameras [1], [2]. This study focuses on motion capture using infrared static video cameras pointing to a subject's performing area from different angles. The cameras track the position of reflective markers placed on the subject.

The automatic analysis of human motion capture data is a thriving area of research. One aspect of it is temporal segmentation, consisting in dividing data into segments so that each segment has some degree of internal homogeneity and at the same time has some degree of dissimilarity with other segments. This process has been recognized as one crucial step for learning and recognition of meaningful motions [3]. An example of a practical application is to organize large databases of motion capture data. This is often done with offline segmentation methods [4]. Another application is in robotic systems that learn human gestures for interaction with people, which requires an online segmentation method [5]. The segmentation process usually comprises two stages, the first of which is to find where the data changes from one regime to another, by finding the boundaries of the segments. The second stage may be a process to classify the found segments by comparing them with previously given examples, which is supervised learning. Alternatively, the second stage may be an unsupervised learning process, consisting in clustering the found segments. However, some methods may perform these two stages in a single process.

The study presented in this article has developed and tested a method for online and unsupervised computation of temporal segmentation boundaries in motion-capture data of continuous human movement. The structure of this article is the following: It begins with a review of research on human perceived temporal segmentation, automatic segmentation systems focusing on online unsupervised segmentation methods and strategies for assessment of effectiveness. Then, a brief introduction is given to previous work that has used the main principle of the method that this study has developed. Following the review there is a detailed explanation of the method developed by this study and then a description of experiments to test it. The results of the experiments are discussed and the contributions of this study are elaborated, proposing directions for future work. The main contribution of this study is an alternative online method for finding segmentation boundaries in motion-captured human motion, which is suitable for real-time application.

• *The author is with the Department of Music, Art and Culture Studies, University of Jyväskylä, 40014, Finland.*
E-mail: juigmend@student.jyu.fi

The original manuscript was produced in May of 2019. This Version is of December of 2022.

2 Related Work

2.1 Perceived temporal segmentation

Temporal segmentation is defined as the perception of distinct successive motions in time. These perceived units of motions are semantically meaningful as they help understand what the observed subject is doing. Studies on perceived visual temporal segmentation have consisted of experiments in which people, called ”annotators” indicate the time location of segment boundaries in a film. These studies have asked the annotators to indicate boundaries of segments in two levels of granularity: coarse and fine. Newtson [6] employed video recordings of a subject performing a sequence of actions such as ”seated writing”, ”standing up”, ”walking”, and ”lighting a cigarette”. Zacks, Tversky and Iyer [7] used video recordings of a subject performing the activities ”making a bed”, ”doing the dishes”, ”fertilizing a houseplant”, and ”assembling a saxophone”. Hard, Tversky and Lang [8] used animations of abstract figures interacting with one another and with static figures, where moving figures performed activities ”chase” and ”hide and seek”. In these studies the agreement among annotators was deemed high, although upon scrutiny it may be argued that in fact the agreement ranged from weak to slightly high, with a majority of results showing moderate agreement. Also, they observed that coarse-granularity boundaries matched fine-granularity boundaries, suggesting that the latter are sub-divisions of the former and vice-versa. Kahol, Tripathi and Panchanathan [9] did a similar experiment, in which two choreographers were presented with videos showing a dance routine and were asked to indicate segmentation points. The points annotated by one choreographer were significantly different to the other, with most segments being significantly longer. This suggests that each choreographer had his or her own strategies for segmentation, thus it was not possible to establish that either was the correct one or better. Following the same experimental paradigm, Bläsing [10] used stimuli composed of dance movements, and observed that previous knowledge of the movements had an influence on the resulting segmentation. The segments indicated by professional and amateur dancers tended to be of coarser granularity than those indicated by non-dancers. Likewise, Zacks, Kumar, Abrams and Mehta [11] observed that perceived segmentation boundaries depend on context information when available, and on kinematics (i.e., movement features such as velocity, acceleration, etc.) if no context is available. Additionally, fine-granularity segmentation was found to correlate more on kinematics than coarse-granularity segmentation. The studies mentioned above are representative of the current literature on perceived segmentation of human motion, and their findings are usually taken in consideration directly or indirectly, when developing methods for automatic segmentation of human motion. In summary, perceptual temporal segmentation is hierarchical (i.e., smaller units of motion can be grouped in larger motions and vice-versa) and depends on context. These two principles are to be taken into account when designing algorithms intended to emulate human perception.

2.2 Emulating perceived temporal segmentation

In the literature about research on automatic systems for human motion segmentation, motions at a coarse level of granularity have been called gestures [12] or activities [13], [14]. At a finer level of granularity, the motions considered to be quantum units have been called primitives [15]. Krüger, Kragic, Ude and Geib [16] proposed a framework in which motions with semantic meaning are composed by shorter coherent units, thus formulating segmentation of human motion in two levels of granularity. Other studies have proposed segmentation in three levels of granularity. Bernard et al. [4] proposed a model consisting of kinematic features, single patterns and groups of patterns. “Kinematic features” refers to any of the univariate time series that can be derived from motion (e.g., position or angle of each articulation, velocity, etc). Dreher, Kulp, Mandery, Wächter and Asfour [17] proposed a model composed of perceptual granularities. In that model, coarse granularity is composed of activities (e.g., jumping, walking), medium granularity is composed by actions (e.g., step with left foot, step with right foot) and fine granularity is composed by motion primitives (e.g., lift a foot for a step, return the foot to the floor). However, there are no published studies that have extensively tested the perceptual validity of a fixed number of granularity levels.

Another challenge that research has faced is the different ways in which motion segments may be concatenated. One possibility is that there might be a moment between the end of one meaningful segment and the beginning of the next, in which motion does not correspond to either segment. This has been deemed to be a transition. For example, the segmentation method described by Krüger et al. [18] excludes segments that are transitions between semantically meaningful segments. Another possibility is that segments are co-articulated [19], which occurs when a distinct motion overlaps with the previous or the next. Conversely, a transition or overlap may be short enough to be perceived as instantaneous. Most studies on automated systems for segmentation of human motion have aimed to find segmentation boundaries that are instantaneous, even if in fact there were transitions or overlaps. This treatment may result in transitions detected as proper segments. For the case of overlaps, a segmentation boundary might be placed somewhere in the middle of the overlap section, or the overlap section may be identified as a segment, or even both overlapping segments may be merged.

2.3 Automatic temporal segmentation of human motion

Automatic methods for segmentation of motion-captured continuous human movement can be classified as supervised or unsupervised. Supervised methods require examples of motions to compare with the data intended to segment [20], [21], [22], [23], [24], [25]. Unsupervised methods detect motion units that are not known in advance [18], [26], [27]. These methods can be further classified as offline or online. Offline methods find unknown segments taking into account the characteristics of the whole data. Online methods perform linear search on data, segmenting according to the similarity of observations (e.g., data frames) within a

neighborhood range. Also, online systems may be suitable for real-time applications as long as computation of results is faster or as fast as the frame-rate of real-time data. The next sub-section provides an overview of online methods for unsupervised segmentation of human movement.

2.4 Online unsupervised methods

This sub-section provides an overview of methods intended to be used in online unsupervised segmentation of human motion-capture data for the full body or selected limbs that may represent an activity (e.g., legs and arms move the most when walking or running). The focus is on the methods for finding segmentation boundaries, regardless of the techniques they may use to classify or cluster found segments. Barbič et al. [27] described two online segmentation methods for motion capture data encoded as joint-angles represented as quaternions. The first method applies Principal Component Analysis to a sliding window of data, of which the highest principal components are retained to build a model. The model is applied to the subsequent data frames and an error score is computed between the model reconstruction and the observation data. A segment boundary is recorded at the point where the reconstruction error is over a threshold. Granularity can be adjusted with the size of the window, the number of principal components to use for the model and a threshold for the reconstruction error. The second method is a probabilistic extension of the first method assuming that each segment has a coherent Gaussian distribution. A segmentation boundary is recorded at the point where the distribution changes over a threshold. Experiments were conducted to test both methods, using coarse motions like "walking", "running", "sitting", "standing idle", "exercising", "climbing", "performing martial arts", "washing a window", and "sweeping a floor".

A significant amount of methods for online segmentation use Hidden Markov Models (HMM). Kohlmorgen and Lemm [28] formulated a method that also assumes that a segment has a coherent Gaussian distribution. The distributions are calculated upon windows whose size is incremented. The probability density functions are used to train a HMM. The optimum state of the model is computed with a modified Viterbi algorithm and a transition of states is regarded as a segmentation boundary. This system was applied by Kulić, Takano and Nakamura [29] to segment human motion, improving it by adding the capability to generate new templates from identified segments that are used to train new HMM models. If the average Kullback-Leibler divergence between a new model and an existing model is below a threshold, the new model is merged with the existing one; otherwise it is added to a collection of models organized as a tree structure that allows observing motions at different levels of granularity. The algorithm was tested with motion-capture data encoded as position coordinates of markers. The performed motions used for the test were "punching", "walking in place", "squatting", "kicking" and "arm-raising". A further development based in HMM was described by Takano and Nakamura [5], in which short movements of a fixed length are encoded in HMM which are optimized by the Baum-Welch algorithm and then used to predict movements in data. High prediction error indicates segmentation boundaries. Experiments were conducted on motion-captured data represented as vectors containing joint angles, vertical bodily position, roll, pitch, and their corresponding velocities. The algorithm was tested setting the length of the HMM to 5 frames, using motions such as "punch", "bend" and "kick". This data was iteratively fed to the algorithm, which incrementally learned segmentation boundaries. Fine-tuning of several parameters was required to obtain satisfactory results. Yet another method using HMM, the one proposed by Taniguchi, Hamahata and Iwahashi [30] was not explicitly intended for online application. This method uses a Hierarchical Dirichlet Process prior to control the HMM. Later it was developed into an online system by Bargi, Da Xu and Piccardi [31], comprising an initial bootstrapping learning phase followed by an adaptive online learning phase in which the values of parameters are refined on each iteration of incoming data. Experiments were conducted to segment data encoded as joint coordinates of torso and hands, of motions such as "reaching", "grasping" and "carrying".

While methods based in HMM seem to be the most popular, other approaches have been explored. Gong, Medioni and Zhao [32] devised a method called Kernelized Temporal Cut, which is a temporal application of Hilbert space kernel embedding and kernelized two-sample test. The granularity of this method's segmentation depends on the length of a window in which the kernel function is evaluated. A segmentation boundary is deemed to happen at a point where the function reaches a minimum. Experiments were conducted to segment motion-captured data encoded as joint angles quaternions, for recordings comprising motions such as "hand-shaking", "sitting-down", "jumping", "picking-up", "boxing", "waving", "running" and "walking". Xia, Sun, Feng, Zhang and Liu [33] found that this method performed poorly in comparison with other methods. Gharghabi et al. [34] described a method that evaluates the similarity between all fixed-length windows within a bigger window. A segmentation boundary is recorded where the similarity is minimal. Also, this method assumes that each segment will be composed of at least two instances of a periodic motion. The method was tested with data produced by different kinds of motion-capture technologies, including coarse granularity cyclic motions such as "basketball-forward-dribble", "walk-with-wild-legs" and "normal-walk", using data of reflective markers placed on one arm and one leg.

These methods do not require templates for learning, but they do require the adjustment of variables to achieve segmentation boundaries at a desired level of granularity, to prevent missing boundaries or to prevent the inclusion of undesired boundaries. This adjustment, whether it is manual or automatic, arguably depends on the context in which a method is applied.

2.5 Assessment of Effectiveness

To measure the effectiveness of segmentation algorithms, most of the published studies have relied at least to some extent in the classic measures *precision*, *recall* and *accuracy*, by comparing ground-truth boundaries (annotated by one or

more humans) with computed boundaries. Those measures work well for discrete classification problems in which the options are either "match" or "not a match" between a computed result and a result in the ground truth. However, the nature of temporal segmentation is not necessarily a discrete classification problem. To that extent, Dreher et al. [17] note that a computed segmentation boundary being slightly off the ground-truth should be counted as a match. This is usually solved by establishing a window around each ground-truth boundary. Dreher et al. propose a method that involves a window weighted with a Gaussian (i.e., normal) distribution. However, the problem with this approach is that the Gaussian window's width is fixed while there is no certainty that any given width will correspond to the true distribution of the data, for all boundaries. This is because there is no known method to generalize the temporal length of the transition from one motion to another. Although not tested with motion capture, the method described by Gharghabi et al., [34] consists of a score that measures the temporal distance between each computed boundary to the closest boundary in the ground-truth. All the distances are added and then divided by the total time. However, this score does not penalize extra or missing computed boundaries. Mendoza [35], and Mendoza and Thompson [36] proposed a similarity score that measures the distance between ground-truth and computed boundaries as in the method by Gharghabi et al., but also penalizes missing or extra computed boundaries. Lin, Karg and Kulić [15] describe another approach for evaluation of results, in which all frames in the ground-truth segments are labeled and the number of frames in the computed segments corresponding to the ground truth-labels constitute the measure of similarity. This last method may be appropriate for classification but it might be too restrictive for evaluating only the boundaries, as boundaries of short false computed segments (e.g., transitions between motions) will break the continuity of parallel labeling resulting in a very high dissimilarity score.

The studies cited in the three preceding sections have directed their efforts to developing algorithms that use fixed heuristics to replicate human perception at one level of granularity, comparing results of the algorithms to one ground-truth for each motion recording. An additional challenge appears when the aim of a segmentation system is to replicate human perception and one ground-truth is not enough to describe human perception. As mentioned previously segmentation depends on context, meaning that there might be different ground-truths for the same motion data. This has not been a problem for the studies mentioned here, because they attempted to segment fairly simple motions. For more complex motions such as dance, relying on a single ground-truth might not fully serve to the assessment of a segmentation algorithm. To conclude this section, it appears that there is no known quantitative method or strategy to properly evaluate the results of automatic segmentation of complex human motion. Thus, qualitative assessment based in observation and description seems to be a valid alternative.

2.6 Boundaries of self-similarity checkerboard patterns

Automatically identifying the location of segmentation boundaries in motion data has been posed as a multivariate change-point detection problem [18], [26], [32], [37]. The detection of change-points in motion data can be seen as equivalent to novelty detection, which is the identification of abrupt changes in data by a system, without training of the system [38]. Foote [39] described a method suitable for finding segmentation boundaries in musical audio signals. This method exploits the characteristic checkerboard patterns that can be observed in a self-similarity distance matrix of audio features through time, by correlating a checkerboard kernel along the diagonal of the matrix. This results in a novelty score that indicates the rate of change in data. The peaks of the novelty score indicate change-points that correspond to perceived changes in the music. The granularity of the novelty score is adjusted with the width of the kernel and relevant peaks can be selected over a threshold. Later, Foote and Cooper [40] described the use of that method for segmentation of video. Self-similarity distance matrices also display distinctive patterns for different human movements encoded as motion-capture trajectories [41]. In the study presented in this article, the principle of correlating a checkerboard kernel with a distance matrix is used to find segmentation boundaries in motion-captured data. Furthermore, the method is described as an online algorithm suitable for real-time application.

3 Method for Computing Temporal Segmentation Boundaries

This section describes the method developed in this study, for finding temporal segmentation boundaries in motion-captured data. First, an explanation is given of the algorithm in its offline form and its improvements upon the algorithm described by Foote [39]. Then, an online version of the algorithm is presented. To be consistent with the software implementation, throughout the rest of this article the indexing of vectors and matrices starts at one. The input to the algorithm is marker-based motion capture data, which is a sequence M of m frames F ordered in time, represented by the matrix $M \in \mathbb{R}^{m \times n}$, so that $M_{1:m} = [F_1, F_2 ... F_m]^\mathsf{T}$. Each frame F at temporal index $t \in \{1...m\}$ is represented by vector $F_t = (p_1^t, p_2^t ... p_n^t) \in \mathbb{R}^{1 \times 3n}$, where $p_i^t = (p_{i,x}^t, p_{i,y}^t, p_{i,z}^t)$ is the coordinate of the i^{th} marker in $\mathbb{R}^3$, respect to a world origin. Each recording is composed of a different number of frames and markers. The output of the algorithm is a vector containing the locations t of change-points that are deemed to be segmentation boundaries. The free variables n_{nov} and n_{filt} are used to adjust granularity, while the free variable θ, sets a threshold to select relevant boundaries.

First, M may be down-sampled to reduce computational time. Also, it may be replaced by a matrix where each frame F_t contains one or more motion features computed from position data (e.g., velocity, acceleration, etc.). In the experiments reported further in this article, an approximation to velocity by numerical differentiation produced satisfactory results across all tested recordings. Additionally, numerical differentiation has a very low computational cost, thus being

suitable for real-time computing. Hence, p_i becomes the approximate velocity of the i^{th} marker.

In the next step of the offline algorithm, distance matrix $S \in \mathbb{R}^{m \times m}$ is produced for data in M, where the distance measure is d.

$$S_{j,k} = d(F_j, F_k) \tag{1}$$

Although different distance measures can be used, the one used in the experiments reported here is euclidean.

$$d(F_j, F_k) = \sqrt{\sum_{i=1}^{n} \left((p_{i,x}^j - p_{i,x}^k)^2 + (p_{i,y}^j - p_{i,y}^k)^2 + (p_{i,z}^j - p_{i,z}^k)^2 \right)} \tag{2}$$

A two-dimension checkerboard kernel K_c of width $n_{\text{nov}} \in \mathbb{Z}^+, (-1)^{n_{\text{nov}}} = 1$, is produced by the Kronecker product of checkerboard matrix C and only-ones matrix J of width $\frac{n_{\text{nov}}}{2}$.

$$C = \begin{bmatrix} -1 & 1 \\ 1 & -1 \end{bmatrix} \tag{3}$$

$$K_c = C \otimes J \tag{4}$$

K_c is then tapered by multiplying it element-wise with Gaussian G_{nov} having minima zero and unit volume.

$$G_{x,y}^* = e^{-\left(\frac{x^2+y^2}{2\sigma_{\text{nov}}^2}\right)}, x = y = 1, 2, ..., n_{\text{nov}} \tag{5}$$

$$G^{**} = G^* - \min(G^*) \tag{6}$$

$$G_{\text{nov}} = \frac{1}{G^{**}} \sum_{x=1}^{n_{\text{nov}}} \sum_{y=1}^{n_{\text{nov}}} G_{x,y}^{**} \tag{7}$$

$$K_{\text{nov}} = K_c \circ G_{\text{nov}} \tag{8}$$

In equations (4) to (8), n_{nov} needs to be even to produce the checkerboard matrix. Next, K_{nov} is correlated along the diagonal of S, resulting in the novelty score N_{nov}.

$$N_{\text{nov}_t} = \sum_{x=1}^{n_{\text{nov}}} \sum_{y=1}^{n_{\text{nov}}} K_{\text{nov}_{x,y}} S_{t+x,t+y}, \tag{9}$$
$$t = \frac{n_{\text{nov}}}{2} + 1, \frac{n_{\text{nov}}}{2} + 2, ..., m - \frac{n_{\text{nov}}}{2}$$

From this point improvements to the algorithm described by Foote [39] are presented. The kernel K_{nov} tapered towards zero prevents artifacts at the borders, while its unit volume preserves the scale of data in S. Vector N_{nov} has a lag respect to M of half the width of kernel K_{nov}. N_{nov} may have peaks that could be considered noise. Smoothing M with a low-pass filter can solve this. However, M may be highly dimensional, and thus the filtering process may take considerable computational time. Instead, smoothing can be done upon N_{nov}, which is only one-dimensional. The filter can be simply a one-dimension Gaussian kernel K_{filt} of width $n_{\text{filt}} \in \mathbb{Z}^+, (-1)^{n_{\text{filt}}} = -1$, with minima zero and unit area to prevent artifacts at borders and to preserve scale, respectively.

$$K_{\text{filt}_z}^* = e^{-\left(\frac{z^2}{2\sigma_{\text{filt}}^2}\right)}, z = 1, 2, ..., n_{\text{filt}} \tag{10}$$

$$K_{\text{filt}}^{**} = K_{\text{filt}}^* - \min(K_{\text{filt}}^*) \tag{11}$$

$$K_{\text{filt}} = \frac{1}{K_{\text{filt}}^{**}} \sum_{z=1}^{n_{\text{filt}}} K_{\text{filt}_z}^{**} \tag{12}$$

$$N_{\text{filt}_t} = \sum_{x=1}^{n_{\text{filt}}} K_{\text{filt}_z} N_{\text{nov}_{t+z}}, \tag{13}$$
$$t = \left\lceil \frac{n_{\text{filt}}}{2} \right\rceil, \left\lceil \frac{n_{\text{filt}}}{2} + 1 \right\rceil, ..., m - \left\lceil \frac{n_{\text{filt}}}{2} \right\rceil$$

In equations (10) to (13) the kernel's width n_{filt} is odd so that it has one maximum. This helps to produce novelty peaks defined at a single index, eliminating uncertainty if local maxima was defined at two indexes. In equations (5) and (10) the relation between the Gaussian's standard deviation and width is controlled by parameters α_{nov} and α_{filt} respectively, as follows:

$$\sigma_{\text{nov}} = \frac{n_{\text{nov}} - 1}{2\alpha_{\text{nov}}} \quad \text{and} \quad \sigma_{\text{filt}} = \frac{n_{\text{filt}} - 1}{2\alpha_{\text{filt}}}. \tag{14}$$

The filtered novelty score N_{filt} has lag $l = \frac{n_{\text{nov}} + \lceil n_{\text{filt}} \rceil}{2}$ respect to M. The peaks of N_{filt} indicate change-points in M. Vector $P_{\text{ind}} \in \mathbb{Z}^{m+l}$ is a logical index for all peaks, vector $P_{\text{val}} \in \mathbb{R}^{m+l}$ has the values of each peak and vector $P_{\text{sel}} \in \mathbb{R}^{m+l}$ has the values of peaks equal or over a threshold θ.

$$P_{\text{ind}_{t+l}} = \begin{cases} 1, & N_{\text{filt}_{t+l-1}} < N_{\text{filt}_{t+l}} > N_{\text{filt}_{t+l+1}} \\ 0, & \text{else} \end{cases} \tag{15}$$

$$P_{\text{val}} = P_{\text{ind}} \circ N_{\text{filt}} \tag{16}$$

$$P_{\text{sel}_{t+l}} = \begin{cases} P_{\text{val}_{t+l}}, & P_{\text{val}_{t+l}} \geq \theta \\ 0, & P_{\text{val}_{t+l}} < \theta \end{cases} \tag{17}$$

Vector $B \in \mathbb{R}^h$ has the values for the time indexes t of the h selected peaks indicating segmentation boundaries.

$$B_b = (t \; \forall \; P_{\text{sel}} \neq 0) - l, \quad b = 1, 2, ..., h \tag{18}$$

The correlations in equations (9) and (13) produce vectors having the length of the input matrix or vector, minus the length of the kernel, resulting in undefined values at the beginning and ending. To counteract this, the input matrix M and the novelty score N can be zero-padded, which will result in peaks at the beginning and ending of N_{filt} if data in M is non-zero at the beginning and ending, respectively. Alternatively, the similarity matrix $S \in \mathbb{R}^{m \times m}$ can be replaced by matrix $T \in \mathbb{R}^{q \times q}$, where $q = m + n_{\text{nov}}$. T is constructed so that it starts as null matrix $T = 0_{q,q}$. Then, square sub-matrices of size n_{nov} are added at the beginning and ending corners, each of which contains the average of the corresponding corner sub-matrix in S.

$$T_{x,y} = \frac{1}{n_{\text{nov}}^2} \sum_j \sum_k S_{j,k}, \tag{19}$$

$$x = y = \begin{cases} 1, 2, ..., n_{\text{nov}} \\ q - n_{\text{nov}} + 1, q - n_{\text{nov}} + 2, ..., q \end{cases}$$

$$j = k = \begin{cases} 1, 2, ..., n_{\text{nov}} \\ m - n_{\text{nov}} + 1, m - n_{\text{nov}} + 2, ..., m \end{cases}$$

The values of are S inserted into T, replacing values in the specified range:

$$T_{x,y} = S, \tag{20}$$

$$x = y = \frac{n_{\text{nov}}}{2} + 1, \frac{n_{\text{nov}}}{2} + 2, ..., m - \frac{n_{\text{nov}}}{2}.$$

The correlation between K_{nov} and the diagonal of zero-padded matrix T will produce a novelty score having an increasing curve at the beginning and a decreasing curve at the end. Additionally, computation time can greatly be reduced by computing only half of the values in the summations of equation (9). Only values for $x > y$ need to be computed because the similarity matrix S and the kernel K_{nov} are symmetric. Likewise, S does not need to be computed entirely, but only the strip of width $n_{\text{nov}} - 1$ starting from the first diagonal after the main diagonal.

The algorithm described above is offline, because S (alternatively T) is computed for the whole data. An online implementation would have a distance matrix of size n_{nov} computed for every frame F_t and a system of dynamic buffers to compute differentiation, element-wise inner products with kernels, and to test for peaks. Algorithm 1 describes the online process, computing the approximate first derivative (velocity) for every input frame F_t. In this algorithm, buffers are D for differentiation of raw data, V for computing the distance matrix from differentiated frames, W for applying the Gaussian filter to the novelty score and P for testing local maxima in the novelty score.

The asymptotical computational complexity of the algorithm is quadratic and computation time depends on the sampling rate, width of the kernels (n_{nov} and n_{filt}) and number of markers (n). Additionally, computation time of steps 10 to 12 of Algorithm 1 can be reduced by computing half of S, either the upper or lower triangle. The resulting triangle array can be represented as vector $S*$ to easily compute its inner product with the corresponding triangle array of K_{nov} represented as vector K^*_{nov}. Also there is no need to compute the whole triangle for each new frame F_t. It is only needed to initialize matrix S with zeros, then compute the distance d of $F'_t = V_{n_{\text{nov}},n}$ with the previous $n_{\text{nov}} - 1$ frames in V. The values within S should be shifted before any new F'_t, discarding the distances between the oldest frame in V and the newer ones. This approach shifts memory indexes instead of doing redundant computations, as shown in Algorithm 2.

The online and offline algorithms have been implemented in the Matlab programming environment. The program calls some functions of the Mocap Toolbox [42] that have been modified. All steps belonging to the algorithms described in this article are coded in the program's script. [1]

Algorithm 1 : Online Computation of Boundaries

Init parameters: $n_{\text{nov}}, \alpha_{\text{nov}}, n_{\text{filt}}, \alpha_{\text{filt}}$
Init: $K_{\text{nov}}, K_{\text{filt}}, D = 0_{2,3n}, V = 0_{n_{\text{nov}},3n}, W = 0_{n_{\text{filt}}}$, $P = 0_3, l, b = 0$
Input: F_t, θ
Output: B_b

1: $\{D_{1,y}\}_{y=1}^{3n} \leftarrow \{D_{2,y}\}_{y=1}^{3n}$
2: $\{D_{2,y}\}_{y=1}^{3n} \leftarrow F_t$
3: **if** $t = 1$ **then**
4: $\quad F'_t \leftarrow 0_{1,3n}$
5: **else**
6: $\quad F'_t \leftarrow \{D_{2,y}\}_{y=1}^{3n} - \{D_{1,y}\}_{y=1}^{3n}$
7: **end if**
8: $\{V_{x,y}\}_{x=1,y=1}^{n_{\text{nov}}-1,3n} \leftarrow \{V_{x,y}\}_{x=2,y=1}^{n_{\text{nov}},3n}$
9: $\{V_{n_{\text{nov}},y}\}_{y=1}^{3n} \leftarrow F'_t$
10: $S \leftarrow d(V)$
11: $\{W_x\}_{x=1}^{n_{\text{filt}}-1} \leftarrow \{W_x\}_{x=2}^{n_{\text{filt}}}$
12: $W_{n_{\text{filt}}} \leftarrow \text{sum}\big(\text{sum}(K_{\text{nov}} \circ S)\big)$
13: $\{P_x\}_{x=1}^{2} \leftarrow \{P_x\}_{x=2}^{3}$
14: $P_3 \leftarrow K_{\text{filt}} \cdot W$
15: **if** $\text{argmax}(P) = 2$ **and** $P_2 \geq \theta$ **then**
16: $\quad b \leftarrow b + 1$
17: $\quad B_b \leftarrow t - 1 - l$
18: **end if**

Algorithm 2 : Efficient Distance Matrix and Novelty Score

Init: $\left\{K^*_{\text{nov}_x}\right\}_{x=1}^{v} \leftarrow \left\{K_{\text{nov}_{x,y}}\right\}_{x=1,y=x+1}^{n_{\text{nov}},n_{\text{nov}}}, v = \frac{n_{\text{nov}}(n_{\text{nov}}-1)}{2}$
$S^* \leftarrow 0_v, S \leftarrow 0_{n_{\text{nov}},n_{\text{nov}}}$
Input: V, W
Output: $W_{n_{\text{filt}}}$

1: $\{S_{j,k}\}_{j=1,k=j+1}^{n_{\text{nov}}-1,n_{\text{nov}}} \leftarrow \{S_{j,k}\}_{j=2,k=j+1}^{n_{\text{nov}},n_{\text{nov}}}$
2: **for** $c = 1, ..., n_{\text{nov}} - 1$ **do**
3: $\quad S_{c,n_{\text{nov}}} \leftarrow d\left(\{V_{c,y}\}_{y=1}^{n}, \{V_{n_{\text{nov}},y}\}_{y=1}^{n}\right)$
4: **end for**
5: $\{S^*_x\}_{x=1}^{v} \leftarrow \{S_{j,k}\}_{j=1,k=j+1}^{n_{\text{nov}},n_{\text{nov}}}$
6: $\{W_x\}_{x=1}^{n_{\text{filt}}-1} \leftarrow \{W_x\}_{x=2}^{n_{\text{filt}}}$
7: $W_{n_{\text{filt}}} \leftarrow K^*_{\text{nov}} \cdot S^*$

4 EXPERIMENTS

4.1 Motion-Capture Data

The method presented in the previous section was tested with five recordings of trajectory-based optical motion-capture data. The motions captured in these recordings range from lower to higher complexity, given by the subjective number of different motions and their overlapping. All recordings feature one person and are publicly available. Recordings 1, 4 and 5 were recorded at the Department

1. Software and motion-capture recordings 1, 4 and 5 are provided with the online version of this article. Alternatively, they may be requested to the author of this article.

of Music, Art and Culture Studies of the University of Jyväskylä. [1] Recordings 2 and 3 were taken from the CMU database [43] and have been used in previous research. The following paragraphs present descriptions of each recording, which in the "Results" section are compared with the segmentation boundaries computed with the online algorithm. Detailed descriptions were made for recordings 1 to 3 as these contain fairly simple motions. Recordings 4 and 5 contain complex motions and overall descriptions have been made of them, as detailed analysis would be too lengthy for this report.

Recording 1 has duration of 52.5 seconds. The subject recorded is a beginner practitioner of Escrima, a traditional martial art from the Philippines that incorporates the use of blunt weapons such as sticks. The subject holds a stick on each hand. The recorded motions at coarse granularity are referred to as *weaving sticks* and *rest*. The *weaving sticks* motion is composed by several repetitions of the Heaven Six exercise, which is a sequence of six motions. The exercise can be further divided in two sequences of three hits, where there is first a hit with one stick, then a hit with the other stick and then a hit with the first stick. The sequence of three motions is further referred to as *hit-hit-hit*. This sequence is repeated in a mirrored fashion, meaning that they are performed to either side, then to the opposite side and so on. Thus, they are further referred to as *hit-hit-hit left* and *hit-hit-hit right*. The recording consists of three *weaving sticks* segments of about 15 seconds each, interrupted by *rests* of about 2 seconds each. At each *rest* the subject *chambers* the sticks by holding them to one side pointing back, with the corresponding arm holding its stick over the shoulder and the opposite arm holding its stick just below the other arm's elbow.

Recording 2, with duration of 38.7 seconds is trial 20 for subject 14, also from the CMU database. It has been previously used by Nakamura et al. [44] to test a segmentation algorithm. The recorded subject performs a sequence of six motions at coarse granularity: *jumping, twisting, raising knees, reaching up and down, reaching toes right and left* and *arm circles*. The *jumping* motion is composed by shorter motions, of which the first is referred to as *jump-up*, consisting of a jump with arms starting at relaxed position with hands down, and then describing an arc trajectory that ends at head level. The other short motion, referred to as *jump-down*, consists of an arc trajectory back to the relaxed arms position, where the next jump-up starts. The *twisting* motion is composed by shorter motions further referred to as *twist right* and *twist left*. The *raising knees* motion is composed by shorter motions further referred to as *left knee* and *right knee*, each consisting of the motion starting with the respective foot touching the floor, then following a path until the highest point the knee can reach to the opposite side, and then returning to the initial position with the foot touching the floor. At the same time the torso twists to the side of the knee that has been raised, with arms naturally following the twisting motion. The *reaching up and down* motion is composed by shorter motions further referred to as *reach up* and *reach down*. The *reach* motion up starts with arms to the sides and hands at about waist level. The arms are raised, stretched upwards and then brought back to the initial position. The *reach down* motion starts in the same position as *reach up*, then reaching with the arms so that the tip of the fingers touch or almost touch the toes or the floor at about the imaginary line connecting the toes. The*reaching toes right and left* motion is composed by shorter symmetric motions further referred to as *reach left* and *reach right*. The *reach right* motion is composed by the movement towards the side almost reaching the respective toe. The starting point is the end of the previous motion. The *arm circles* motion is composed by shorter motions further referred to as *circle*, each of which is performed with arms extended to the side describing a circumference in the vertical plane, with center just below shoulder level. Two classes of this motion are performed: *small circle* and *big circle*, where the difference is the diameter.

Recording 3, with duration of 12.8 seconds is trial 15 for subjects 18 and 19, from the CMU database. This recording is of both subjects performing the Chicken Dance at the same time, twice. It has been previously used to test the segmentation algorithms by Matsubara, Sakurai and Faloustos [45], and Nagano et al. [46], although the latter study used only the markers on feet and hands of one subject. The study reported here has used the data for the same subject (subject 18, also referred to as "subject A" in the database's documentation). The Chicken Dance is composed by a sequence of four motions at coarse granularity: *beaks, wings, tail* and *clapping*. This sequence is performed twice. The *beaks* motion is composed by motions further referred to as *peck*. Each *peck* consists in closing and opening the thumbs against the other fingers, mimicking the pecking motion of a bird, with the hands just below shoulder level and arms bent. The *wings* motion is composed by motions further referred to as *flap*. Each *flap* is a movement that mimics a bird flapping its wings, with arms bent so that the elbows are pointing to the sides with the hands near and almost touching the chest. The *tail* motion can be decomposed into two motions, first a downward squatting and then the inverse motion upwards. These will be referred to as *tail down* and *tail up*, respectively. Each of these is composed by shorter motions further referred to as *twist*, each of which consists in horizontally moving the buttocks and arms while moving vertically. The *clapping* gesture is composed by short motions further referred to as *clap*. Each clap is done standing still, by quickly joining the hands in front of the body at a height of about the middle of the torso.

Recording 4 has a total duration of 134.4 seconds, consisting of a woman and a man dancing as a couple in *Lindy Hop* style to the first 126.1 seconds of the song "I Like Pie, I like Cake" [47]. Only the woman's data is used, trimmed to match the start and end of the music. During approximately the first 27 seconds of the song, the subject performs the dance individually. This means that she remained doing the dance patterns without horizontal displacement of the whole body. After that point the motions are performed with the man, involving several turns alternated with in-place motion. Qualitative inspection reveals that there might be several plausible segmentations for the complex motions in this recording, which is an observation congruent with the ones made by previous research [9], [10].

Recording 5 has a total duration of 119.5 seconds. It consists of a woman moving *quasi spontaneously* [48] to the first 108.5 seconds of the *disco* song "Stayin' Alive" [49]. Data

has been trimmed to match the end and start of the music. Like in recording 4, the motion patterns in this recording are complex and several plausible segmentations may be observed.

4.2 Evaluation of the Online Algorithm

Extensive experimentation was done manually manipulating the free variables of the algorithm. No ground truth was used to evaluate the effectiveness of the method, for two reasons. The first is that recorded motions at a coarse level of granularity have deemed to be evident and at finer levels any configuration of segmentation boundaries is deemed to be context-dependent and thus there may be several hierarchical layers, all of which may be considered true. The second reason is that there is no definite rule to evaluate segment transitions when they are overlapped. Therefore, the evaluation of results presented in the next section is qualitative and chiefly descriptive. The values for fixed variables used in the experiments are: resampled frame rate $= 30$ fps, derivative order $= 1$ (velocity), euclidean distance, $\alpha_{\text{nov}} = 2.5$ and $\alpha_{\text{filt}} = 2.5$.

The reflective markers that were selected from the raw motion capture recording also have an effect on the results. Any configuration of markers that represent the upper and lower limbs yields similar results. The results presented in this article were obtained by using markers on head, torso and articulations of upper and lower limbs. The exact configurations of markers are hard-coded in the program's script. The experiments were performed in an Apple Powerbook computing machine with an Intel Core 2 Duo processor capable of running at 2,66 GHz.

4.3 Results

Plausible sequences of segmentation boundaries were produced by the online segmentation algorithm, when evaluating the free variables shown in Table 1. Notably, a median ratio $n_{\text{nov}}/n_{\text{filt}} = 4$ is observed for recordings 1 to 4. This ratio also produced plausible segmentation sequences for recording 5, although sequences deemed to be better were obtained with different ratios. Table 1 also shows the number of markers used (n) and the ratio between the time length of recorded data T_{data} and the computation time of boundaries T_{comp} (i.e., how many times faster the machine outputs results compared to real-time data) rounded to one decimal place for each segmentation sequence. Figure 1 shows plots of all segmentation sequences produced and their novelty score. Due to zero padding of data, some sequences have boundaries at the beginning or end of the segmentation sequence. This effect may be reduced using the offline alternative segmentation algorithm described in the preceding section, although results using the offline algorithm are not shown in this article.

TABLE 1
Number of markers, free variables and ratios for resulting sequences of temporal segmentation boundaries. Each sequence number corresponds to a recording while letters indicate different granularities.

sequence	n	n_{nov}	n_{filt}	θ	$\frac{n_{\text{nov}}}{n_{\text{filt}}}$	$\frac{T_{\text{data}}}{T_{\text{comp}}}$
1a	27	144	35	6	4.114	35.6
1b	27	74	19	0	3.895	72.2
2a	16	450	111	0	4.054	5
2b	16	54	13	0	4.154	103
3a	41	108	27	0	4	36.5
3b	41	16	3	5	5.333	205
4a	17	540	135	0	4	3.1
4b	17	330	87	1.3	3.793	9.6
4c	17	210	53	1.9	3.962	25.5
5a	24	240	81	0	2.963	19
5b	24	240	59	0	4.068	17.4
5c	24	198	49	0	4.041	23.8
5d	24	198	33	0	6	23.9

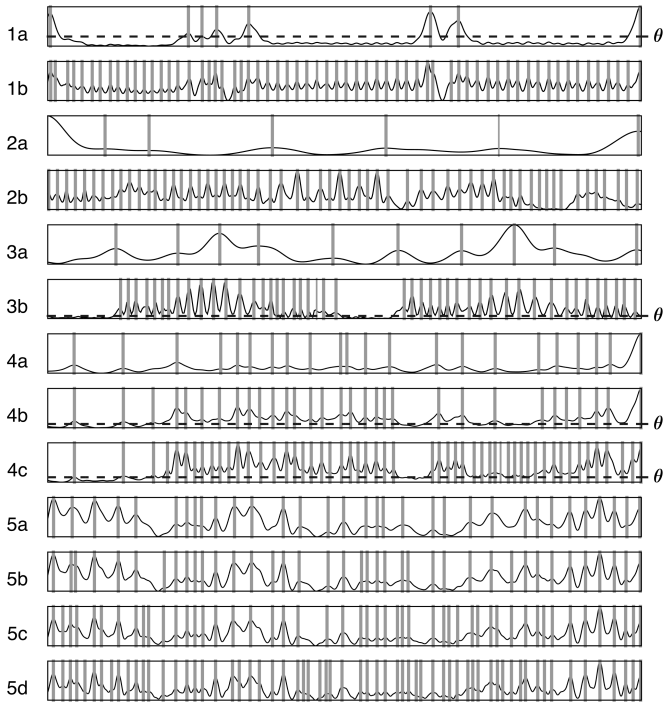

Fig. 1. Sequences corresponding to Table 1. Segmentation boundaries are shown as vertical grey lines and novelty scores are the black curves. Non-zero peak thresholds are shown by a dashed horizontal line. Vertical axes are normalized novelty and horizontal axes are normalized time.

4.3.1 Recording 1

Segmentation boundaries at coarse granularity discriminate between *weaving sticks* and *rest* (Fig. 1.1a). At about one third of the sequence there is a group of four boundaries, which indicate stop of motion, false start, sticks dropped down to *rest*ing position and raising sticks to chamber position. At about two thirds of the sequence there is another group of segments, of which the first indicates chambered sticks dropped to *rest* position and the second indicates chambering sticks. These boundaries are located at the most pronounced novelty peaks, while the threshold θ discards minor peaks. Figure 2 shows plots of traces for all markers. The first two panels to the left show the middle *weaving*

sticks segment and the subsequent *rest* segment. These are segments 6 and 7 of sequence '1a'. At fine granularity, each of the short segments within the *weaving sticks* gesture consists of alternating *hit-hit-hit left* and *hit-hit-hit right* (Fig. 1.1b). The four panels to the right in Figure 2 show traces for segments 33 to 36 of sequence '1b', viewed from upside down with the subject facing to the upper part of the image. The plots show movement having consistent asymmetry left-right.

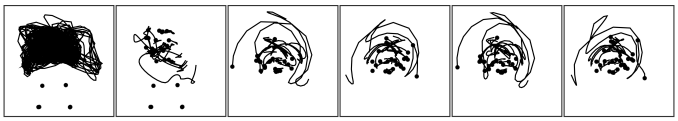

Fig. 2. Markers' trajectory traces of example results for recording 1. The first two plots to the left show a frontal view of coarse-granularity motions *weaving sticks* and *rest* (sequence '1a'). The remaining plots show a zenithal view with halfway azimuth rotation of fine-granularity segments *hit-hit-hit* (sequence '1b').

4.3.2 Recording 2

Segmentation boundaries discriminating between *jumping, twisting, raising knees, reaching up and down, reaching toes right and left* and *arm circles*, were produced at coarse granularity (Fig. 1.2a). Figure 3 shows a frontal view of traces for all markers. Segmentation boundaries at fine granularity discriminate between the motions within each of the coarse motions (Fig. 1.2b). Within *jumping*, the shorter motions *jump-up* and *jump-down* constitute separate segments. Within *twisting*, four shorter motions were discriminated, being sub-divisions of *twist right* and *twist left*. The first motion within *twist right* is a twist from the left side of the subject ending in the center, while the second motion is the continuation of the twist towards the right side. The same kind of subdivision is observed for *twist left*. Likewise, the motions *left knee* and *right knee* were subdivided, resulting in separate segments for the left knee going up, left knee going down, right knee going up and right knee going down. In the same fashion, the motions *reach up* and *reach down* were subdivided in segments reaching up, reaching down until the center of the body, reaching from the center to the feet, from the feet to the center and repeating the sequence. However the first *reaching down* motion was not sub-divided. At the end of this sequence there is a segment of rest. Then, the sequence shows subdivisions of *reach left* and *reach right*, resulting in segments for reaching to the left, return to the center, reaching to the right, returning to the center and repeating the sequence. Next, there are two segments of short motions in which the subject seems to accommodate for the following movements. The circles, no matter being *small circle* or *big circle*, are segmented mostly at half of the circumference, except for one segment before the last, which comprises an almost complete circle. Also the transition from small circle to big circle constitutes a separate segment.

4.3.3 Recording 3

Segmentation boundaries at coarse granularity discriminate between *beaks, wings, tail down, tail up* and *clapping* (Fig. 1.3a). The segmentation boundary in the middle of the sequence indicates the starting point of the sequence's repetition. Also, the novelty peaks are distinctively higher at the boundaries of the third motion (*tail*). This motion is the only one in the sequence that involves a vertical movement of the torso. The squatting motion, along with shaking the buttocks horizontally, strongly contributes to distinctiveness with previous and following gestures. The other motions are performed standing, only with hands or arms. The sequence produced at fine granularity (Fig. 1.3b) shows segments that compose motions at coarse granularity, except for the *beaks* motions because there were no markers placed on the thumbs. The weak novelty peaks produced within the *beaks* segment were discarded with the threshold θ, as they do not correspond to *pecks*, but to movement of other parts of the body, mainly elbows and forearms. Each *flap* motion within the *wings* motion is subdivided into arms going up and arms going down. The *tail down* and *tail up motions* are subdivided in twist segments. Each clap within the *clapping gesture* is also subdivided, into the closing of the hands and the opening of the hands. When this sequence is ended, a further segment appears, consisting of rising up the hands to the beaks position starting the second iteration of the Chicken Dance.

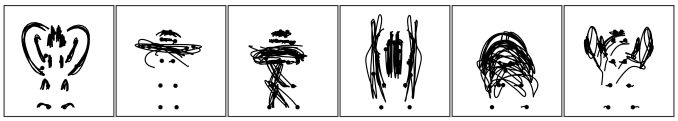

Fig. 3. Visualization of example results for recording 2. Markers and traces for coarse granularity segments (sequence '2a'). Frontal view.

Fig. 4. Visualization of example results for recording 3. Markers and traces for the first five coarse granularity segments (sequence '3a'), all of which comprise the full Chicken Dance. Frontal view.

4.3.4 Recording 4

Parameters for coarse granularity produced a sequence comprising segments of different lengths (Fig. 1.4a). Most of the segments are long and consist of several short motions. A sequence at medium granularity (Fig. 1.4b) discriminates several shorter segments while still retaining long segments. A sequence at fine granularity (Fig. 1.4c) has segments that indicate individual short motions. Segments at finer granularity were not produced. These segments would be composed by the individual footsteps, which were not possible to separate in the second segment because they are overlapped, resulting in merged segments. Hence, boundaries produced by individual or overlapped short and low-energy motions were discarded in sequences '4b' and '4c' by the peak threshold θ. This resulted in sequences with segments at different granularity levels, meaning large and short segments. For sequence '4a' there was no need for a peak threshold as short low-energy motions were filtered out.

Fig. 5. Visualization of example results for recording 4. Markers and traces for the first six segments of sequence '4c', of which the first three are coarse motions and the following are fine-grain motions. Frontal view.

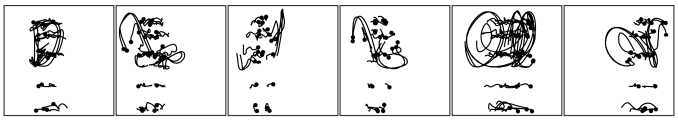

Fig. 6. Visualization of example results for recording 5. Markers and traces for segments 45 to 50, of fine-granularity sequence '5d'. Frontal view.

4.3.5 Recording 5

Sequences at coarse and fine granularity (Fig. 1.5a and Fig. 1.5d) were deemed to be the most satisfactory. Sequence '5a' consists of a combination of short and long segments, the latter merging short movements not necessarily being repetitions of the same motion. Sequence '5d' discriminates almost every possible short motion, except when there is substantial overlap. The medium-granularity sequences produced with $n_{\text{nov}}/n_{\text{filt}} = 4$ were also deemed to be plausible (Fig. 1.5b and Fig. 1.5c).

5 Discussion and Future Work

This section discusses the results of the experiments conducted to test the effectiveness of the proposed method for computing temporal segmentation boundaries in motion-captured data. The qualitative analyses presented in the previous section show that the method is able to compute plausible boundaries for semantically meaningful motions. When comparing results for the same data recording, at different levels of granularity, it is possible to observe that boundaries at coarser granularity match those at finer granularity. This is coherent with the observations on the hierarchical nature of perceived segmentation made by the studies in the sections at the beginning of this article. The rest of this section elaborates on the results, recognizing advantages and disadvantages of the method, and proposing directions for future work.

The use of velocity computed by differentiation as kinematic feature, is one factor that makes the online algorithm fast. The algorithm has proved to be fast to be implemented in real-time in a modest personal computer with the slowest tested computation time being about three times faster than sampling rate. This suggests that more than one instance of the algorithm could run in parallel at different granularity levels. To that extent, data buffers and the distance matrix may be used to compute several novelty scores without the need of redundant computation. Also the buffers and the distance matrix may be used for recognition and classification, or for cluster analysis. A practical real-time application may be found in systems that assist in giving quick feedback of performed actions, as seen in the results for recordings 1 and 2. This is especially true because due to the speed of the movement and the viewpoint constraint, a quick and accurate human assessment might not be possible to do at a glance.

Velocity is viewpoint invariant, which might be a disadvantage when the goal is to emulate human perception. This is because human visual observation can be affected by the viewpoint, as it is mostly a two-dimensional projection. However, the viewpoint-invariance may be regarded as an advantage as it will give results as if the motion-captured subject was observed from all possible point of views. Additionally, velocity may have a considerable effect when there is full-body displacement. While this may not affect the detection of segmentation boundaries, it may affect a subsequent classification or clustering.

Another characteristic that may be seen as disadvantage is that the three free variables of the algorithm need to be fine-tuned for optimal effectiveness. However, this also allows adaptation to produce expected results, which as it has been mentioned, depend on the context of application. Furthermore, as the experimental results show, the threshold parameter θ may be omitted in some cases and the kernels' widths n_{nov} and n_{nfilt} may be set to a fixed ratio. Nonetheless, converging to the optimal parameters may be attained with one of already explored methods (e.g., [5], [29], [31], [32]).

The algorithm only detects change-points that are regarded as segmentation boundaries, without evaluating if the resulting segments are transitions or merged segments. Also in the case of segments having merged sections, the method will indicate a boundary somewhere in the middle of the merged section. The pertinence of discrimination between strict segmentation boundaries, meaning without considering if there is a transition or an overlap, will depend on the application. If the aim is to emulate human perception, there is a vast area of future research to be done to elucidate the perceptual workings of the human mind regarding transitions and overlaps in temporal segmentation.

Delay in the output of results may also be a disadvantage. The time of output for each segmentation boundary, without counting computation time, will be of the length of both kernels (novelty and filter) added together, plus one sample per derivative order, plus one sample for testing novelty local maximum. For a real-time application, the appropriateness of this delay time will depend on the required reaction time. A special case is that when an immediate reaction to a change in an observed subject's motion pattern is required. For example, when a system has to produce an output at the exact time that the observed subject's motion changes form one pattern to another. This may only be accomplished by predicting the change-point from the observation of the preceding motions and by incorporating knowledge of human motion into the design of the method [50], [51].

Computation of segmentation boundaries with the described method is deterministic, as a given collection of data and values for parameters will always produce the same output. Therefore it can be used as a measurement standard opposed to human annotation, which will have some degree of subjectivity observed as inter-subject and intra-subject variability. Nonetheless, investigation of methods for developing more sophisticated methods to measure

human perceived segmentation remains a wide avenue for further research. As observed in the results for recording 4, there might be more than one plausible ground-truth. This is consistent with the observations on annotators' agreement made by studies mentioned at the beginning section of this article. The methods used for assessing segmentation algorithms have typically relied on the "majority voting" paradigm, disregarding the segmentations proposed by a reduced number of annotators. However, those rejected annotations might still be plausible.

Future work should also consider testing algorithms that have been published in research articles, especially considering that many of them have also published the software code implemented in fairly accessible programming environments such as Matlab, C++, Python or Java [17]. This testing should be done using a variety of motion capture data, featuring a range of motions from simple actions to complex motions like spontaneous dance. Likewise, multi-granularity and the plausibility of different ground truths for a single level of granularity should be added to the analyses.

6 Conclusion

This article describes a method for automatically finding segmentation boundaries of human motion. The method is based in an online algorithm that correlates a checkerboard kernel with a distance matrix computed upon a moving window of time-ordered kinematic features extracted from motion-captured data. The result of the correlation is a novelty score that has local maxima when there is a change in the kinematic features. This change is deemed to be a segmentation boundary. The granularity of the observed changes may be adjusted by the size of the checkerboard kernel and the width of a Gaussian kernel used to filter noise from the novelty score. Additionally, irrelevant local maxima may be discarded by setting a threshold. The asymptotical computational complexity is quadratic, and computation time depends on the sampling rate, number of kinematic features and size of the kernels.

The algorithm has been evaluated using five recordings of optical motion capture data based in markers attached to one person. Hence, the raw data consisted of a multi-dimensional array of markers' spatial coordinates in time. The kinematic features used to compute the novelty score were the velocity of each marker. The recorded motions were performed by different people and corresponded to a variety of motions ranging from simple motions like repetitive exercises to complex motion like choreographed dance and quasi-spontaneous movement to music. A descriptive qualitative analysis show that by fine-tuning its three free variables, the algorithm can produce segmentation boundaries that are perceptually plausible at different granularity levels. Furthermore, it was observed that for data containing fairly simple motions, it is possible to establish a fixed ratio between the sizes of the kernels. Also most of the produced segmentation results did not need a threshold for novelty peaks. This suggests that for many applications the number of user-defined parameters may be reduced to only one. The discussion on the results highlights the suitability of the algorithm for real-time applications and to be incorporated into systems that perform further classification or clustering.

Acknowledgments

This study has been partially funded by the Finnish Foundation for Technology Promotion (Tekniikan edistämissäätiö). Deniz Duman helped with logistics and pre-processing of recording 4.

References

[1] S. Xia, L. Gao, Y.-K. Lai, M.-Z. Yuan, and J. Chai, "A survey on human performance capture and animation," *Journal of Computer Science and Technology*, vol. 32, no. 3, pp. 536–554, May 2017. [Online]. Available: https://doi.org/10.1007/s11390-017-1742-y

[2] E. van der Kruk and M. M. Reijne, "Accuracy of human motion capture systems for sport applications; state-of-the-art review," *European Journal of Sport Science*, vol. 18, no. 6, pp. 806–819, 2018, pMID: 29741985. [Online]. Available: https://doi.org/10.1080/17461391.2018.1463397

[3] P. Paliyawan, W. Choensawat, and R. Thawonmas, "Mossar: motion segmentation by using splitting and remerging strategies," *Multimedia Tools and Applications*, vol. 77, no. 21, pp. 27761–27788, Nov 2018. [Online]. Available: https://doi.org/10.1007/s11042-018-5965-x

[4] J. Bernard, E. Dobermann, A. Vögele, B. Krüger, J. Kohlhammer, and D. Fellner, "Visual-interactive semi-supervised labeling of human motion capture data," in *Visualization and Data Analysis (VDA)*, ser. Electronic Imaging. USA / Vereinigte Staaten: Society for Imaging Science and Technology, 2017, pp. 34–45.

[5] W. Takano and Y. Nakamura, "Real-time unsupervised segmentation of human whole-body motion and its application to humanoid robot acquisition of motion symbols," *Robotics and Autonomous Systems*, vol. 75, pp. 260 – 272, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S092188901500216X

[6] D. Newtson, "Attribution and the unit of perception of ongoing behavior." *Journal of Personality and Social Psychology*, vol. 28, no. 1, p. 28, 1973.

[7] J. M. Zacks, B. Tversky, and G. Iyer, "Perceiving, remembering, and communicating structure in events." *Journal of Experimental Psychology: General*, vol. 130, no. 1, p. 29, 2001.

[8] B. M. Hard, B. Tversky, and D. S. Lang, "Making sense of abstract events: Building event schemas," *Memory & cognition*, vol. 34, no. 6, pp. 1221–1235, 2006.

[9] K. Kahol, P. Tripathi, and S. Panchanathan, "Automated gesture segmentation from dance sequences," in *Sixth IEEE International Conference on Automatic Face and Gesture Recognition, 2004. Proceedings.* IEEE, 2004, pp. 883–888.

[10] B. E. Bläsing, "Segmentation of dance movement: effects of expertise, visual familiarity, motor experience and music," *Frontiers in psychology*, vol. 5, p. 1500, 2015.

[11] J. M. Zacks, S. Kumar, R. A. Abrams, and R. Mehta, "Using movement and intentions to understand human activity," *Cognition*, vol. 112, no. 2, pp. 201–216, 2009.

[12] S. Mitra and T. Acharya, "Gesture recognition: A survey," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 37, no. 3, pp. 311–324, 2007.

[13] M. A. R. Ahad, J. Tan, H. Kim, and S. Ishikawa, "Human activity recognition: Various paradigms," in *2008 International Conference on Control, Automation and Systems*. IEEE, 2008, pp. 1896–1901.

[14] O. D. Lara and M. A. Labrador, "A survey on human activity recognition using wearable sensors," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 3, pp. 1192–1209, 2013.

[15] J. F.-S. Lin, M. Karg, and D. Kulić, "Movement primitive segmentation for human motion modeling: A framework for analysis," *IEEE Transactions on Human-Machine Systems*, vol. 46, no. 3, pp. 325–339, 2016.

[16] V. Krüger, D. Kragic, A. Ude, and C. Geib, "The meaning of action: a review on action recognition and mapping," *Advanced Robotics*, vol. 21, no. 13, pp. 1473–1501, 2007. [Online]. Available: https://www.tandfonline.com/doi/abs/10.1163/156855307782148578

[17] C. R. G. Dreher, N. Kulp, C. Mandery, M. Wächter, and T. Asfour, "A framework for evaluating motion segmentation algorithms," in *2017 IEEE-RAS 17th International Conference on Humanoid Robotics (Humanoids)*, 2017, pp. 83–90.

[18] B. Krüger, A. Vögele, T. Willig, A. Yao, R. Klein, and A. Weber, "Efficient unsupervised temporal segmentation of motion data," *IEEE Transactions on Multimedia*, vol. 19, no. 4, pp. 797–812, 2017.

[19] F. Meier, E. Theodorou, F. Stulp, and S. Schaal, "Movement segmentation using a primitive library," in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2011, pp. 3407–3412.

[20] F. Patrona, A. Chatzitofis, D. Zarpalas, and P. Daras, "Motion analysis: Action detection, recognition and evaluation based on motion capture data," *Pattern Recognition*, vol. 76, pp. 612 – 622, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0031320317304910

[21] R. Lan and H. Sun, "Automated human motion segmentation via motion regularities," *Vis. Comput.*, vol. 31, no. 1, pp. 35–53, Jan. 2015. [Online]. Available: http://dx.doi.org/10.1007/s00371-013-0902-5

[22] S. Salamah, L. Zhang, and G. Brunnett, *Hierarchical Method for Segmentation by Classification of Motion Capture Data*. Cham: Springer International Publishing, 2015, pp. 169–186. [Online]. Available: https://doi.org/10.1007/978-3-319-17043-5_10

[23] L. Santos, K. Khoshhal, and J. Dias, "Trajectory-based human action segmentation," *Pattern Recognition*, vol. 48, no. 2, pp. 568 – 579, 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S003132031400329X

[24] F. Lv and R. Nevatia, "Recognition and segmentation of 3-d human action using hmm and multi-class adaboost," in *Computer Vision – ECCV 2006*, A. Leonardis, H. Bischof, and A. Pinz, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 359–372.

[25] M. Müller, T. Röder, and M. Clausen, "Efficient content-based retrieval of motion capture data," *ACM Trans. Graph.*, vol. 24, no. 3, pp. 677–685, Jul. 2005. [Online]. Available: http://doi.acm.org/10.1145/1073204.1073247

[26] F. Zhou, F. D. l. Torre, and J. K. Hodgins, "Hierarchical aligned cluster analysis for temporal clustering of human motion," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 3, pp. 582–596, 2013.

[27] J. Barbič, A. Safonova, J.-Y. Pan, C. Faloutsos, J. K. Hodgins, and N. S. Pollard, "Segmenting motion capture data into distinct behaviors," in *Proceedings of Graphics Interface 2004*, ser. GI '04. School of Computer Science, University of Waterloo, Waterloo, Ontario, Canada: Canadian Human-Computer Communications Society, 2004, pp. 185–194. [Online]. Available: http://dl.acm.org/citation.cfm?id=1006058.1006081

[28] J. Kohlmorgen and S. Lemm, "A dynamic hmm for on-line segmentation of sequential data," in *Advances in Neural Information Processing Systems*, vol. 14. MIT Press, 2002, pp. 793–800.

[29] D. Kulic, W. Takano, and Y. Nakamura, "Online segmentation and clustering from continuous observation of whole body motions," *IEEE Transactions on Robotics*, vol. 25, no. 5, pp. 1158–1166, 2009.

[30] T. Taniguchi, K. Hamahata, and N. Iwahashi, "Unsupervised segmentation of human motion data using a sticky hierarchical dirichlet process-hidden markov model and minimal description length-based chunking method for imitation learning," *Advanced Robotics*, vol. 25, no. 17, pp. 2143–2172, 2011. [Online]. Available: https://doi.org/10.1163/016918611X594775

[31] A. Bargi, R. Y. D. Xu, and M. Piccardi, "Adon hdp-hmm: An adaptive online model for segmentation and classification of sequential data," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 9, pp. 3953–3968, 2018.

[32] D. Gong, G. Medioni, and X. Zhao, "Structured time series analysis for human action segmentation and recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 7, pp. 1414–1427, 2014.

[33] G. Xia, H. Sun, L. Feng, G. Zhang, and Y. Liu, "Human motion segmentation via robust kernel sparse subspace clustering," *IEEE Transactions on Image Processing*, vol. 27, no. 1, pp. 135–150, 2018.

[34] S. Gharghabi, C.-C. M. Yeh, Y. Ding, W. Ding, P. Hibbing, S. LaMunion, A. Kaplan, S. E. Crouter, and E. Keogh, "Domain agnostic online semantic segmentation for multi-dimensional time series," *Data Mining and Knowledge Discovery*, vol. 33, no. 1, pp. 96–130, Jan 2019. [Online]. Available: https://doi.org/10.1007/s10618-018-0589-3

[35] J. I. Mendoza, "Self-report measurement of segmentation, mimesis and perceived emotions in acousmatic electroacoustic music," Master's thesis, University of Jyväskylä, 2014.

[36] J. I. Mendoza and M. R. Thompson, "Modelling perceived segmentation of bodily gestures induced by music," in *ESCOM 2017 : Conference proceedings of the 25th Anniversary Edition of the European Society for the Cognitive Sciences of Music (ESCOM). Expressive Interaction with Music*, E. Van Dyck, Ed. Ghent University, 2017, pp. 128–133.

[37] D. Endres, A. Christensen, L. Omlor, and M. A. Giese, "Emulating human observers with bayesian binning: Segmentation of action streams," *ACM Trans. Appl. Percept.*, vol. 8, no. 3, pp. 16:1–16:12, Aug. 2011. [Online]. Available: http://doi.acm.org/10.1145/2010325.2010326

[38] M. Markou and S. Singh, "Novelty detection: a review—part 1: statistical approaches," *Signal Processing*, vol. 83, no. 12, pp. 2481–2497, 2003. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0165168403002020

[39] J. Foote, "Automatic audio segmentation using a measure of audio novelty," in *2000 IEEE International Conference on Multimedia and Expo. ICME2000. Proceedings. Latest Advances in the Fast Changing World of Multimedia (Cat. No.00TH8532)*, vol. 1, 2000, pp. 452–455 vol.1.

[40] J. Foote and M. L. Cooper, "Media segmentation using self-similarity decomposition," in *Proc. SPIE, Storage and Retrieval for Media Databases*, vol. 5021. International Society for Optics and Photonics, January 2003, pp. 167–175.

[41] I. N. Junejo, E. Dexter, I. Laptev, and P. Perez, "View-independent action recognition from temporal self-similarities," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 1, pp. 172–185, 2011.

[42] B. Burger and P. Toiviainen, "MoCap Toolbox – A Matlab toolbox for computational analysis of movement data," in *Proceedings of the 10th Sound and Music Computing Conference*, R. Bresin, Ed. Stockholm, Sweden: KTH Royal Institute of Technology, 2013, pp. 172–178.

[43] (2018, October) Carnegie Mellon University Graphics Lab motion capture database. [Online]. Available: http://mocap.cs.cmu.edu/

[44] T. Nakamura, T. Nagai, D. Mochihashi, I. Kobayashi, H. Asoh, and M. Kaneko, "Segmenting continuous motions with hidden semi-markov models and gaussian processes," *Frontiers in neurorobotics*, vol. 11, no. 67, 2017.

[45] Y. Matsubara, Y. Sakurai, and C. Faloutsos, "Autoplait: Automatic mining of co-evolving time sequences," in *Proceedings of the 2014 ACM SIGMOD international conference on Management of data*. ACM, 2014, pp. 193–204.

[46] M. Nagano, T. Nakamura, T. Nagai, D. Mochihashi, I. Kobayashi, and M. Kaneko, "Sequence pattern extraction by segmenting time series data using gp-hsmm with hierarchical dirichlet process," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 4067–4074.

[47] G. Little, A. Sizemore, and L. Shay, "I like pie, i like cake," Recorded by Steven Mitchell and Gordon Webster band. On Live in Rochester (CD). Harro East Ballroom, Rochester, N.Y.: Gordon Webster Swings (18-19 November, 2012), n.d.

[48] B. Burger, S. Saarikallio, G. Luck, M. R. Thompson, and P. Toiviainen, "Relationships between perceived emotions in music and music-induced movement," *Music Perception: An Interdisciplinary Journal*, vol. 30, no. 5, pp. 517–533, 2013.

[49] B. Gibb, R. Gibb, and M. Gibb, "Stayin' alive," On Saturday Night Fever, The Original Motion Picture Soundtrack. Germany: RSO, 1977.

[50] J. Bütepage, H. Kjellström, and D. Kragic, "Anticipating many futures: Online human motion prediction and generation for human-robot interaction," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 1–9.

[51] B. Reily, F. Han, L. E. Parker, and H. Zhang, "Skeleton-based bio-inspired human activity prediction for real-time human–robot interaction," *Autonomous Robots*, vol. 42, no. 6, pp. 1281–1298, 2018.

Juan Ignacio Mendoza received a professional degree in Composition and Arrangements of Popular Music from the "Escuela Moderna de Musica" institute in Chile. Later he received a M.A. in Music, Mind and Technology from the University of Jyväskylä, where he is currently a doctoral student in Musicology. His research interests include embodied music cognition, machine learning and human-machine musical interaction.