IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 10 November 2022, accepted 24 November 2022, date of publication 1 December 2022,
date of current version 6 December 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3225914

RESEARCH ARTICLE

Unsupervised Segmentation of Human Manipulation Movements Into Building Blocks

LISA GUTZEIT[1] **AND FRANK KIRCHNER**[1,2]
[1]Robotics Research Group, University of Bremen, 28359 Bremen, Germany
[2]Robotics Innovation Center, DFKI, 28359 Bremen, Germany

Corresponding author: Lisa Gutzeit (Lisa.Gutzeit@uni-bremen.de)

This work was supported by the German Federal Ministry for Economic Affairs and Energy (BMWi) under Grant FKZ 50 RA 1703 and Grant 50 RA 2023.

This work involved human subjects or animals in its research. Approval of all ethical and experimental procedures and protocols was granted by the Ethics Committee of the University of Bremen, and performed in line with the Declaration of Helsinki.

ABSTRACT During the last years, new approaches were proposed in which robotic behavior is generated by imitating human movement examples. This process can be sustainably simplified by an automatic detection of the movement sequences which should be imitated. For this, automated approaches for human movement segmentation are needed to avoid time-intensive manual data analysis. Suitable examples for imitation learning are building block movements, which are basic movements that can be combined to solve different tasks. Recently, we introduced the velocity-based Multiple Change-point Inference (vMCI) algorithm, which automatically segments human demonstrations of manipulation movements into sequences with a bell-shaped velocity of the hand which is said to be a characteristic feature of manipulation building blocks. In this paper, the velocity of the hand as well as other features of human manipulation movements recorded with a marker-based motion tracking system are evaluated with respect to their suitability to detect segment boundaries of manipulation building blocks. Additionally, we perform a more intensive evaluation of vMCI compared to the original publication by evaluating the algorithm on different manipulation movement demonstrations recorded from several subjects and comparing the approach to other state-of-the-art segmentation algorithms. The results support the assumption that the velocity of the hand is one of the main features to detect segment boundaries in human manipulation movements and that the vMCI algorithm can detect these segment borders online and unsupervised, also in movement recordings with a noisy velocity.

INDEX TERMS Human movement analysis, movement segmentation, unsupervised segmentation, time series segmentation.

I. INTRODUCTION

To realize a natural and intuitive interaction between humans and robotic systems, the human behavior needs to be analyzed and understood to make a precise reaction of the system possible. For this, methods are needed to identify the current movement as well as the intentions of the human. Such methods are also of high interest in learning from demonstration (LfD) applications in which human movement demonstrations are directly used to generate robotic behavior in an intuitive way that does not require expert knowledge [1]. If a mapping from human body positions to the robotic system is available, the demonstrations for LfD can directly be obtained from human movement data, recorded for example, by using marker-based motion tracking systems [2]. By segmenting the recorded data, the movements of interest can be identified. Human movement segmentation is a challenging problem because in naturally performed behavior, the variations in the execution of the same behavior can be big, even if it is performed by the same person. By using automated

The associate editor coordinating the review of this manuscript and approving it for publication was Sajid Ali.

segmentation approaches, which can handle inter-subject as well as intra-subject variations, this process can be sustainably simplified and accelerated.

In LfD, most state-of-the-art methods are monolithic approaches in which one movement is learned that covers the whole demonstration. This can be time consuming and for complex behavior it even can be impossible. In humans, complex behavior is learned incrementally by first learning simple building blocks which are combined to complex behavior [3]. Graybiel refers to this principle as *chunking of action repertoires* [4]. If this is transferred to behavior learning in robotic systems, i.e., movement building blocks are learned which can be combined to execute various behaviors, more complex robotic movements can be learned more effectively and the learned building blocks movements can be recombined to tasks which were not directly demonstrated by the human. For this, movement building blocks must be identified, which are central movement identities in different, but similar motions.

Although the generation of human arm movement it not fully understood, several studies indicate that human movement is generated based on a small number of building blocks, often referred to as motor primitives [5]. These primitives are described in the literature from the neuronal, kinematic and dynamic perspective [6]. On the behavioral level, which can be observed and recorded, e.g., by using motion tracking, regularities in the movement execution of the hand were investigated in several studies. In different experiments, a bell-shaped hand velocity was observed in point-to-point as well has curved hand motions [7], [8], indicating the relevance of this feature for human arm motions. For this reason, we define **building blocks** in this work as central elements in human manipulation movements consisting of goal-directed hand movements, which are characterized by a bell-shaped velocity profile of the hand.

An example of the characteristic velocity profiles in human arm movements can be seen in in Fig. 1, which shows a motion capture setup in which a stick is moved through a step pattern. In the first movement recording, the stick is moved fast through the pattern. At the corners, the participant naturally slowed down the movement, which automatically separates the demonstration into single point-to-point movements. Each of these movements shows a bell-shaped curve in the absolute velocity of the hand. In the second demonstration, where the participant was introduced to hit the boundaries as seldom as possible, the absolute velocity of the hand is noisier, but still bell-shaped curves can be observed. With this information, we aim to segment human manipulation movements into movement building blocks with a bell-shaped velocity profile.

To detect these building blocks automatically, we present the probabilistic and unsupervised segmentation method *velocity-based Multiple Change-point Inference* (vMCI). To our knowledge, vMCI is the first unsupervised behavior segmentation algorithm, which includes the characteristic velocity pattern into the segmentation process to detect building blocks in manipulation movements. We introduced this

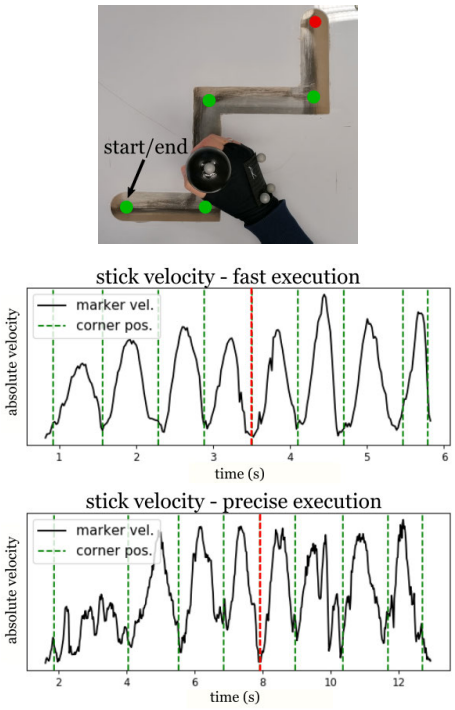

FIGURE 1. Example of bell-shaped velocity pattern in human point-to-point movements. The subjects were instructed to move a stick though a step pattern, starting at the lower right, going to the upper right and backwards. The absolute velocity of the hand is visualized on the right side. The positions of the corners are marked as green dots in the left image and green dashed lines on the right side respectively. The turning point of movement is indicated in red. The movement was performed fast (right top) and precisely, i.e., with the intention to not hit the boundaries. In both cases, bell-shaped curves can be observed in the velocity of the hand for single point-to-point movements.

approach already in a previous publication [9], in which the vMCI algorithm was compared to the Multiple Change-point Inference (MCI) algorithm [10], on the basis of which vMCI was developed, as well as to a segmentation method called beta-process autoregressive Hidden Markov Models (BPARHMM) [11] and a method based on local minima (locMin). The results showed that vMCI is robust against noise, has a reduced influence of the selected hyperparameters, and can handle variations in movement execution more efficiently than MCI, BPARHMM, and locMin. However, in [9] vMCI was only evaluated on artificial data and a small real dataset consisting of 3 movement examples of one subject of the point-to-point movements of the step setup depicted in Fig. 1.

In this work, we present several data sets recorded using a marker-based motion tracking system which captures marker position with high precision and resolution where also small variation in the movement can be detected. The datasets contain simple point-to-point movements as well as more complex demonstrations of throwing and pick-and-place movements. For each movement, demonstrations of at least 3 subjects were recorded in order to cover inter-subject as well as intra-subject variations. We first compare different movement features including the absolute velocity

of the hand with respect to their suitability to characterize movement building blocks in point-to-point and pick-and-place movements. Afterwards, we evaluate the vMCI algorithm on the three datasets and compare it to MCI, BPARHMM, locMin, and a more recent probabilistic segmentation approach called ProbS [12]. These evaluations are an extension compared to the experiments performed in our previous publication to bigger datasets. With them the performance of vMCI on different manipulation datasets can be determined in comparison to other state-of-the-art segmentation approaches.

This paper is structured as follows: First, a short overview about related work is given, which is followed by the presentation of the vMCI algorithm in section III. The process of data generation, including data acquisition and the generation of labeled ground truth is described in section IV. Afterwards, the experiments regarding the feature analysis and the evaluation of vMCI in comparison to other approaches are presented (section V and VI). At the end, we discuss our work and give a conclusion.

II. RELATED WORK

For a long time, most approaches to segment time series data of human movements were supervised and required manual segmentation of movement examples to generate training data. An overview is given in [13]. In the last years, several new approaches were proposed which can be run unsupervised [14], i.e., manual efforts can be reduced and the methods are also applicable if the behavior segments are not know in advance. Fod et al. presented a heuristic approach, in which segment boundaries are detected using the angular velocities of several degrees of freedom [15]. If these cross zero, the start of a new movement segment is assumed. Similarly, segments are identified based on 'swings' in the velocity of joint angle data in [16]. In both approaches, thresholds need to be defined in advance, which probably must be adapted for different datasets.

Probabilistic approaches generalize to different movement executions by integrating movement variations directly into the model. With a predefined number of segments to be detected, table tennis demonstrations were segmented using a belief network in [17]. Kulić et al. automatically segmented and clustered human motions based on Hidden Markov Models (HMMs) [18] and a method based on Hilbert space embedding of distributions is presented in [19]. In both approaches, full-body human motions are segmented online.

A probabilistic model, for which also the implementation is available online, is presented in [11]. In this method, repeated building blocks of the same movement are inferred using the so-called *beta process autoregressive HMM* (BPARHMM). Like in our proposed method, the segments are represented by linear regression models (LRMs). The Multiple Change-point Inference (MCI) algorithm presented by Fearnhead and Liu uses Bayesian Inference to detect time points in which the underlying LRM changes [10]. It was used in [20] to segment trajectories recorded by maneuvering a robot through a corridor and is the basis for the segmentation approach presented in this paper. In [12], a probabilistic segmentation approach called ProbS is presented, in which segmentation points are inferred using expectation maximization based on initially over-segmented data. The detected segments are represented using dynamical movement primitives (DMPs; [21]), which is a popular representation of movement demonstrations in LfD. Using this approach, it was possible to automatically learn primitive movements that are needed to assemble a chair with a robotic system from human demonstrations. In the experimental evaluation, ProbS performed better than BPARHMM on the examined datasets [12]. Other approaches in the literature are used to detect periodic segments in videos, such as the P-MUCOS algorithm presented in [22].

Most of the methods presented in the literature are evaluated on small datasets of single subjects [14]. The approach presented in this paper is evaluated on multiple datasets, each containing movement data of different subjects in order that the ability of the algorithm to generalize can be shown.

III. vMCI-ALGORITHM

The velocity-based Multiple Change-point Inference (vMCI) is an algorithm which automatically detects the borders of building blocks with a bell-shaped velocity in human movement data. It is based on the MCI algorithm introduced in [10] and was first published by us in [9]. An open source implementation is available online.[1] In this section, we repeat the main concept of vMCI with slightly adapted mathematical notations.

A. DATA REPRESENTATION

In the vMCI algorithm, it is assumed that a data sequence $y = (y_1, \ldots, y_T)$ of length T, with $y_i = (y_i^p, y_i^v) \in \mathbb{R}^{d+d^v}$ being an observation at time point i with position information of the observation y_i^p and its velocity y_i^v, consists of an unknown number of segments with a bell-shaped velocity. The segments are modeled independently of each other with linear regression models (LRMs). In this manner, they are represented as a weighted sum of basis functions with added noise, see the schematic overview in Fig. 2. The position information of a single segment $y_{i+1:j}$, starting at time point $i+1$ and ending at time point j, is represented as:

$$y_{i+1:j}^p = \sum_{k=1}^{q} \beta_k \phi_k + \varepsilon, \tag{1}$$

with q basis functions $\phi_k, k \in \{1, \ldots, q\}$, model parameters $\beta = (\beta_1, \ldots, \beta_q)$, and independent and identically distributed Gaussian noise ε with zero mean and variance Σ. This model is identical to the data model in the MCI algorithm [10]. To infer the model from the data, prior distributions are set over the weights and the noise variance. The parameters β are assumed to be matrix-normal distributed

[1] Open source implementation can be found here: https://github.com/dfki-ric/vMCI_segmentation

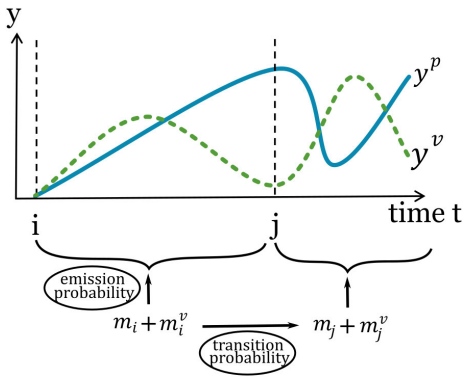

FIGURE 2. Schematic presentation of the Hidden Markov Model to detect segment borders with vMCI. The observable data sequence y, consisting of position y^p and velocity y^v, is generated by hidden models m and m_v separately for each segment. Segment borders are detected at positions where the underlying models change, either to new models or the same with different parameters.

with zero mean and covariances D and Σ along rows and columns respectively. To ensure direct calculation of the posterior probability of the data, conjugate priors are assumed. This results in an inverse Wishart prior for the variance Σ with hyper-parameters ν and S.

To account for the bell-shaped velocity for each segment, the velocity information y_i^v, which is the absolute velocity in the direction of the movement, is in contrast to the MCI algorithm modeled separately in vMCI with a basis function ϕ_v that represents the bell-shaped structure. The basis ϕ_v is defined as a single radial basis function with center c and width r:

$$\phi_v(x_t) = \exp\left\{-\frac{(c - x_t)^2}{r^2}\right\}. \tag{2}$$

The width parameter r is chosen to be half of the assumed segment length, i.e., $r = (j - i)/2$, so that the whole segment can be covered by the model. The center c regulates the alignment to different velocity curves. For example, a center location closer to the starting point of the segment allows to approximate a segment with high velocity at the beginning and rather low velocity at the end. In the vMCI algorithm, different center positions are fitted to the observed data. These centers are equally distributed over the segment length and their number is predefined. During inference, the best fitting model is determined.

With the basis ϕ_v, the velocity y_i^v is represented using:

$$y_{i+1:j}^v = \alpha_1 \phi_v + \alpha_2 + \varepsilon^v, \tag{3}$$

with weights $\alpha = (\alpha_1, \alpha_2)$ and noise ε^v. Again, weights and noise are matrix-normal distributed with $\alpha \sim \mathcal{MN}(0, D_v, \Sigma_v)$ and $\varepsilon \sim \mathcal{MN}(0, I, \Sigma_v)$. D_v and Σ_v are the prior parameters. The prior distribution of Σ_v is again chosen to be inverse Wishart, $\Sigma_v \sim \mathcal{IW}(\nu_v, S_v)$ to provide conjugate priors. The model order is fixed to 2, with the two basis functions ϕ_v and 1. The constant, weighted with α_2, is added to account for velocities unequal to zero at start or end of the segment.

To determine the emission probability (see Fig. 2), the likelihood of the data $y_{i+1:j}$ given model m and velocity model m_v needs to be determined. As independent models are assumed for the velocity and the position, the likelihood of the data sequence $p(y_{i+1:j}|m, m_v)$ given the model m of order q and velocity model m_v, can be derived by marginalizing out the model parameters β and α, i.e.,

$$\begin{aligned} p(y_{i+1:j}|m, m_v) &= \int p(y_{i+1:j}^p|\beta, m) p(\beta)\, d\beta \\ &\quad \cdot \int p(y_{i+1:j}^v|\alpha, m_v) p(\alpha)\, d\alpha \qquad (4) \\ &= \int\int p(y_{i+1:j}^p|\beta, \Sigma) \cdot p(\beta|D, \Sigma) \\ &\quad \cdot p(\Sigma|\nu, S)\, d\Sigma\, d\beta \\ &\quad \cdot \int\int p(y_{i+1:j}^v|\alpha, \Sigma_v) \cdot p(\alpha|D_v, \Sigma_v) \\ &\quad \cdot p(\Sigma_v|\nu_v, S_v)\, d\Sigma_v\, d\alpha. \qquad (5) \end{aligned}$$

Due to the chosen conjugate priors for both LRMs, the integrals can directly be solved resulting in

$$\begin{aligned} p(y_{i+1:j}|m, m^v) &= (2\pi)^{-\frac{nd}{2}} \frac{|M|^{\frac{d}{2}}}{|D|^{\frac{d}{2}}} \frac{|S|^{\frac{\nu}{2}}}{|(y^p)^T P y^p + S|^{\frac{n+\nu}{2}}} \\ &\quad \cdot \frac{\Gamma_d(\frac{n+\nu}{2})}{\Gamma_d(\frac{\nu}{2}) 2^{\nu d/2}} \cdot (2\pi)^{-\frac{nd_v}{2}} \\ &\quad \cdot \frac{|M_v|^{\frac{d_v}{2}}}{|D_v|^{\frac{d_v}{2}}} \frac{|S_v|^{\frac{\nu_v}{2}}}{|(y^v)^T P_v y^v + S_v|^{\frac{n+\nu_v}{2}}} \\ &\quad \cdot \frac{\Gamma_{d_v}(\frac{n+\nu_v}{2})}{\Gamma_{d_v}(\frac{\nu_v}{2}) 2^{\nu_v d_v/2}}, \end{aligned} \tag{6}$$

with $M = (H^T H + D^{-1})^{-1}$, $P = I - HMH^T$ used to determine the model evidence for the position data y_p and $M_v = (H_v^T H_v + D_v^{-1})^{-1}$, $P_v = I - H_v M_v H_v^T$ used to determine the model evidence for the velocity data y_v. In these notations H and H_v refer to the matrices of basis functions, i.e., $H = (\phi_1, \ldots, \phi_q)$ of shape $n \times q$ and $H_v = (\phi_v, 1)$ of shape $n \times 2$, with n defining the length of the segment, i.e., $n = j - i$. The matrix I is the $n \times n$ identity matrix and Γ_d is the d-dimensional Gamma function. The derivation of this formula for the general case can be found, e.g., in [23].

B. ONLINE INFERENCE OF CHANGE-POINTS

Using the model likelihood $p(y_{i+1:j}|m, m_v)$, the segmentation points in the data y as well as the underlying LRMs of the observed data can be determined using an online Viterbi algorithm presented in [10] which will be described in this section. Here, a segmentation point is named change-point and refers to a time point i in the series y where the underlying LRM changes. The segment models are assumed to be independent of each other and the change-point positions are modeled via a Markov process, as depicted in Fig. 2. The transition probabilities dependent on the segment length between two change-points and are defined as:

$$P(\text{next change-point at j}|\text{change-point at i}) = g(j - i), \tag{7}$$

where $g(l)$ is the probability of a segment having length l. The cumulative distribution function of this length is given by $G(l) = \sum_{k=1}^{l} g(k)$. As proposed in [10], we assume a geometric distribution for $g(l)$, so that $g(l) = (1-p)^{l-1}p$ and $G(l) = 1-(1-p^l)$. Using these distributions, the parameter p regulates the expected segment length, which is $1/p$.

For each time point t, the most likely change-point position j prior to t and the most likely model of this segment from j to t is calculated using an online Viterbi algorithm. This is done by determining the posterior probabilities for each segment which ends at t and for all data models. The algorithm calculates for each $t > 0, j = 0, \ldots, t-1$, and every model $m, m_v \in \mathcal{M}$:

$$P_t(j, m, m_v) = (1 - G(t-j-1))p(y_{j+1:t}|m, m_v) \cdot p(m)p(m_v)P_j^{MAP}, \quad (8)$$

and

$$P_t^{MAP} = \max_{j,m} \left(\frac{P_t(j, q)g(t-j)}{1 - G(t-j-1)} \right). \quad (9)$$

Equation (8) gives the probability that the most recent change-point prior to t occurs at time j with models m and m_v for the segment $y_{j+1:t}$ that has a length of at least $t-j$. The first term is the probability that the assumed segment starting at $j+1$ has a length of at least $t-j$. It is multiplied with the marginal likelihood of that segment having models m and m_v, $p(y_{i+1:j}|m, m_v)$, times the prior probability of the models, $p(m)$ and $p(m_v)$. The last term P_t^{MAP} denotes the most likely change-point position prior to j. In (9) the most probable j, m, and m_v are determined. We chose the initial P_0^{MAP} to be $1/|\mathcal{M}|$. Because the probabilities $P_t(j, m, m_v)$ are very close to zero for most of the possible segments, a particle filter as proposed in [10] is used to reduce computation time.

IV. DATA ACQUISITION

For the evaluations done in this paper, several manipulation movements were recorded with a Qualisys motion capture system, consisting of seven Oqus300 cameras.[2] With this system, the positions of makers are tracked using infrared light, with less than 1 mm deviation, at a maximal frame rate of 500 Hz. To track manipulation movements of a human, markers were attached to the arm, hand, and back of the subject. The positions can be seen in Fig. 3. At the back of the hand as well as on the back of the subject a marker cluster consisting of three markers were attached. With this, the orientation of the hand and the back can be determined. Single markers were used to record positions near the shoulder and the elbow joint of the subject.

Three different manipulation movements were recorded at 500 Hz from several subjects. In the first dataset, which we call **step-data**, a stick was moved through a step pattern, see Fig. 1. In this movement, the position of the hand is very restricted, as the subjects have only little possibilities to move the stick differently than on the straight line through the pattern. Thus, the subjects are forced to do point-to-point movements to go through the step pattern. Due to the restricted movement position, the recorded movements can be segmented into the main building blocks *up, down, right*, and *left*, with ground truth segment borders at the corners of the step pattern as indicated with circles in Fig. 1. The movement through the step pattern was recorded for six different subjects. Each subject performed several repetitions of the movement, where the task in one repetition was to move the stick from lower left of the pattern to the upper right and back, resulting in eight point-to-point movements. This was repeated several times for roughly three minutes with a short break after each minute. The subjects were instructed to perform the movement as precisely as possible, i.e., without hitting the borders, while moving as fast as possible. In total, 171 repetitions of the movement were recorded, which corresponds to 1368 movement building blocks.

Two additional datasets were used for evaluation in this paper, which consist of movement sequences where the subjects had more freedom in movement execution. The **stick-throwing** movements were previously used as demonstrations for imitation learning in [24]. In these movement recordings, the subjects were instructed to throw a stick into a cylindrical box from a fixed distance, see Fig. 3. The throwing movements from three subjects were recorded with the subjects performing the throw 10, 11, and 13 times respectively. Each throw consisted of a strike out phase, the actual throwing movement, and a swing out phase. In between, short sequences of no movement could be observed. Thus, the throwing data can be divided into four movements classes. In total, 34 throwing demonstrations were available containing 126 movement building blocks.

In the **pick-and-place** dataset, which was previously used to evaluate movement classifiers in [25], the subjects grasped a box from a shelf, placed it on a table standing on the right side of the subject, and placed the box back into the shelf. This dataset was recorded at 60Hz. To increase movement variability, the exact position where the box should be placed was not specified. After placing the box on the table or the shelf, the subject moved the arm into a resting position in which the arm lies relaxed besides the body. This resulted in movements composed of seven different segments assigned to the following movement categories: *approach forward*, *move object to table*, *move to rest right*, *approach right*, *move object to shelf*, *move to rest down*, and the class *idle* for periods in which the subjects did not move their arms. The pick-and-place task was performed by three different subjects, repeated nine times by one of these and six times by the other two. In total, 172 pick-and-place building blocks were recorded. In [25], the vMCI algorithm was used to segment these demonstrations, but without comparing the results to a manually generated ground truth.

All movement data was conducted in accordance with the declaration of Helsinki and approved with written consent by the ethics committee of the University of Bremen. Subjects gave informed and written consent to participate.

[2] www.qualisys.com

FIGURE 3. Setups for movement recordings. Left: Setup for stick-throwing recordings. (Image extracted from [24].) Right: Pick-and-place task, in which a box is grasped from a shelf, placed on a table, and vice versa. (Image adapted from [25].)

A. MANUAL SEGMENTATION

In order to analyze different features of the recorded movements and to evaluate the automatic segmentation, a ground truth segmentation is needed. Whereas this ground truth segmentation comes naturally with the position of the stick in the step-data, it becomes more difficult to determine exact segment borders with increasing movement complexity. For labeling recorded movement trajectories manually, a time series visualization tool was implemented at our institute. With this tool, the 3D position of the markers can be visualized as time series in a two-dimensional plot, which is synchronized to a 3D visualization of the marker positions. The tool allows to directly define segmentation points in the time series by clicking on the time frame at which a segment border is assumed. By using this labeling tool, all recorded movements were manually segmented by the same person to generate a ground truth for the evaluations in this paper.

For the step-data, the segment borders coincidence with the corners of the step pattern, which segments the data into the movements *up, down, right*, and *left*. The stick-throwing and pick-and-place data were segmented into individual segments referring to the four and seven different classes contained in the movements as described in the previous paragraph. Please note that all movements were manually segmented solely based on the visualization of the positions of the markers. With increasing complexity in terms of increasing movement freedom, the movement borders were more difficult to determine which may have resulted in an imperfect ground truth segmentation.

V. COMPARISON OF FEATURES FOR MANIPULATION BUILDING BLOCKS

Experiments in the literature indicate that in human arm movements bell-shaped curves in the velocity of the hand are a characteristic feature of movement segments [7], [8]. We used the data recordings of the reference step pattern and the pick-and-pace data to examine this theory on movement data recorded at high precision. Using the position-based visualization, the point-to-point movements of the step pattern and the pick-and-place movements were manually divided into the 4 and 7 movement building blocks as described in the previous section. Parts of the movement which could not be assigned to one of the building block classes remained unsegmented and were not considered for further analysis. For all movement segments, the following features were calculated from the recorded marker positions: 1. hand velocity in x, y, and z direction; 2. absolute velocity of the hand; 3. elbow joint velocity calculated using the temporal derivative of the elbow joint position, which is determined using the angle between lower and upper arm; and 4. shoulder joint velocity calculated using the temporal derivative of the shoulder joint position, which is calculated using the angle between upper arm and upper body. The features of all segments were normalized to the same time period between 0 and 1 seconds. For all segment features belonging to the same movement class, the mean value and its standard deviation were calculated. This mean feature value of each class is visualized to determine similarities and differences in the features of the different manipulation movements.

The resulting plots for the point-to-point movement building blocks are shown in Fig. 4. In these plots, the 2D position of the hand is visualized as well as the mean values of the four calculated features. Each segment class is performed two times in one demonstration of the step pattern movement. Additionally, the reference setup was located at different positions in the global coordinate frame for individual movement recordings, so that movements belonging to the same movement class can be located at different positions (first column in Fig. 4). Based on the direction of the movement, the highest velocity can be observed in a different coordinate of the hand position. For some movement classes an increasing elbow joint velocity can be observed. In the *up* and *down* movements, the main change in the velocity is in

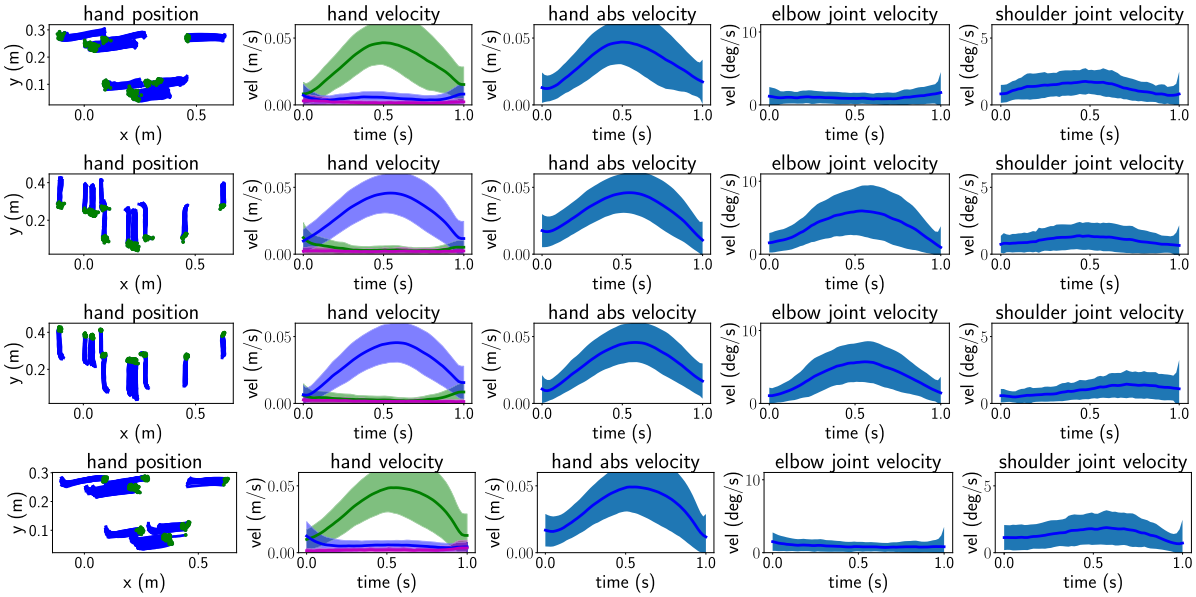

FIGURE 4. Different features of the point-to-point movements in the reference testbed shown in Fig. 1. Shown is the mean value and standard deviation of the point-to-point movements *right*, *up*,*down*, and *left*(columns 1-4) for the movement features: hand velocity in x, y, and z direction (lines colored in green, blue, and magenta), absolute hand velocity, elbow joint velocity, and shoulder joint velocity. In the first column the position of all analyzed segments is plotted. The only feature that looks similar for all movements is the absolute hand velocity, which is a bell-shaped curve.

the y coordinate of the hand and in the elbow joint velocity. A change in the x coordinate of the hand can be observed in the movements belonging to the classes *right* and *left*, without an increasing elbow joint velocity. The only feature that looks similar for all movement segments regardless of the position and direction of the performed movement is the absolute velocity of the hand.

The visualization of the features for the pick-and-place movements can be seen in Fig. 5. In comparison to the point-to-point movements, these movements are three-dimensional and not restricted in the position of the hand. The participants were free in the selection of the exact position to place the box on the table or shelf. This resulted in more variety in movement execution. Nonetheless, the common feature for all building blocks of the pick-and-place task is the bell-shaped pattern in the absolute velocity of the hand, whereas the angular velocity of the elbow and shoulder joint differ between building block movements. Especially the elbow joint velocity shows multiple minima within one movement class, which would result in an over-segmentation if segment borders would be assumed at points where this angular velocity is minimal. However, compared to the simple point-to-point movements, the absolute hand velocity shows more variations for different building blocks in the pick-and-place movements with the main impact that the velocity peak is for some movement classes clearly shifted to one side, for example for the *move object to shelf* class. These variations in the bell-shaped absolute velocity are considered in the vMCI algorithm by the parameters c and r in (2).

These observations support the assumption that manipulation building blocks may be characterized using the absolute velocity of the hand. As shown in the next section, segmentation based on these findings improves segmentation accuracy.

VI. EXPERIMENTAL EVALUATION OF MOVEMENT SEGMENTATION

The vMCI algorithm was already evaluated with respect to the segmentation accuracy in the presence of noise in the data and to the influence of the parameter selection in comparison to a segmentation method based on local minima (locMin), the MCI, and the BPARHMM algorithm on a synthetic dataset and three real human movements of one subject in the restricted environment of the reference step setup in [9]. Additionally, the vMCI algorithm was run on ball-throwing movements in the same publication without a comparison to a ground truth segmentation. In this paper, we extend the evaluation of the vMCI segmentation on movement examples of the reference setup using a bigger data set consisting of multiple demonstration of different subjects as introduced in section IV. Next to the comparison to MCI, locMin, and BPARHMM, we additionally compared vMCI to the ProbS algorithm, which is a promising segmentation technique published recently. The algorithms were first evaluated on movement recordings in the restricted environment of the step reference setup, consisting of point-to-point movements with only little variation in the position between different recordings but with noisy velocity patterns. Additionally, we evaluated all approaches on the stick-throwing data which have a very smooth velocity but with higher variations in the positions between recordings and on the pick-and-place data. We compared the algorithms using the F1-score, which is the harmonic mean of precision and recall. Additionally,

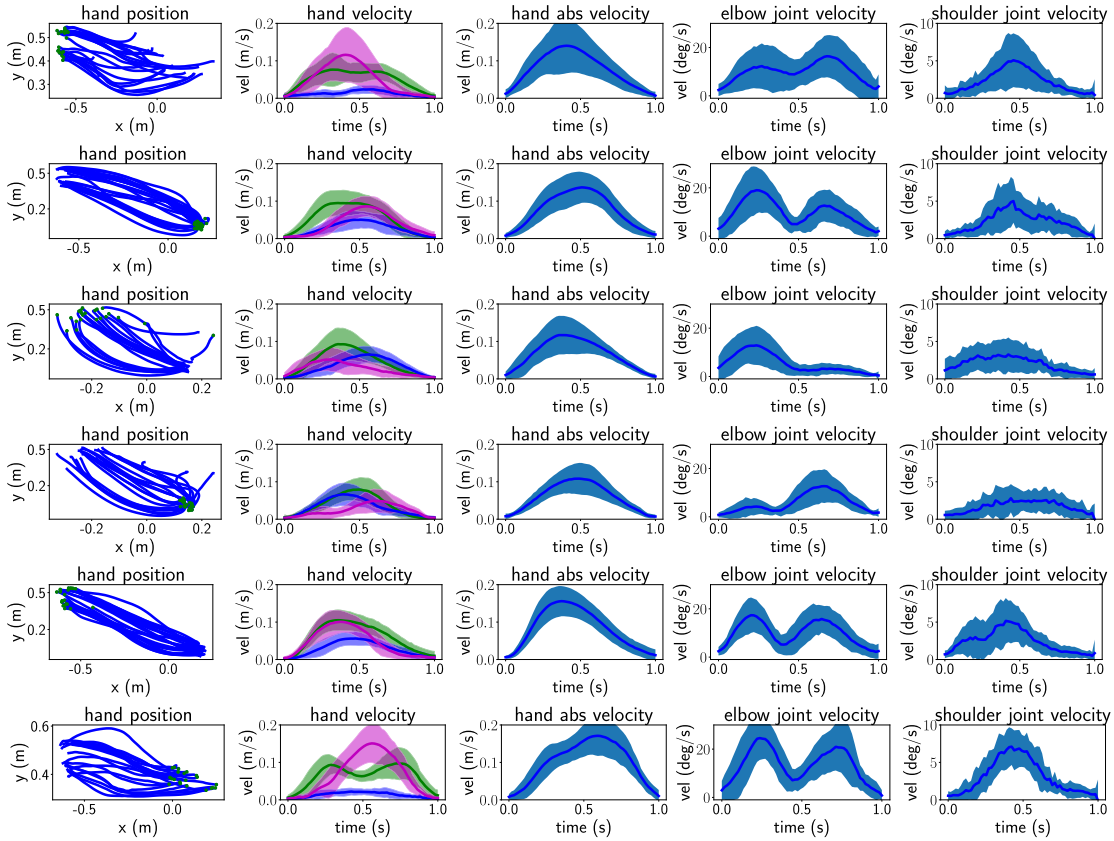

FIGURE 5. Different features of the pick-and-place movements. Shown is the mean value and standard deviation of the pick-and-place movements *approach forward, move object to table,move to rest right, approach right, move object to shelf,* and *move to rest down* (columns 1-6) for the movement features: hand velocity in x, y, and z direction (lines colored in green, blue, and magenta), absolute hand velocity, elbow joint velocity, and shoulder joint velocity. In the first column the position of all analyzed segments is plotted. The only feature that looks similar for all movements is the absolute hand velocity, which is a bell-shaped curve.

we investigated the number of true positives (TP) and false positives (FP) in more detail to deduce how many of the segments contained in the data are correctly detected by the methods.

In all experiments, the basis functions ϕ of the LRM in vMCI, MCI, and BPARHMM were chosen to be autoregressive with order $q = 1$, i.e., $\phi(x_t) = y_{t-1}$. For this, the data was preprocessed to a mean of zero and such that the variance of the first order differences of each dimension is equal to 1. This preprocessing was not done for the velocity dimension if vMCI was used. The parameter for the distribution of the segment length p in the MCI and vMCI algorithm was fixed to $p = 0.02$ because it has a small influence on the segmentation results due to the Bayesian model of the algorithm [9]. The parameter D, which regulates the variance of the model parameters β along the data dimensions, was set to the identity matrix. This is a good choice because an autoregressive basis is chosen and the data is preprocessed to a variance of one. The parameters S and ν influence the variance of the weights as well as the Gaussian noise of the LRM along the time dimension. These parameters can directly be calculated from the data to estimate the true variance by determining the variance of the first order differences of the data along the time dimension. As shown in [9], these hyper-parameters have only little influence on the segmentation result. In the vMCI algorithm, the number of centers was fixed to three and the hyper-parameters were calculated identically to the ones of the position LRM from the velocity data.

To run BPARHMM, next to the selection of the basis function and corresponding hyper-parameters, several other hyper-parameters must be set in advance. We used the same hyper-parameter configuration for all datasets based on the suggestions made in [11]. The sampling algorithm was run 5 times, with 5000 iterations each. To compare to ProbS, we implemented the probabilistic segmentation based on the formulas given in [12]. In this algorithm, the data is represented using DMPs. Based on an initial over-segmentation the algorithm infers the number and parameters of the DMPs needed to generate the observed data. We used locMin to generate the initial over-segmentation. Due to the computationally intensive inference steps, we limited the number of iterations to 50 and run the algorithm separately for demonstrations of each subject. The locMin approach detects a segmentation point at positions where a local minimum occurs in the velocity in a predefined window [9]. This window was varied for each dataset and the margin with the best result was selected for the final evaluation.

TABLE 1. Mean segmentation results on step pattern movements.

	F1-score	num. TP (opt.: 7)	num. FP (opt.: 0)
vMCI	**0.85**	6.0	**1.1**
MCI	0.63	4.0	1.2
locMin	0.83	**7.0**	3.7
BPARHMM	0.81	6.9	3.1
ProbS	0.18	1.5	1.9

A. SEGMENTATION OF POINT-TO-POINT MOVEMENTS

For the automatic segmentation of the step data, the recorded movements were down-sampled to 30 Hz. In the locMin approach, the window size was set to 0.27 seconds and for pre-segmentation in ProbS the window size was set to 0.13 seconds to achieve an over-segmentation. All other parameters of the compared algorithms were selected as described in the previous paragraph.

The results of evaluated algorithms can be seen in Table 1, in which the mean F1-score, TP, and FP are shown. The mean values were determined using all 171 demonstrations of going through the step pattern upwards and back, resulting in an optimal segmentation into eight building blocks which corresponds to an optimal number of true positives of seven. A determined segmentation point was treated as correct when it laid within a margin around the ground truth segmentation point of 0.2 seconds. The dataset contained examples with very smooth velocity changes as well as noisy velocity profiles. An example result of the segmentation using the different approaches on a sample with noisy velocity is shown in Fig. 6.

Using the simple locMin approach, all segmentation points could be detected in all movement examples, but with a mean value of 3.7 FP. The lowest number of FP was achieved using vMCI, which detected on average 6 segmentation points with 1 FP. With this, the vMCI algorithm had the highest F1-score on this dataset and outperformed all other approaches, whereby locMin and BPARHMM over-segmented the data. ProbS was not able to detect the segmentation points at all, possibly because there was only little variation in movement execution in the data, i.e., nearly the whole demonstration can be represented by the same DMP. However, on movement examples with a noisy velocity, ProbS over-segmented the data, see the example in Fig. 6. In this experiment, vMCI achieved a perfect segmentation despite the noise in the velocity. However, this noise resulted in an over-segmentation using locMin, ProbS and BPARHMM.

B. SEGMENTATION OF STICK-THROWING MOVEMENTS

On the stick-throwing dataset, the same parameters configurations were used as for the step data for vMCI, MCI, BPARHMM, and ProbS. The window for locMin was set to 0.2 seconds and 0.07 seconds for initial over-segmentation needed in ProbS.

The mean segmentation results for all algorithms can be seen in Table 2. For this dataset, the optimal number of TP is 3.7, i.e., the throwing movements are on average segmented

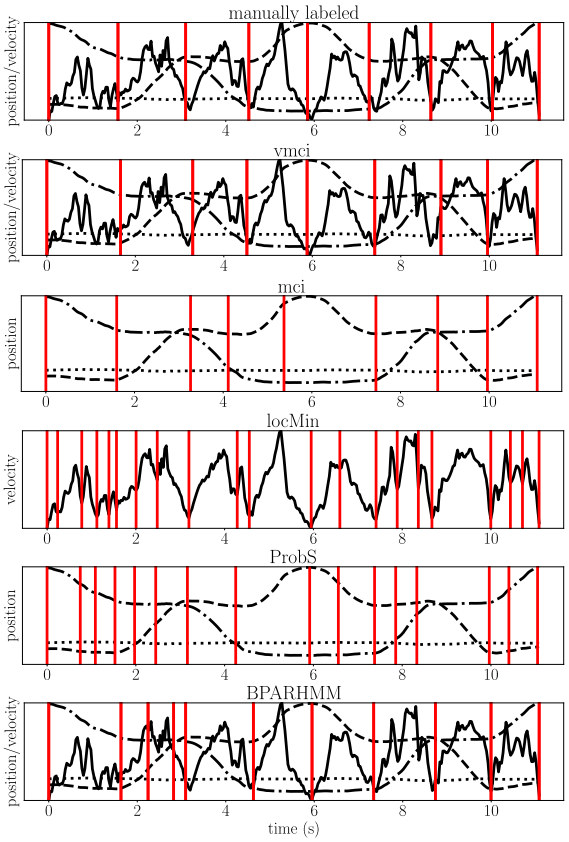

FIGURE 6. Example demonstration of a movement through the step pattern, consisting of the building block sequence *right*, *up*, *right*, *up*, *down*, *left*, *down*, *left*. Shown is the absolute velocity (solid line) and the position (x,y,z - dashed/dotted lines) of the hand. The top left plot shown the manual segmentation result. The other plots show the result of vMCI, MCI, locMin, ProbS, and BPARHMM, in which the data which is used as basis for segmentation is plotted (position and/or velocity).

TABLE 2. Mean segmentation results on stick-throwing movements.

	F1-score	num. TP (opt.: 3.7)	num. FP (opt.: 0)
vMCI	**0.70**	1.9	**0.9**
MCI	0.58	1.5	**0.9**
locMin	**0.71**	**2.0**	1.8
BPARHMM	0.56	2.0	2.5
ProbS	0.23	0.6	1.0

into the three main building blocks *strike out*, *throw*, *swing out*, and some occurrences of additional segments. Again, the highest number of detected TP was achieved using locMin and the lowest number of FP using vMCI, where both algorithms have a similar F1-score of approximately 0.7. With that, vMCI, and locMin outperform all other approaches. On this dataset, in which nearly no noise can be observed, BPARHMM and locMin again detected some FP, but not as much as on the step-data. Again, ProbS does not perform well on this data. An example result can be seen in Fig. 7.

C. SEGMENTATION OF PICK-AND-PLACE MOVEMENTS

To segment the pick-and-place dataset, the recorded data was down-sampled to 20 Hz, because in this dataset the movement were on average much slower (mean segment length

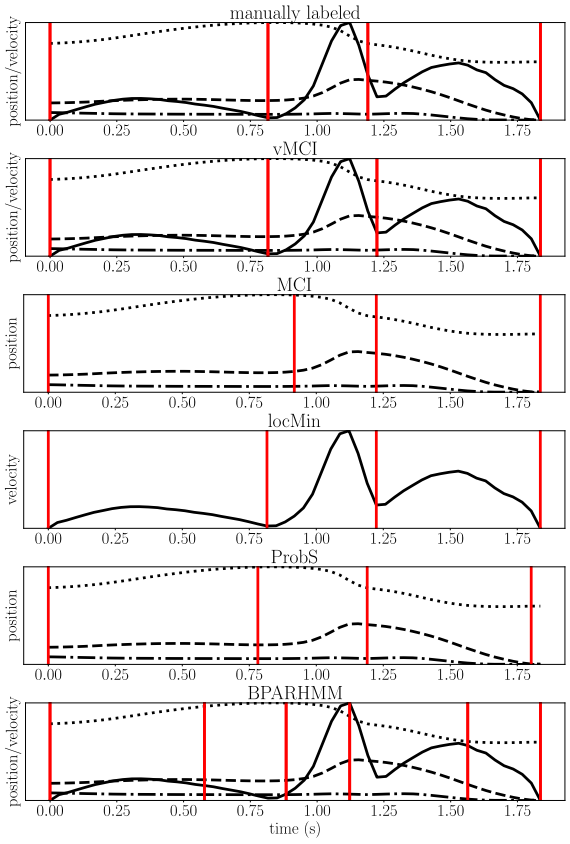

FIGURE 7. Example demonstration of a stick-throwing movement, consisting of the building blocks *strike out*, *throw*, and *swing out*. Shown is the absolute velocity (solid line) and the position (x,y,z - dashed/dotted lines) of the hand. The top left plot shown the manual segmentation result. The other plots show the result of vMCI, MCI, locMin, ProbS, and BPARHMM, in which the data which is used as basis for segmentation is plotted (position and/or velocity).

of 1.3 seconds) compared to step dataset (mean segment length of 0.82 seconds). The parameters for vMCI, MCI, BPARHMM, and ProbS were identical to the ones in the previous evaluations. The range in which the minima are detected in locMin was set to 0.2 seconds and 0.1 seconds for the initial over-segmentation in ProbS. Due to the slower movement speed, the margin in which a detected segmentation point is still treated as correct is increased to 0.3 seconds.

This dataset has a bigger movement variability compared to the other two datasets and the ground truth segmentation was more difficult to manually define. Due to rather slow movements when the object is grasped or placed, the exact movement start and end point were difficult to determine based on the position of the hand. If the subject did not move or did a movement which was not part of the defined movement classes, the segment borders were still added to the ground truth data. To accomplish the pick-and-place tasks, the subject had to perform 6 basic movements: *approach forward*, *move object to table*, *move to rest right*, *approach right*, *move object to shelf*, *move to rest down*. In between there are *idle* phases. The average number of manually defined segmentation points per demonstration was 8.1.

TABLE 3. Mean segmentation results on pick-and-place movements.

	F1-score	num. TP (opt.: 8.1)	num. FP (opt.: 0)
vMCI	0.67	5.0	3
MCI	0.60	4.9	4.0
locMin	**0.79**	6.5	3.0
BPARHMM	0.52	**6.3**	10.7
ProbS	0.32	1.6	**1.1**

As can be seen in the results in Table 3, most TP could be detected using BPARHMM and locMin, but with a high number of FP using BPARHMM. On this data, vMCI segmentation resulted in a higher number of FP compared to the other two datasets. If the results on the individual demonstrations are examined more closely, one can observe that vMCI detected most of the segments but with a very inaccurate position of the segment boundaries, which lie often outside the margin of 0.3 seconds. This can also be seen in the example result in Fig. 8. Again, the segmentation points could not be reliably detected using ProbS.

VII. DISCUSSION

In the comparison of several features in human point-to-point and pick-and-place movements in section V, a bell-shaped curve in the absolute velocity of the hand was identified as a feature which was identical for all movement segments, independent from the movement direction. In the experimental evaluations of vMCI run on the same point-to-point movements of the step reference setup (section VI-A) it can be seen that vMCI was able to detect these segments with bell-shaped hand velocity. Also in movement examples with a very noisy velocity, the majority of the true segmentation points were detected using vMCI. This confirms the results we obtained on a small number of demonstrations of the step setup of one subject in our earlier publication. Based on the velocity features, segmentation points could also reliably be detected using locMin. In comparison to vMCI, the resulting segment borders were more accurate because they are located directly a time points where the velocity was in a local minimum. However, the threshold in locMin needs to be adapted to different datasets or even different movement examples of the same tasks executed in different speeds. Furthermore, in data with a noisy velocity profile locMin over-segmented the data and detects a lot of false positives. By using vMCI, this over-segmentation can be prevented without an additional preprocessing, such as smoothing.

In the evaluation performed on the stick-throwing and pick-and-place data, vMCI again yielded good results and detected the majority of the segmentation points correctly. The ground truth segmentation points were determined based on the recorded marker positions and the vMCI approach detected these based on the position and velocity of the hand marker. In these more complex manipulation movements compared to the point-to-point movements, building blocks could still be detected based on the bell-shaped velocity of the hand using the vMCI approach.

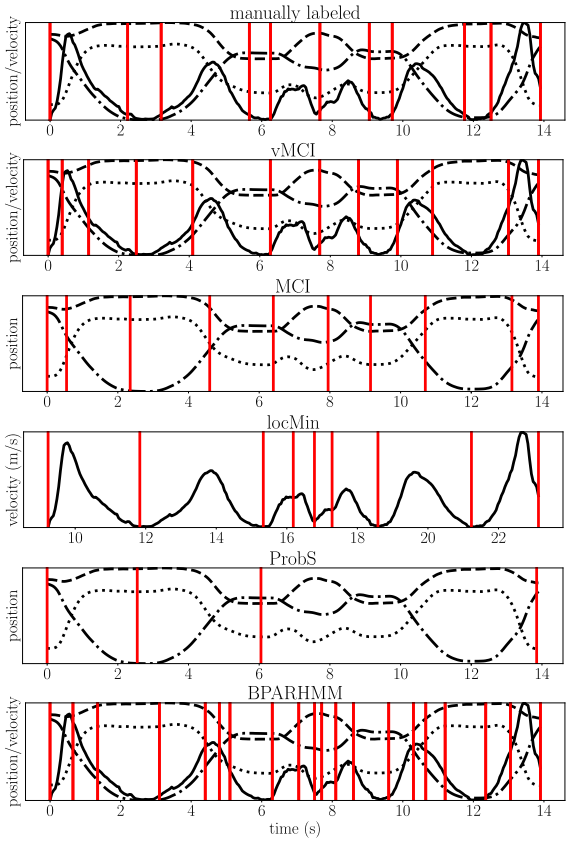

FIGURE 8. Example demonstration of the pick-and-place movement, consisting of the building blocks *approach forward, move object table, go to rest right, approach right, move object shelf, move to rest,* with small periods of no movement in between (building block *idle*). Shown is the absolute velocity (solid line) and the position (x,y,z - dashed/dotted lines) of the hand. The top left plot shown the manual segmentation result. The other plots show the result of vMCI, MCI, locMin, ProbS, and BPARHMM, in which the data which is used as basis for segmentation is plotted (position and/or velocity).

In all analyzed movements, but especially in the pick-and-place data, vMCI detected some segment points inaccurately, resulting in a lower number of TP with increasing number of FP. However, the total number of detected segments was nearly the same as the manually defined number of segments in all experiments. This indicates that the data is not over-segmented using vMCI.

On all three datasets, vMCI performed better than the MCI algorithm which does not model the velocity dimension of the data differently than the position. This also applies to the comparison to BPARHMM, which models the data in a very similar data as vMCI and MCI but does not have the special look at the velocity. Using BPARHMM, also good results could be achieved but the algorithm is more sensitive to noise. Furthermore, it is a batch method which needs a lot more computation time and cannot be run online.

In the performed experiments, we were not able to achieve good results using our implementation of the ProbS algorithm. Although it was shown in [12] that the algorithm converges, this could not be achieved in a reasonable time period in our evaluations and calculations had to be stopped after a fixed number of iterations. On all observed movements, only unsatisfying results could be obtained on most of the movement examples using this method. Possibly, this results from the design of the algorithm which identifies a set of basic movement the analysed demonstrations are generated from. In the experiments of Lioutikov et al, the algorithm was tested on movement demonstrations which contain different combinations of a small number of basic movements. In that data, ProbS successfully identified basic movements [12]. Our datasets, on the other hand, contain repetitive demonstrations of the same manipulation movement without changes in the order of the concatenated building blocks. Our results indicate, that ProbS is not suited to segment this kind of data. However, BPARHMM and ProbS give not only the segmentation points but also clustered segments that describe which segments belong to the same movement. For vMCI, a successive step is needed to obtain grouped or labeled segments. Due to the simple structure of the building blocks detected using vMCI, we showed that labeled segments can be obtained using a simple k-Nearest Neighbor classifier with $k = 1$ and a small number of training examples, see [26].

VIII. CONCLUSION

In this paper, we complemented experiments conducted in related publications and showed that the velocity of the hand is a relevant feature to characterize building blocks in human manipulation movements. We presented the vMCI algorithm which detects these building blocks with a bell-shaped profile in the hand velocity in an unsupervised and online manner. In vMCI, this knowledge about characteristics of manipulation movements is directly integrated into the segmentation process. Evaluations performed on different manipulation datasets showed that building blocks can reliably be detected using vMCI, also in movement examples with a noisy velocity. In comparison to other state of the art segmentation methods, segmentation points can be determined fast and the algorithm can be applied to different movements without parameter tuning.

Because the bell-shaped velocity is a movement feature that can mainly be observed in manipulation movements, the vMCI approach works best on this kind of data. In other movements, such as gestures or generative movements such as walking, building blocks may show different reoccurring patterns. To detect building blocks in these movements, other approaches may be needed. Furthermore, vMCI is designed for single-handed manipulation movements. To segment dual-arm movements into meandingful building blocks, vMCI was extended to a hierarchical segmentation approach in [27].

Overall, our experiments show that automatic approaches for movement segmentation can benefit from taking regularities in human movements into account. For future work, it should be evaluated if the resulting manipulation building blocks can be used in robotic applications to generate different behaviors using learning from demonstrations. First applications already show that meaningful goal-directed

robotic movement can be generated using the building blocks obtained using vMCI [2].

REFERENCES

[1] B. D. Argall, S. Chernova, M. Veloso, and B. Browning, "A survey of robot learning from demonstration," *Robot. Auto. Syst.*, vol. 57, no. 5, pp. 469–483, 2009.
[2] L. Gutzeit, A. Fabisch, M. Otto, J. H. Metzen, J. Hansen, F. Kirchner, and E. A. Kirchner, "The BesMan learning platform for automated robot skill learning," *Frontiers Robot. AI*, vol. 5, p. 43, May 2018.
[3] E. Adi-Japha, A. Karni, A. Parnes, I. Loewenschuss, and E. Vakil, "A shift in task routines during the learning of a motor skill: Group-averaged data may mask critical phases in the individuals' acquisition of skilled performance," *J. Experim. Psychol., Learn., Memory, Cognition*, vol. 34, no. 6, pp. 1544–1551, Nov. 2008.
[4] A. M. Graybiel, "The basal ganglia and chunking of action repertoires," *Neurobiol. Learn. Memory*, vol. 70, nos. 1–2, pp. 119–136, Jul. 1998.
[5] F. A. Mussa-Ivaldi and S. A. Solla, "Neural primitives for motion control," *IEEE J. Ocean. Eng.*, vol. 29, no. 3, pp. 640–650, Jul. 2004. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1353417
[6] T. Flash and B. Hochner, "Motor primitives in vertebrates and invertebrates," *Current Opinion Neurobiol.*, vol. 15, pp. 600–666, Jan. 2005.
[7] P. Morasso, "Spatial control of arm movements," *Exp. Brain Res.*, vol. 42, no. 2, pp. 223–227, Apr. 1981.
[8] P. Morasso and F. A. M. Ivaldi, "Trajectory formation and handwriting: A computational model," *Biol. Cybern.*, vol. 45, no. 2, pp. 131–142, Sep. 1982. [Online]. Available: http://link.springer.com/article/10.1007/BF00335240
[9] L. Senger, M. Schroer, J. H. Metzen, and E. A. Kirchner, "Velocity-based multiple change-point inference for unsupervised segmentation of human movement behavior," in *Proc. 22nd Int. Conf. Pattern Recognit.*, Aug. 2014, pp. 4564–4569.
[10] P. Fearnhead and Z. Liu, "On-line inference for multiple change point models," *J. Roy. Stat. Soc., B, Stat. Methodol.*, vol. 69, pp. 589–605, Jan. 2007.
[11] E. Fox, E. Sudderth, M. Jordan, and A. Willsky, "Sharing features among dynamical systems with beta processes," in *Proc. Conf. Neural Inf. Process. Syst. (NIPS)*. Cambridge, MA, USA: MIT Press, 2009, pp. 549–557.
[12] R. Lioutikov, G. Neumann, G. Maeda, and J. Peters, "Learning movement primitive libraries through probabilistic segmentation," *Int. J. Robot. Res.*, vol. 36, no. 8, pp. 879–894, 2017.
[13] J. K. Aggarwal and Q. Cai, "Human motion analysis: A review," *Comput. Vis. Image Understand.*, vol. 73, no. 3, pp. 428–440, Mar. 1999. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1077314298907445
[14] J. F.-S. Lin, M. Karg, and D. Kulić, "Movement primitive segmentation for human motion modeling: A framework for analysis," *IEEE Trans. Hum.-Mach. Syst.*, vol. 46, no. 3, pp. 325–339, Jun. 2016.
[15] A. Fod, M. J. Matarić, and O. C. Jenkins, "Automated derivation of primitives for movement classification," *Auto. Robots*, vol. 12, no. 1, pp. 39–54, Jan. 2002.
[16] O. C. Jenkins and M. J. Mataric, "Automated derivation of behavior vocabularies for autonomous humanoid motion," in *Proc. 2nd Int. Joint Conf. Auto. Agents Multiagent Syst.*, 2003, pp. 225–232.
[17] S. Chiappa and J. Peters, "Movement extraction by detecting dynamics switches and repetitions," in *Proc. Adv. Conf. Neural Inf. Process. Syst. (NIPS)*, 2010, pp. 1–15.
[18] D. Kulic, D. Lee, J. Ishikawa, and Y. Nakamura, "Incremental learning of full body motion primitives and their sequencing through human motion observation," *Int. J. Robot. Res.*, vol. 31, no. 3, pp. 330–345, 2012.
[19] D. Gong, G. Medioni, S. Zhu, and X. Zhao, "Kernelized temporal cut for online temporal segmentation and recognition," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 229–243.
[20] G. Konidaris, S. Kuindersma, R. Grupen, and A. Barto, "Robot learning from demonstration by constructing skill trees," *Int. J. Robot. Res.*, vol. 31, no. 3, pp. 360–375, 2012.
[21] A. J. Ijspeert, J. Nakanishi, H. Hoffmann, P. Pastor, and S. Schaal, "Dynamical movement primitives: Learning attractor models for motor behaviors," *Neural Comput.*, vol. 25, no. 2, pp. 328–373, Feb. 2013.
[22] C. Panagiotakis, G. Karvounas, and A. Argyros, "Unsupervised detection of periodic segments in videos," in *Proc. 25th IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2018, pp. 923–927.
[23] C. M. Bishop, *Pattern Recognition and Machine Learning*. New York, NY, USA: Springer-Verlag, 2006.
[24] L. Gutzeit, A. Fabisch, C. Petzoldt, H. Wiese, and F. Kirchner, "Automated robot skill learning from demonstration for various robot systems," in *Proc. Joint German/Austrian Conf. Artif. Intell.*, C. Benzmüller and H. Stuckenschmidt, Eds., vol. 11793. Cham, Switzerland: Springer, 2019, pp. 168–181.
[25] L. Gutzeit, M. Otto, and E. A. Kirchner, "Simple and robust automatic detection and recognition of human movement patterns in tasks of different complexity," in *Physiological Computing Systems*. Cham, Switzerland: Springer, 2016, pp. 39–57.
[26] L. Gutzeit, "A comparison of few-shot classification of human movement trajectories," in *Proc. 10th Int. Conf. Pattern Recognit. Appl. Methods*, 2021, pp. 243–250.
[27] L. Gutzeit, "Hierarchical segmentation of human manipulation movements," in *Proc. 26th Int. Conf. Pattern Recognit. (ICPR)*, Aug. 2022, pp. 2742–2748.

LISA GUTZEIT received the bachelor's and master's degrees in computational life science from the University of Lübeck, Germany, in 2009 and 2011, respectively. She is currently pursuing the Ph.D. degree with the University of Bremen, Germany. Since 2012, she has been working as a Researcher at the Robotics Research Group, University of Bremen. Since 2022, she has been leading the Team Human-Centered Interaction. Her scientific research interests include the fields of machine learning and human–machine interaction.

FRANK KIRCHNER received the degree in computer science and neurobiology from the University of Bonn, Germany, in 1994, and the Dr. rer. nat. degree in computer science from the University of Bonn, in 1999. Since 1994, he has been a Senior Scientist at Gesellschaft für Mathematik und Datenverarbeitung (GMD), Sankt Augustin, Germany. Since 1998, he has been a Senior Scientist at the Department for Electrical Engineering, Northeastern University, Boston, MA, USA, where he was first appointed as an Adjunct Professor and then a Tenure Track Assistant Professor. In 2002, he was appointed a Full Professor at the University of Bremen, Germany, and since 2005, he has also been the Director of the Robotics Innovation Center, DFKI, Germany. Since 2013, he has also been the Scientific Director of the Brazilian Institute of Robotics (BIR), Salvador, Bahia, Brazil, where he was awarded an Honorary Doctorate, in 2017, for his achievements in the field of robotics and artificial intelligence. His scientific research interests include the fields of robotics and AI. Since 2015, he has been a member of the Berlin Brandenburg Academy of Sciences and Humanities (BBAW).