IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received January 27, 2021, accepted April 1, 2021, date of publication April 26, 2021, date of current version May 4, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3075766

Content-Based Management of Human Motion Data: Survey and Challenges

JAN SEDMIDUBSKY, PETR ELIAS, PETRA BUDIKOVA, AND PAVEL ZEZULA
Department of Machine Learning and Data Processing, Faculty of Informatics, Masaryk University, 602 00 Brno, Czechia
Corresponding author: Jan Sedmidubsky (xsedmid@fi.muni.cz)

This work was supported by the Czech Science Foundation under Project GA19-02033S.

ABSTRACT Digitization of human motion using skeleton representations offers exciting possibilities for a large number of applications but, at the same time, requires innovative techniques for their effective and efficient processing. Content-based processing of skeleton data has developed rapidly in recent years, focusing mainly on specialized prototypes with limited consideration of generic data management possibilities. In this survey article, we synthesize and categorize the existing approaches and outline future research challenges brought by the increasing availability of human motion data. In particular, we first discuss the problems of suitable representation and segmentation of continuous skeleton data obtained from various sources. Then, we concentrate on comparison models for assessing the similarity of time-restricted pieces of motions, as required by any content-based management operation. Next, we review the techniques for evaluating similarity queries over collections of motion sequences and filtering query-relevant parts from continuous motion streams. Finally, we summarize the usability of existing techniques in perspective application domains and discuss the new challenges related to current technological and infrastructural developments. We especially assess the existing techniques from the perspective of scalability and propose future research directions for dealing with large and diverse volumes of skeleton data.

INDEX TERMS Action detection, content-based processing, deep features, metric learning, motion capture data, skeleton sequences, similarity, sub-sequence search.

I. INTRODUCTION

With the ever-increasing number of everyday facts becoming digital, the permanent computational research challenge is to develop tools that provide relevant information needed by individual users. Though a lot has already been done, the spatial-temporal, complex, and bulky *human skeleton data*, sometimes called *motion-capture* or *stick-figure* data, certainly represent such a challenge. Application-driven early initiatives, especially in computer animation, healthcare, and sports, capture precise 3D skeleton data by specialized hardware technologies and markers attached to human bodies. Nowadays, less precise, typically 2D, skeleton data can be extracted from a simple video by pose-estimation software tools [1]. As a result, we can expect an explosion of skeleton data in the near future, which opens completely new application possibilities but also poses new challenges for research. At the brink of this new era of motion processing, this work surveys existing approaches to content-based management of human skeleton data and analyses their strengths and weaknesses from the perspective of large-scale data management, which will be vital for future applications.

Motion data processing is a large research field that comprises a number of issues, ranging from data acquisition to specific motion retrieval tasks. State-of-the-art research focuses mainly on the task of action recognition, i.e., selecting the correct semantic class for a given piece of motion data. This problem is typically solved by machine learning techniques and has attracted a lot of attention in many application areas, as witnessed also by several action-recognition surveys summarized at the end of this section. However, there are many other motion processing tasks that require data processing techniques beyond the machine learning. To better perceive the scope of motion processing and associated challenges, let us consider an example from the sports domain: a figure-skating competition is composed of performances of individual skaters, where each performance consists of many skating elements, e.g., jumps or spins. Given a (possibly extensive) collection of competition recordings, we might be interested in detecting all the triple-Axel jumps, finding the

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

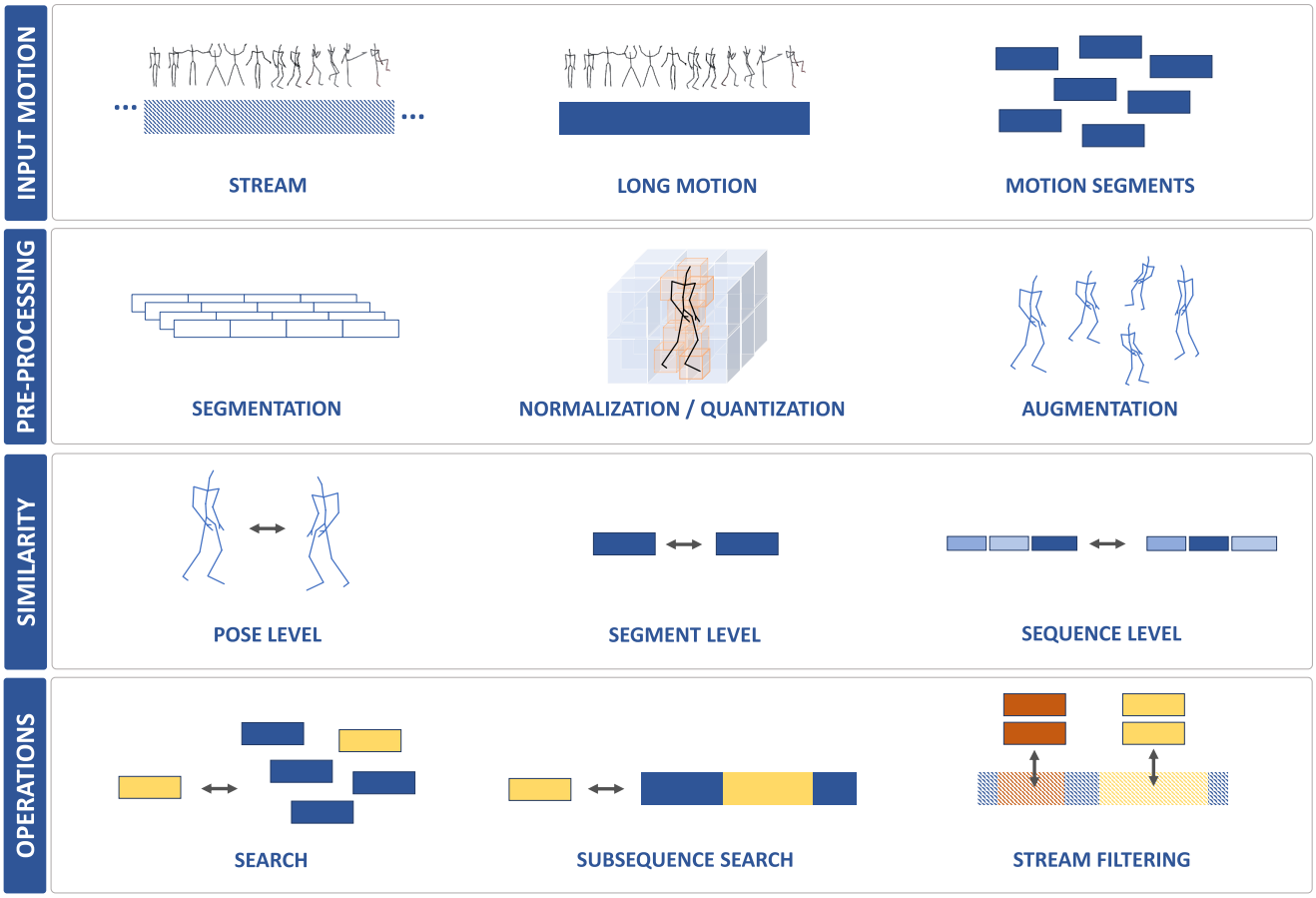


FIGURE 1. Overview of motion representations and motion processing objectives that are reviewed in this paper. The schema covers the typical motion processing pipeline: unsegmented motion streams (or long motions) are partitioned into short segments, pre-processed to enable the most effective similarity metric learning, and compared to facilitate efficient retrieval that can be employed in (sub-sequence) searching and filtering tasks.

most similar sub-motions to a given spin example, identifying the most common figures, or determining performances with similar choreographies. Alternatively, we might need to process a real-time stream of the motion data and provide annotations of elements currently being performed. For most of these tasks, it is vital to have effective motion similarity models as well as efficient data organization structures and retrieval algorithms that allow fast identification of similar motion segments.

This survey aims to categorize existing works in content-based processing of skeleton data, which has not been done in any previous study. Furthermore, we identify and discuss new research directions entailed by the rapid explosion of such data in terms of quantity, precision, and availability. More specifically, we focus on processing similarity-based user queries over motion data provided in the form of (1) a large collection of short pre-segmented motions that often correspond to specific application semantics, (2) a large collection of long motions without explicit information about their partitioning, or (3) a pseudo-infinite stream in which a limited motion content is available at a given time moment. To deal with such type of queries, a number of interrelated sub-problems need to be solved; some of them are illustrated in Figure 1.

The survey is structured as follows. After the clarification of basic concepts in Section II, we study the individual sub-problems of content-based motion data processing in Sections III-V: we discuss the challenges of individual tasks, review state-of-the-art techniques, and evaluate their strengths and weaknesses. More specifically, Section III introduces segmentation techniques that transform long motions or continuous motion streams into a sequence of short and meaningfully-comparable segments. Section IV deals with assessing similarity between two motion segments, which is an essential underlying operation required by most data-management tasks. This operation especially includes the extraction of content-preserving segment features, typically using different architectures of deep neural networks. In Section V, we describe how segment similarity is used for motion searching and filtering. We mainly review different approaches for query-by-example searching in collections of pre-segmented motions, sub-sequence searching in unsegmented long motions, and detecting events in continuous streams. In Section VI, we then shortly characterize current application environments, which were, in fact, the driving forces behind developing such techniques. In Section VII, we summarize the shortcomings of current data management techniques and discuss computational challenges that arise

especially when bulky and dirty 2D skeleton data are considered. We also claim that future research should focus on new types of operations and management of groups in order to support more powerful applications.

RELATED SURVEYS

The majority of existing motion processing surveys focus on motion classification in different contexts, e.g., recognizing classes of actions [4]–[6], two-person interactions [7], or person-object interactions [8]. Several studies discuss the possibilities of multi-modal motion recognition [5], [9], [10], focusing on the fusion of skeleton data with inertial sensor data [11], [12] or RGB and depth modalities [6]. Other works map the action recognition methods in specific application domains, e.g., gait recognition [11], martial arts [13], rehabilitation [14], or sports [15]. All these surveys focus on the processing of short pre-segmented motions, leaving aside the inherent continuous character of skeleton-data recordings. Several studies also cover skeleton-data acquisition approaches [5], [8], [10], [12], [13] or summarize publicly-available datasets [6]–[10].

II. SKELETON DATA DOMAIN

Motion data capture spatial and temporal components of human movement by recording positions of selected body points, typically *joints*, in time. Recorded joints captured at a given time moment form a *pose*, which can be visualized by a stick-figure resembling a *skeleton*. Therefore, human motion data are often denoted as *skeleton sequences*. The temporal dimension is captured by regular sampling of the moving person in time, which results in a sequence of consecutive poses $(P_1, P_2, \ldots)$. Each pose P_i is described by 2D/3D spatial coordinates of selected body joints. In the case of 3D data, $P_i \in \mathbb{R}^{j \cdot 3}$ represents the 3D skeleton configuration estimated at the time moment i and consists of the xyz-coordinates of j tracked joints; the 2D case is analogous.

A. SKELETON DATA ACQUISITION

Precise and view-invariant 3D skeletons can be obtained by high-end marker-based motion capture tracking systems (e.g., Vicon, xSens). Such data are preferred in expert analytical systems but require specialized hardware (e.g., optical or inertial sensors, markers) and proprietary software. Alternative approaches aim to optimize the accuracy-portability-cost trade-off by estimating the 3D skeletons from multiple synchronized digital cameras [2] or using low-end sensors such as the Microsoft Kinect. Recently, much attention has been devoted to extracting skeleton data from ordinary videos, which would assure mobile and cheap motion data acquisition. Several deep-learning pose estimators are available for extracting view-dependent 2D skeletons [3], [16], [17], but no guarantees on joint tracking accuracy can be given. Therefore, 2D skeletons are typically used for the analysis of general activities or high-level interactions. The most recent skeleton data acquisition trends involve direct 3D pose estimation from ordinary videos [1]. A high-level overview of existing acquisition methods is provided in Table 1. For more details about motion capturing methods, we refer to thorough comparisons in [15].

B. TYPES OF SKELETON SEQUENCES

An isolated skeleton pose does not contain any temporal motion context. Therefore, motion management focuses on the effective and efficient processing of the skeleton *sequences*. The sequences may appear in different forms – long or short, segmented or continuous, labeled or unlabeled. In this section, we define four types of sequences that play prominent roles in motion data processing. First, there are two types of long sequences that reflect two distinct modes in which the motion data can be produced and shared between applications:

- **Motion stream** (e.g., *a figure-skating performance stream*) – a pseudo-infinite feed-forward recording of poses that are broadcast in real-time; the stream is never available as a whole and hardly any assumptions on what comes next can be made;
- **Long motion** (e.g., *a recording of the whole figure-skating competition*) – a long persistent sequence of poses that is available as a whole; the long motion can be arbitrarily pre-processed, partitioned, and organized to ensure searchability.

Second, we need to distinguish between two types of short motion sequences. Both are used as data processing primitives, but they significantly differ on the semantic level:

- **Motion segment** (e.g., $(P_{750}, \ldots, P_{850})$) – a short sub-sequence of poses that is contained in a stream or in a long motion; the segments are not required to correspond to any semantic motion entities but serve as basic data organization units;

TABLE 1. Methods for 2D and 3D Skeleton acquisition.

Modality	Device / Method	Sensors	Joints	Frame-rate	Occl. resist.[1]	Error margin	Cost	Mobility	Markers
3D	Vicon[2] (optical sensors)	10–40	32	360	◐	mm	$$$	–	✓
	xSens[3] (inertial sensors)	17	17	240	●	mm–cm	$$	✓	✓
	Kinect v2[4] (RGB + depth)	2	25	30	○	cm	$	–	–
	synchronized video cameras [2]	3	25	~video	◐	cm	$	–	–
	video + xNect [1]	1	14	~video	○	>cm	$	✓	–
2D	video + hrnet [3]	1	16	~video	○	>cm	$	✓	–

[1] Degree of resistance towards occlusions – resistant (●), partially resistant (◐), not resistant (○)
[2] https://www.vicon.com/; [3] https://www.xsens.com/; [4] https://developer.microsoft.com/en-us/windows/kinect/

- **Action** (e.g., *the Axel jump*) – a segment or multiple consecutive segments with a clear semantics that is subjective to an observer, typically expressed by a textual label that is assigned either by a human or by a machine; actions are the smallest semantic units that are relevant to users.

Most existing research works assume precise 3D skeleton data acquired from specialized hardware or synchronized cameras. These devices are typically used in controlled environments to record short or medium-sized motion sequences, each containing one or several semantic actions. Processing of long motions and motion streams is usually studied on semi-artificial data constructed from the shorter recordings.

III. SEGMENTATION OF CONTINUOUS SKELETON SEQUENCES

For efficient data organization, it is often necessary to partition long skeleton sequences into reasonably-sized segments. In some situations, human experts can be asked to manually pre-process the sequences, i.e., mark the precise positions of all actions and assign their labels. However, this is not feasible in most cases; therefore, various automated segmentation policies have been developed. The segmentation policies can be utilized in a *virtual mode* when the acquired segments are only used temporarily and discarded afterward, or in a *physical mode* when the segments are kept to be accessed repeatedly. The virtual mode is typically used for stream filtering or query segmentation, whereas the physical mode is employed for pre-processing of long motions in content-based search scenarios. In the following, we review three typical segmentation policies illustrated in Figure 2.

A. FIXED-SIZE SEGMENTATION

The most straightforward segmentation is realized by a mechanical slicing of the motion sequence into non-overlapping fixed-size segments. There is no generally accepted optimal size of segments, but the rule of thumb suggests that the segment length should be upper-bounded by the length of the shortest retrievable action. The crucial problem of this approach is that semantically coherent parts can be divided by artificial cuts. The placement of these cuts in a given semantic action is determined by the precise temporal position of this action within the long motion sequence, so the segmentation of two identical actions may differ if they appear in various parts of the containing sequence.

B. OVERLAPPING AND HIERARCHICAL SEGMENTS

In real applications, the fixed-sized segments are mostly implemented as overlapping, which suppresses the problems of mechanical cutting at the price of increasing data redundancy. There are two basic strategies for constructing the overlapping segments: either we use same-sized segments and only shift their beginnings, or a hierarchical segmentation is applied with different sizes of segments on individual levels. The same-sized segments appear, for example, in [18]–[20] with the recommended overlap values ranging from 50 % to 80 %. The hierarchical segmentation is often used to ease matching actions performed at different speeds [21] and is typically implemented by the Temporal pyramids [22], [23] that model multiple different temporal scales simultaneously.

C. SEMANTIC SEGMENTATION

The objective of semantic segmentation is to produce non-overlapping segments of variable sizes that correspond to the semantic actions contained in the sequence. The discovery of the semantic segment boundaries can be based on prior knowledge that exploits pre-learned motion characteristics from known training data [24]–[26]. No-prior-knowledge solutions are based on significant changes in intrinsic dimensionality [27], discovery of repeating patterns [28], or significant accumulation of specific feature characteristics [29]. The category-blind semantic segmentation typically combines unsupervised feature learning with data mining to learn frequent motion patterns [30]–[32]. Backward high-confidence discovery of such patterns then determines the final segmentation.

D. STRENGTHS AND WEAKNESSES

Segmentation is vital for all motion processing tasks where the input data are not provided as individual actions. The semantic segmentation is an ideal solution that would reduce most of the motion processing problems to text-like searching–we could simply cut the data into actions, label them, and do all the processing over the labels. However, it is generally believed that a reliable semantic segmentation is possible only for a limited range of motion processing problems. In particular, high-quality semantic detectors can be trained for simple and well-understood motions such as gait cycles, or actions with many training samples. The training process is often costly, but the actual searching over the segmented data is very efficient. On the other hand, the main advantages of the fixed-size segmentation are its simplicity, minimum construction costs, and wide applicability. However, it is not clear how to determine a suitable segment size, and the simple fixed-size segmentation cannot correctly deal with motions that are slightly shifted in time.

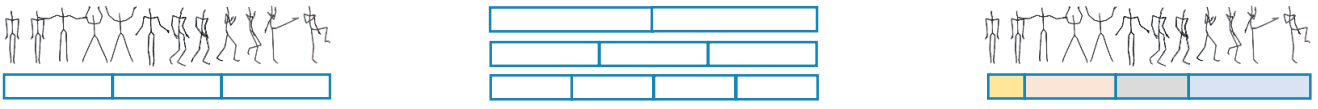


FIGURE 2. Types of skeleton data segmentation.

Overlapping fixed-size segments allow better alignment of two compared motions but require additional storage and/or processing costs, which limits the processing scalability. A possible solution is to apply a hierarchical segmentation in combination with approximate retrieval techniques. Alternatively, we may accept the lower precision of unsupervised semantic segmentation and use it to obtain a non-overlapping segmentation, trading the lower precision and high segmentation costs for increased efficiency and scalability of the query evaluation.

IV. SIMILARITY METRIC LEARNING

A fundamental prerequisite for content-based motion data management is an ability to determine the *similarity* between two skeleton sequences. There are several levels on which the similarity may be evaluated: individual poses, short artificial segments, actions, and sequences of segments. For each of these situations, there exists a variety of *features* that can be used to represent the skeleton data, and associated distance functions that evaluate the similarity of two features. Together, the feature space and the distance function form the *similarity metric*. In the following, we review the state-of-the-art feature extraction methods together with their corresponding distance functions and provide a short overview of feature augmentation and transformation methods.

A. FROM HANDCRAFTED FEATURES TO DEEP ONES

Raw skeleton data consist of poses represented by vectors of absolute [41] or relative [42] joint coordinates. Individual poses can be numerically compared for similarity using standard vector similarity measures, i.e, the Manhattan or Euclidean distance. The similarity between any two sequences of poses can be measured by time-warping functions, such as the Dynamic Time Warping [33] or Longest Common Subsequence [43]. However, the raw skeleton data are quite bulky and their processing is computationally demanding, especially in combination with the quadratic complexity of the time-warping function. Therefore, the raw skeleton data are rarely used except for baseline evaluations. In the early years of motion processing, *handcrafted* pose features, such as joint angle rotations [34], [44] or relationships between selected pairs of joints [25], [45], were used to reduce the data volume. These were again compared by linear-time distance functions on the level of poses, and quadratic-time DTW on the level of pose sequences. However, the handcrafted features have to be designed by domain experts and have limited ability to represent more complex dependencies in movement patterns. Therefore, the handcrafted features have been practically abandoned and replaced by *deep features* extracted from well-trained neural-network models [5].

Deep neural networks are often used for classification of actions into a predefined set of classes. The learned parameters of hidden network layers can then be utilized to extract content-preserving features from input actions or segments. Such features are typically represented as fixed-size high-dimensional vectors (e.g., 4,096D features in [36]) and generalize very well when varied training data are provided. Contrary to the handcrafted features, the deep features have higher descriptive power, and, importantly, their fixed-size nature enables efficient and indexable comparison of whole segments/actions, e.g., by the Manhattan, Euclidean, or Hamming distance functions.

B. DEEP FEATURE LEARNING WITH LABELED DATA

State-of-the-art supervised deep learning is based on convolutional neural networks, graph convolutional networks, or recurrent neural networks. These three architectures (or their fusion) currently give the ten best classification results on the popular NTU-RGB+D dataset [46]. We provide a brief survey of these architectures; for a more detailed discussion, we refer the readers to the recent action-recognition survey [4].

1) RECURRENT NEURAL NETWORKS (RNN)

The recursive connection inside recurrent networks well suits the sequential nature of motion data. Individual skeletons, represented as vectors of joint coordinates or their derived features, are gradually fed to RNN cells, and the output of a previous time step is passed to the input of the current step [47]. Most attempts suggest employing the Long Short-Term Memory (LSTM) variant of RNN cells to better learn long-term temporal dependencies [35], [48], [49] and avoid the vanishing gradient problem. However, RNNs generally suffer from the inability to model spatial dimensions explicitly. The spatial modeling can be improved by exchanging the pose and temporal axes [50] or by adding global context-aware attention that selectively focuses on the most informative joints [48]. RNNs enable users to specify the output feature size (i.e., the hidden state size) to control the trade-off between efficiency and descriptive power.

2) CONVOLUTIONAL NEURAL NETWORKS (CNN)

In contrast to RNNs, convolutional networks possess a great ability to learn high-level spatial characteristics. However, they assume input data in the form of matrices and do not explicitly consider a temporal dimension. Both the issues are typically solved by encoding 3D skeleton sequences into so-called *motion images*, where rows correspond to the joints, columns to the time dimension, and the RGB colors to the 3D joint coordinates [36], [51], [52] or joint dynamics [53], [54]. The motion images can be combined with models pre-trained on ordinary photographs, such as the AlexNet in [36] or Inception-v3 in [55], to achieve higher descriptive power of extracted features [36], [54], [55]. The size of output features depends on the CNN architecture but usually ranges between 512 and 4,096 dimensions, e.g., 4,096D features in AlexNet and 2,048D features in Inception-v3. A general disadvantage of CNNs is that they often consider only neighboring joints in convolutional kernels, which tends to learn local co-occurrence characteristics rather than some latent correlation possibly appearing among all the joints.

3) GRAPH CONVOLUTIONAL NETWORKS (GCN)

Since the human skeleton is naturally characterized by a graph where vertices correspond to human joints and edges to bones [37], [56], there is a trend to utilize GCNs, a generalization of CNNs working with graphs of arbitrary structures. In [37], a spatio-temporal graph convolutional network (ST-GCN) is constructed by adding temporal edges that connect the same joints across consecutive poses. Such representation can automatically capture the patterns embedded in the spatial configuration of the joints and their temporal dynamics, which leads to higher expressive power and better generalization capability compared with CNNs. However, a fixed-structure graph that models only the physically-connected joints ignores the dependencies between distant joints that are not connected. Therefore, some recent works [57]–[59] try to learn the relationships between distant joints automatically.

4) FUSION METHODS

To enhance the model accuracy, different kinds of neural networks or data modalities can be fused. Some papers propose to learn spatial configurations and temporal dynamics within two independent LSTM [50] or CNN [53], [55] streams, whose features from the last pooling layer [55] or directly softmax scores [50], [53] are finally fused. The fusion approach in [60] proposes to learn spatial features of individual 3D skeletons using CNN and then train an LSTM network on top of such features. In [61], multi-modal features are first extracted from the input actions and then fused by an autoencoder network. In [62], the authors propose to fuse the RGB and 3D skeleton modalities.

C. DEEP FEATURE LEARNING WITH UNLABELED DATA

Models based on RNNs, CNNs, or GCNs are powerful but require labeled actions for supervised training. However, there are scenarios where no semantic labeling is available in advance, and unsupervised training is the only possibility. In such cases, *Siamese* or *triplet-loss* networks can be trained to learn the similarity between unlabeled motion segments, using examples of similar and dissimilar segment pairs. To find suitable segment pairs for training, it is necessary to use additional domain-expert knowledge or some simple metric (e.g., based on comparison of handcrafted features [19]) that can at least roughly estimate the low-level segment similarity.

Siamese networks are characterized by two identical sub-networks that learn a mapping from the input segment space to an embedding space [63]. The networks employ a contrastive loss function that minimizes the distance between the embeddings of two similar segments and maximizes the distance between embeddings of dissimilar ones. Nevertheless, the contrastive loss forces all similar segments to be close, while the dissimilar ones are separated by a certain fixed distance [19]. This restriction is suppressed using the triplet loss function [64] that only requires dissimilar segments to be farther away than any similar segment on a per-example basis. Such a triplet loss approach is integrated within a CNN architecture in [19], [51].

D. DATA AUGMENTATION

Independent of a specific neural-network architecture, the size and variability of the training dataset significantly influence the quality of the resulting model. Large and rich datasets improve model generalization and reduce the risk of overfitting. However, available training datasets are often limited in size, especially for supervised learning. *Augmentation* is a way of automatically enlarging and enriching the training skeleton data using various transformation techniques. For example, image cropping transformation is used in [65] to generate random patterns in 2D motion images. Different normalization techniques, such as skeleton rotation and scaling, are used to generate additional 3D skeleton segment samples [50]. In [35], both spatial and temporal dimensions are modified by adding noise into joint coordinates, or by cropping and extending the original content of segments. An advanced approach in [66] trains a generative adversarial network to emphasize the differences between actions with very similar gestures.

E. FEATURE TRANSFORMATION

Both the handcrafted and deep features are often very high-dimensional (e.g., a 4,096-dimensional deep feature used in [36]), which is not convenient for large-scale data management. Therefore, various feature transformation methods are used to produce more compact data representations. A wide range of general-purpose dimensionality reduction techniques can be applied to motion features [67] to enable more efficient processing by metric- or vector-space indexes [36].

Most of the features discussed so far are primarily designed for actions, i.e., semantically meaningful pieces of motions that are compared as a whole. The same type of features can also be used for unsegmented data, when the artificial segments are created so that their sizes are similar to actions. However, there are also alternative approaches that cut the unsegmented data into a higher number of short overlapping segments, represent these by very compact features, and then compare sequences of such features. To obtain the compact features, the short segments are first represented by arbitrary high-dimensional features (e.g., raw skeleton data in [39] or deep features in [19]), then the space of the segment features is clustered, and cluster identifiers are used to form a vocabulary (codebook). Individual segments are then represented by the one-dimensional identifiers of the closest cluster, so the similarity of two short segments is reduced to a trivial equality over the *quantized features*. The skeleton sequences can be then represented by sequences [39], histograms [19], [68], [69] or bags [40] of the quantized features. The bag-of-words representation proposed in [40] is mainly interesting from the large-scale processing perspective, since it enables applica-

TABLE 2. Most common approaches for determining similarity of poses, actions, segments, or sequences of segments.

Level	Features			Comparison function		
	Type	Representation	Size (dim.)	Type	Complexity[1)]	Indexability
pose	raw skeleton data	joint coordinates [33]	$\approx 10^2$ D	Hausdorff	$\mathcal{O}(n^2)$	low
	handcrafted	joint angles [34], relational [25]	$\approx 10^1$–10^2 D	L_1, L_2	$\mathcal{O}(n)$	high
action	handcrafted	relational [25]	$\lvert seq\rvert \cdot 40$ D	DTW	$\mathcal{O}(n^2)$	low
	handcrafted	histograms [33]	$\approx 10^3$ D	Bhattacharyya	$\mathcal{O}(n)$	low
	deep	RNN [35], CNN [36], GCN [37]	$\approx 10^2$–10^3 D	L_1, L_2	$\mathcal{O}(n)$	medium
segment	deep	autoencoder [38]	160 bits	Hamming	$\mathcal{O}(n)$	high
	quantized	motion words [39]	1 D	L_1	$\mathcal{O}(n)$	high
	deep + quantized	motifs [19]	1 D	L_1	$\mathcal{O}(n)$	high
sequence of segments	deep	autoencoder [38]	$\lvert seq\rvert \cdot 160$ bits	DTW	$\mathcal{O}(n^2)$	low
	quantized	bag of motion words [40]	$\lvert BoW\rvert$ D	Cosine	$\mathcal{O}(n)$	medium
	deep + quantized	signatures [19]	$\lvert BoW\rvert$ D	EMD	$\mathcal{O}(n^3 \cdot \log n)$	low

[1)] The time complexity is expressed in relation to the input length, i.e., the size of the respective features. In many cases, the feature size is constant, so the similarity evaluation times are constant as well. However, there may be significant differences in the actual times needed for the similarity computations.

tion of efficient and scalable text-retrieval techniques (e.g., inverted files) for a variety of motion processing tasks.

F. STRENGTHS AND WEAKNESSES

Handcrafted features can be easily extracted from raw skeleton sequences but their descriptiveness is limited. While deep features are generally considered as more effective, they require high-quality training data, a time-consuming training process, and non-negligible costs needed for feature extraction. Widely-used LSTM networks are convenient for modeling the temporal dimension of skeleton data but fail in learning dependencies in the spatial domain. On the other hand, CNNs are successful in learning local spatial characteristics but hardly learn some latent correlation related to all the joints and further require the fixed-size input, which leads to deformation of at least the temporal dimension of skeleton sequences. Modeling motions in the form of a graph in combination with RNNs or CNNs seems to be effective for learning spatio-temporal dependencies, however, a suitable transformation of skeleton data into some content-preserving graph-like representation is still challenging.

The simplest way for deep-feature learning is to train a neural network on the classification task. However, this approach requires labeled training samples known in advance. On the other hand, the triplet-loss learning approach does not require data labeling but the preparation of training triplets is time-consuming and semantically difficult. Moreover, such process usually requires larger amounts of training data, which naturally leads also to a time-consuming training process compared to common action classifiers.

From the efficiency point of view, the fixed-size deep features allow much faster processing than the variably-sized raw skeleton data or handcrafted features. The variably-sized features are typically compared by time-warping functions such as DTW, which are computationally expensive and difficult to index. On the other hand, the deep features are usually compared by efficient L_1, L_2 or Hamming distance functions that enable a straightforward application of multi-dimensional index structures. Still, the deep features are typically high-dimensional, which is not optimal for efficient indexing. Different transformations into more compact representations improve the indexability of features but decrease their descriptiveness. A high-level summary of feature types along with their comparison functions and suitability for indexing is available in Table 2.

V. MOTION SEARCHING AND FILTERING

Searching and filtering are fundamental data-processing operations that aim to answer users' *queries* over motion data collections and streams. In the *search paradigm*, the data are provided as a collection of segments or long motion sequences, and the query can be an arbitrary short motion. In our model domain of figure skating, this can be exemplified by searching a database of figure skating performances for motions similar to a given spin example. In the *filtering paradigm*, the data are represented by known exemplars of to-be-detected events, whereas the queries are formed by virtual segments temporarily extracted from continuous motion streams. A typical example would be annotating a life broadcast of a figure skating competition, using a database of common figure-skating components. In both cases, the retrieval algorithms can hardly make any assumptions about the query, but the data can be arbitrarily pre-processed (e.g., segmented, clustered, or indexed) to enable efficient query matching. Both operations including the pre-processing pipeline are illustrated in Figure 3; the searching task is especially difficult when the data consist of long unsegmented motions, whereas the filtering task is hard in stream environments that require real-time responses.

A. SEARCHING

In a query-by-example search, users first need to specify a motion query. This can be simply selected from available skeleton sequences, drawn in visualization-driven graphical user interfaces [77], [80], physically modeled by puppet interfaces [81], programmed as a set of logical constraints [70], [74], [77], or artificially synthesized from different body parts acquired from multiple distinct motions [76]. The constructed query is then compared to the data, which need to be prepared so that segments of comparable length are matched.

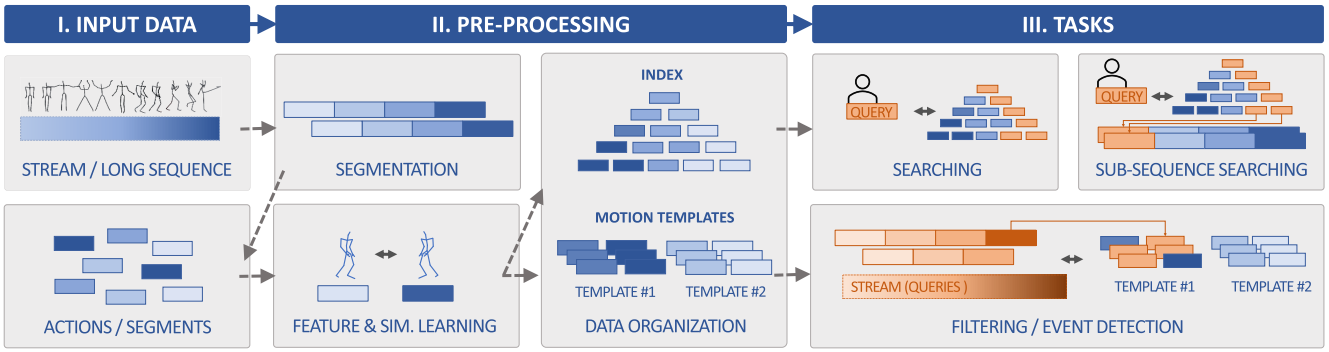


FIGURE 3. Illustration of data management processing pipeline. Continuous input skeleton sequences need to be partitioned into short segments. Based on labeled known actions or unlabeled segments, supervised or unsupervised methods are used to learn the similarity metric. The data are organized into index structures to facilitate large-scale search, or into motion templates to facilitate real-time annotation.

TABLE 3. Methods for searching in motion collections.

	Method	Query repr.[1)]	Features	Data replication[2)]	Query expansion	Pre-search	Refinement	Efficiency QRT	Efficiency DB size
Search	[70] *(2009)*	3D, LR	SOM indices	○	–	binary tree	Smith-Water.	240 ms	≈ 2.5 h
	[71] *(2010)*	3D	3D coordinates	○	✓	kd tree	graph traversal	180 ms	12 h
	[72] *(2012)*	HDS	SH functions	○	–	–	L_2	18 ms	2.5 h
	[38] *(2015)*	3D	20B signatures	○	–	–	Hamming	512 ms	4 h
	[36] *(2018)*	3D	deep features	○	–	–	L_2	11 ms	1.5 h
	[39] *(2020)*	3D	motion words	◐	–	–	DTW	13 ms	1.5 h
	[73] *(2020)*	video	deep features	◐	–	–	–	56 ms	≈ 0.5 h
Sub-sequence search	[74] *(2005)*	3D + LR	geom. relations	○	–	linear scan	DTW	294 ms	3 h
	[75] *(2006)*	3D	geom. relations	○	–	–	DTW	10^4 ms	0.5 h
	[76] *(2009)*	3D	motion patterns	○	✓	–	body-part fusion	72 ms	4 h
	[43] *(2011)*	3D	joint rotations	○	–	–	LCS	10^5 ms	9 h
	[44] *(2013)*	3D	joint rotations	○	✓	M-index	temporal filter	10^3 ms	1 h
	[77] *(2013)*	3D, LR, UI	geom. relations	○	✓	trie	temporal filter	40 ms	35 h
	[78] *(2017)*	3D	deep features	◐	✓	linear scan	Euclidean	10^3 ms	1 h
	[19] *(2018)*	3D	deep motifs	◐	–	–	EMD	10^{3-4} ms	0.25 h
	[21] *(2019)*	3D	deep features	●	–	–	L_2	84 ms	1 h
	[79] *(2019)*	3D + T	3D coordinates	○	–	text search	PageRank	40 ms	9 h

[1)] 3D – skeleton sequence, LR – logical rules, HDS – hand-drawn sketch, T – text keyword, UI – skeletons drawn via a user interface
[2)] Data replication – none (○), overlapping data segments (◐), overlapping data segments in multiple levels (●)

This is implicitly satisfied when the data collection consists of pre-segmented actions [36], [38], which leads to the standard *search* task. However, many databases contain long motion sequences [19], [77], which naturally results in the *sub-sequence search* task.

The retrieval process of both search tasks can be divided into two steps: *pre-search* and *refinement*. In the pre-search step, a set of query-relevant *candidate* results is efficiently retrieved, e.g., using various index structures [82], such as the binary tree [70], kd tree [71], or tries [77]. In the refinement step, the retrieved candidates are re-ranked by more expensive techniques (e.g., traversal of a graph structure [71] or ranking by DTW [74]) to determine the final results. When the pre-search step is not supported [19], [38], [75], the refinement is evaluated over the whole data collection.

One of the main retrieval issues, mostly occurring in the sub-sequence search task, is to find the accurate alignment of an arbitrary query within a data sequence. This can be solved by expensive matching on the level of individual poses [44], [71], or by partitioning either the query [77] or data [21] motions into *overlapping* segments. In particular, unsegmented queries are typically combined with an overlapping and hierarchical segmentation of the data, where the segment sizes on individual levels correspond to expected query sizes [21]. Alternatively, both the query and data can be partitioned into short segments to support the evaluation of variable-length queries. However, the retrieval phase is more difficult as a sequence of multiple query segments has to be located within the sequence of many data segments using temporal filters [44], [77] or expensive warping functions, such as DTW [75], Longest Common Subsequence (LCS) [43], Earth-mover's distance (EMD) [19], or Smith-Waterman algorithm [70]. While the overlapping data segments increase space requirements due to *data replication*, the *query expansion* into multiple segments increases query response times due to the necessity of evaluating multiple sub-queries.

In Table 3, we provide a comparative summary of standard and sub-sequence search methods from several perspectives: the volume of replication of data segments, the expansion of query leading to several sub-queries that need to be separately evaluated, the existence of the pre-search step, and the way

TABLE 4. Methods for motion stream filtering.

Method / Temporal mechanism	Skeleton	Type	Early detection	Prediction	FPS speed
DTW + motion templates [25] *(2009)*	3D	segment	–	–	240
Naive Bayes + Riemannian manifold [83] *(2017)*	3D+depth map	segment	–	–	7
kNN classifier + CNN features [20] *(2017)*	3D	segment	–	–	131
Sparse group lasso + directional features [84] *(2018)*	3D	segment	–	–	N/A
Curvilinear seg. + fusion classifier [29] *(2018)*	3D	segment	✓	–	667
LSTM + sliding window features [26] *(2019)*	2D	segment	–	–	5.4
kNN classifier + Moving pose [85] *(2013)*	3D	frame	✓	–	N/A
Linear regression + SSS features [86] *(2013)*	3D	frame	✓	–	500
SVM + temporal pyramids [22] *(2015)*	3D	frame	–	–	380
Linear search + BoG + sliding window [69] *(2016)*	3D	frame	✓	✓	93
Classification-regression LSTM [87] *(2016)*	3D	frame	✓	✓	1,230
Linear search + bag of gestures (BoG) [88] *(2018)*	3D	frame	✓	–	N/A
Attention-based LSTM [89] *(2018)*	3D	frame	✓	–	N/A
Bi-Directional LSTM [90] *(2019)*	3D	frame	✓	✓	7,700

of evaluation of the refinement step. We also provide the *query response time* (QRT) that shows the actual time to answer a single query. The individual QRTs are taken from individual papers and cannot be directly comparable, as they significantly depend on the database size (DB) and other factors, such as the frame-per-second rate, hardware, feature selection, length of the query, and the number of retrieved results (e.g., the value of k in k-nearest neighbor queries). Noticeably, the current methods only work with dozens of hours of motion data or less, so they may not be sufficient for future large-scale retrieval applications.

B. STREAM FILTERING

Time-critical environments, such as security analysis of surveillance cameras, require continuous online processing of the streaming skeleton data, only with knowledge of the very recent past and without any assumptions on the future. The most common operation is *event detection*, a supervised online filtering task that annotates events in ongoing streams by determining their precise beginnings and endings and recognizing their types. Each type of event to be recognized is specified by a set of known action examples that can be pre-processed in advance. Such pre-processing includes extraction of deep features of individual actions [20] or aggregation of the same-class actions into motion templates [25], [88]. The pre-processed actions or templates are then used to detect the desired events in the streaming input either on the level of virtual segments or individual frames (i.e., poses).

Segment-level detectors model the temporal context by partitioning the stream into overlapping segments mechanically obtained by a sliding window principle [25], [69], [84], or into disjoint semantic segments [26], [29], [83]. The segments are then directly classified (e.g., using Naive Bayes [83]) or matched against the pre-processed actions or templates using various distance functions, such as the Dynamic Time Warping in [25], Euclidean distance in [20], or fusion of linear classifiers in [29]. The event is finally detected if the distance satisfies some predefined threshold.

Frame-level detectors [22], [85]–[90] typically train various models on the provided actions to estimate a class-relevance probability for each frame of the stream. These probabilities are estimated based on LSTM networks [87], [89], [90], Support Vector Machines [22], or linear regression classifiers [86]. To deal with the neighboring context of individual frames, the recent past is encoded within enriched frame features (e.g., Moving Pose [85] or Structured Streaming Skeleton [86]) or within the memory of hidden states of LSTM networks (e.g., a whole attention module is dedicated to learning temporal evolution in [89]). Noticeably, the frame-level approach can reveal events before they finish [91] (i.e., early detection), or even predict future ones [92], [93].

The state-of-the-art frame- and segment-level event detection methods are summarized in Table 4. As a final remark, let us observe that a general disadvantage of all classification models is that they need to be completely retrained whenever a new type of event is introduced.

C. STRENGTHS AND WEAKNESSES

Searching, sub-sequence searching, and filtering are data-intensive operations that combine similarity matching, data organization, and temporal segmentation issues. Processing motion data in the form of raw skeleton sequences or high-dimensional features is computationally inefficient. Therefore, feature-extraction and dimensionality-reduction techniques aim at achieving a high descriptiveness-compactness trade-off. Compact representations reduce the processing costs dramatically but introduce the risk of over-simplification that leads to the deterioration of distinctiveness between originally dissimilar motions. For this reason, compact features (e.g., motion words [39], motifs [19], or signatures [38]) are more suitable for the pre-search phase, whereas the high-dimensional features (e.g., deep features [36] or raw 3D coordinates [44]) for the more expensive refinement phase.

Even very compact features can become a performance bottleneck when particularly large data volumes are accessed frequently by sequential processing. Feature indexing with respect to the used similarity matching function enables sub-linear processing costs and thus ensures a reasonable degree of scalability. As observed in Table 3, the solutions [19], [43], [74], [78] that combine complex features and linear search strategy report poor performance

already on small (∼1-hour) datasets. On the other hand, index-based approximate retrieval strategies claim to be able to efficiently search in data volumes that correspond to weeks of motion recordings [21].

To achieve the findability of short queries within long data sequences, the data sequences need to be systematically partitioned into a multitude of segments. Particularly, the type of query segmentation determines the pre- and post-processing costs. A single-segment query implies the need for dense, usually overlapping, segmentation of the data sequences to increase the chances of the accurate query-to-segment temporal alignment. Such dense data segmentation is associated with high space complexity [21] and requires efficient retrieval algorithms. On the other hand, densely segmented queries can be matched against sparsely segmented data sequences, which dramatically reduces the space requirements but introduces non-trivial post-processing costs to filter out the candidate results that do not follow the temporal order of the evaluated sub-queries.

While retrieval-based solutions offer a high degree of universality and scalability especially in searching tasks, end-to-end deep learning-based methods excel at domain-specific action filtering. Particularly, the recurrent LSTM networks are suitable not only for learning highly-descriptive features, but they also demonstrate superior throughput by performing multi-label frame-level annotation at constant processing costs. On the other hand, examples of actions of interest need to be a priori available to supervise the network training, which makes this approach less suitable for scenarios where the actions of interest (i) tend to change dynamically, (ii) cannot be obtained in advance, or (iii) are represented by an insufficient number of examples.

VI. CURRENT APPLICATIONS

Content-based motion matching, searching, and filtering are important operations that already find applicability in many domains. However, due to high data acquisition costs, caused mainly by the need of human experts (actors) and expensive infrastructure, the majority of existing applications is domain-specific and only uses limited volumes of specialized data. Such situations do not explicitly require scalable management solutions and mainly concentrate on various data analyses. Though exceptions exist, current skeleton data applications typically involve analysis and recognition of pre-segmented actions based on well-annotated training data, detection of a small number of events, or content-based retrieval in collections of hundreds, maximally thousands of actions. These applications can be seen as small-scale prototypes that open different areas of motion processing. In the following, we give a concise summary of selected representative application domains.

A. ENTERTAINMENT

Motion capture initially emerged in computer animation to render realistic-looking movements in movies and games. This involves direct-mapping of captured movements from live subjects to virtual characters (motion re-targeting), and deep-learning-based generative models that synthesize sequences of actions with authentic and fluent movement transitions between actions [94], [95]. Requirements for interactive and visual-based browsing in large motion collections [80] also emerged in the movie industry to increase the re-usability [96] of the expensively captured data. Finally, motion data appear in virtual and augmented reality applications and games, where real-time detection and analysis of user actions are needed for the immersive experience [97].

B. HEALTH CARE

Motion capture technology has become a new medical tool that assists doctors and therapists during the diagnosis and treatment of their patients. Gait analysis helps determine a neurodegenerative disease [98], evaluate different treatment outcomes for cerebral palsy [99], or identify individualized therapeutic strategies for running injuries [100]. The skeleton data are also used for motion monitoring and injury prevention [101]. Rehabilitation monitoring systems assist patients during recovery [102] and increase their engagement via gamification [103].

C. SPORTS

In professional sporting, motion processing applications typically focus on posterior analysis and evaluation of athletic performances, e.g., in dancing [104], martial arts [13], golf [105], diving, or figure-skating [106]. Real-time and detection applications involve prediction of the future tennis-shot direction [107], detection of swimming strokes [108], or stride-, jump- and landing phases of a long and triple jump [109]. Motion data analysis also assists people in learning dancing [110] or possibly any moves according to a projected performance [111].

D. SMART CITIES

Real-time sensors and video-based motion data can be also used to analyze situations in crowded spaces (e.g., pedestrian zones, shopping malls, stadiums), smart homes, or autonomous driving vehicles. Open-space applications focus on public safety (e.g., identification of subjects by posture and gait [11] in the prevention of organized crime), customer analysis and shopping support [112], and social interaction understanding [113]. Applications at smart homes involve the detection of abnormal movement patterns [114]. In autonomous driving vehicles, skeleton data can be utilized in better subject tracking [115] and movement prediction of pedestrians and cyclists [116].

VII. CHALLENGES

The research of the last decade has established many fundamental techniques for motion data matching, searching, and filtering. However, almost all existing solutions work with motion data in the form of small and precise single-person skeleton sequences. With the arrival of hardware and software

tools that allow large-scale motion data acquisition, it is essential to review the existing techniques from the perspective of scalability and identify new directions for future research.

Current trends suggest that massive volumes of skeleton data will soon be available either from videos uploaded and freely available on the web or via low-cost sensors or ordinary cameras. Apart from being voluminous, such motion data are likely to be imprecise due to the low accuracy of the capturing devices, reduced frequency of frame rates, or occlusions. At the same time, the video-based data will often contain multiple, possibly interacting, entities (e.g., individuals and groups). In general, the expected shift in research focus is from *a single-person, uni-modal, small, and precise* data collections to *groups of people, huge, dirty, and multi-modal* datasets.

In the following, we discuss two types of challenges brought by the changing motion data. First, we focus on the applicability of existing techniques to the massively-produced data. Then, we take a step beyond the established areas of motion searching and filtering and outline new possibilities for analyzing the human motion data.

A. SCALABILITY ISSUES

When the motion data acquisition becomes easily available, we can expect significant growth of both the *volumes* of the data and the *diversity* of application domains. Consequently, the techniques applied in all phases of motion data processing should be scalable in both of these directions.

1) DATA PRE-PROCESSING

The extraction of skeleton sequences from ordinary videos is likely to produce datasets of uncertain quality that will need to be cleaned and enhanced. The state-of-the-art literature offers some works on motion data cleaning, but these mostly focus on correcting small errors in marker-based motion capture data using statistical methods that are not applicable to highly erroneous video-based skeleton data [117]. Therefore, alternative approaches need to be studied. A promising direction is to enhance the imprecise skeleton sequences by *additional modalities* such as colors, faces, or context in general, which can also be extracted from the video [6].

Automatic harvesting of skeleton data from web videos also brings the need to detect duplicate and near-duplicate motion sequences. For this purpose, the *similarity join* operator [118], which computes all pairs of motions within a certain similarity threshold, should be adopted from general content-based retrieval to the motion processing domain.

2) SIMILARITY METRIC LEARNING

The majority of state-of-the-art metric learning approaches are based on supervised learning, working with a rather low number of application-specific motion classes for which high-quality training data exist. The scalability of these techniques to new domains and larger datasets is limited by the ability of the machine learning techniques to deal with a growing number of classes, which has not been much studied yet, and the availability of the training data. We believe that two important research directions should be pursued in this area. First, new *reference collections* of clean, precise, and labeled data should be created to be used for supervised similarity metric learning as well as for testing and evaluations. In contrast to current training data, which are mostly created manually, the future reference collections could be built in a crowdsourcing manner, e.g., using relevance feedback, crowdsourcing, or gamification [119]. Second, a *multi-modal processing* should be employed for the similarity metric learning to help distinguish among the growing number of classes [62]. The utilization of orthogonal modalities should be especially useful in situations when reliable training data are not available and unsupervised learning has to be applied.

To support efficient large-scale retrieval, it is also vital that the learned motion features and distance functions can be efficiently indexed. State-of-the-art deep features are typically high-dimensional vectors, which are known to be difficult to index due to the curse of dimensionality [120]. Therefore, researchers should continue to look for *indexable motion features* that provide a reasonable trade-off between feature expressiveness and its complexity [39]. The indexable features would be used for fast identification of candidate motions, which could be then refined using more precise and costly similarity evaluations.

3) SEARCHING AND FILTERING

To accommodate the growing amounts of data, motion retrieval methods need to be supported by *robust data-processing techniques* that optimize the query response times and the throughput of simultaneously processed queries [82]. Although a number of existing solutions employ some form of indexing, no comparative studies have been conducted to assess the efficiency and scalability of the proposed methods. There is also little work on using *approximate retrieval strategies* to limit the number of database objects that need to be accessed [121]. In both these aspects, the motion processing—initially cultivated in the computer vision community—could benefit from a closer cooperation with the database, information retrieval, multimedia, and data mining communities. Finally, it is important to establish *large-scale evaluation datasets* to allow fair comparison of various data management strategies.

B. NEW MOTION PROCESSING TASKS

The increased availability of motion data will very likely inspire new types of applications in both traditional and new data domains, which will introduce new types of motion processing operations. In the following, we briefly comment on possible extensions of the single-person motion processing, and introduce the challenging area of analyzing group activities.

1) COMPLEX QUERIES OVER SINGLE-SKELETON SEQUENCES

State-of-the-art motion processing focuses strongly on the analysis and retrieval of short motion sequences, i.e., segments or actions. However, we can easily find real-world scenarios where more complex motion objects and their relationships need to be analyzed. Considering the figure-skating domain again, we might be interested in comparing and searching whole skating performances instead of individual skating elements. While the skating elements are typical examples of motion actions, the whole performance is different – it is formed by a sequence of actions that comprise a single real-world semantic unit. This type of motion data is denoted as *motion episodes* in [40]. The episodes may be targeted by structured queries that expand the query-by-example paradigm by additional requirements on episode primitives or their sequentiality; for instance, we could be interested in all performances that contain at least three triple-jumps within a given time window. Such queries cannot be solved by existing motion search methods and their incorporation requires a fundamental re-thinking of the motion data management. To support systems of the *publish-subscribe* type, the new operations will also need to be implemented on streams.

2) ANALYSIS OF GROUPS

Current motion processing focuses mainly on single-subject movements, but in real world people frequently interact and form groups or even crowds. Understanding and modeling groups of people and crowd behavior is a critical problem in many domains, including human-computer interaction, smart cities, psychology, and behavior learning. Some of the driving applications include investigation of pathological processes in mental disorders, virtual reality therapy, training of law enforcement officials or military personnel, urban layout design, or intelligent crowd management. In all these areas, a precise modeling of individuals and their interactions is highly desirable.

Research in social psychology reveals that the groups can be identified based on their *entitativity* [122] – a level of perception of a group as a single entity defined by the similarity, cohesiveness, and uniformity of its members. For example, a football player team in matching dresses is highly entitative compared to people waiting at a tram stop. Though the concepts of group entitativity are broad (e.g., appearance, social background, common fate), movement synchrony and motion-based similarity play the central role [123]. A critical challenge lies in understanding how the individual motions combine into the group-level entitative behavior. It is generally agreed that the main motion entitative factors are attractive and repulsive forces (e.g., movement speed and directions) and physical interaction between individuals. The semantic unit at the core of any group interaction is a *motion dyad*, i.e., a movement rhythm of a pair engaged in social interaction (e.g., dancing, fighting, holding hand), often displaying signs of coordination. Skeleton-based identification

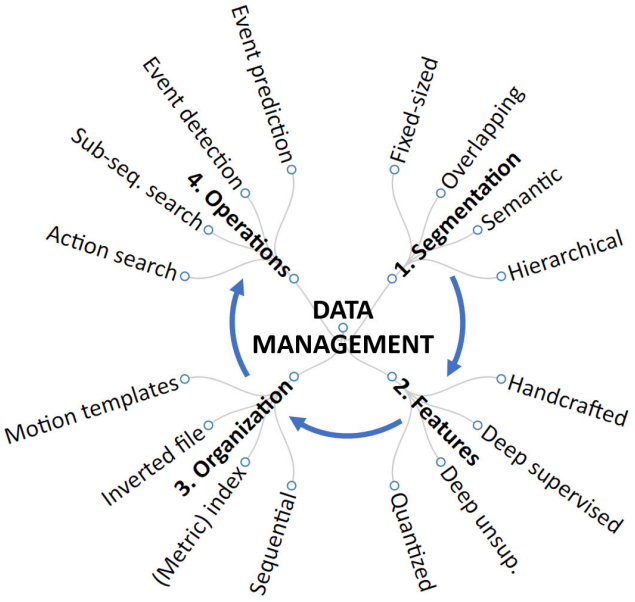


FIGURE 4. Taxonomy dendrogram illustrating the key motion processing topics surveyed in this work.

of dyads and their efficient processing is thus the first step towards more advanced group modeling.

A related but distinct area of psychological research is the *crowd behavior* analysis [124]. It assumes each individual to be a self-organized object, a collection of which (a crowd) can demonstrate an emergent behavior with a specific objective. What appears to make crowds unique is their ability to act as a united mass in a socially coherent manner without any prior awareness. In this respect, the skeleton models of human individuals can much improve crowd identification.

VIII. CONCLUSION

This survey provides a systematic overview of the state-of-the-art techniques for similarity-based management of human motion data, which can provide incentives and guidelines to other researchers as well as potential users developing their own applications. We discuss the topics of skeleton data representation, segmentation, similarity modeling, and data organization, upon which rest the widely-applicable operations of similarity-based searching and filtering. A high-level taxonomy of the surveyed directions is provided in Figure 4. Since a significant change in the quantity and quality of motion data is expected in the near future, we further offer a scalability-oriented review of the limitations of existing solutions, and suggest several possible directions for future research. We believe that the most imminent challenges lie in scalable motion data management and support for complex operations and data types, such as motion episodes and multi-person skeleton sequences.

REFERENCES

[1] D. Mehta, O. Sotnychenko, F. Mueller, W. Xu, M. Elgharib, P. Fua, H. Seidel, H. Rhodin, G. Pons-Moll, and C. Theobalt, "XNect: Real-time multi-person 3D motion capture with a single RGB camera," *ACM Trans. Graph.*, vol. 39, no. 4, p. 82, 2020.

[2] A. Elhayek, E. de Aguiar, A. Jain, J. Tompson, L. Pishchulin, M. Andriluka, C. Bregler, B. Schiele, and C. Theobalt, "Efficient ConvNet-based marker-less motion capture in general scenes with a low number of cameras," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3810–3818.

[3] K. Sun, B. Xiao, D. Liu, and J. Wang, "Deep high-resolution representation learning for human pose estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 5693–5703.

[4] B. Ren, M. Liu, R. Ding, and H. Liu, "A survey on 3D skeleton-based action recognition using learning method," 2020, *arXiv:2002.05907*. [Online]. Available: https://arxiv.org/abs/2002.05907

[5] J. Wang, Y. Chen, S. Hao, X. Peng, and L. Hu, "Deep learning for sensor-based activity recognition: A survey," *Pattern Recognit. Lett.*, vol. 119, pp. 3–11, Mar. 2019.

[6] P. Wang, W. Li, P. Ogunbona, J. Wan, and S. Escalera, "RGB-D-based human motion recognition with deep learning: A survey," *Comput. Vis. Image Understand.*, vol. 171, pp. 118–139, Jun. 2018.

[7] B. Liu, H. Cai, Z. Ju, and H. Liu, "RGB-D sensing based human action and interaction analysis: A survey," *Pattern Recognit.*, vol. 94, pp. 1–12, Oct. 2019.

[8] R. Singh, A. Sonawane, and R. Srivastava, "Recent evolution of modern datasets for human activity recognition: A deep survey," *Multimedia Syst.*, vol. 26, no. 2, pp. 83–106, Apr. 2020.

[9] F. Han, B. Reily, W. Hoff, and H. Zhang, "Space-time representation of people based on 3D skeletal data: A review," *Comput. Vis. Image Understand.*, vol. 158, pp. 85–105, May 2017.

[10] L. Lo Presti and M. La Cascia, "3D skeleton-based human action classification: A survey," *Pattern Recognit.*, vol. 53, pp. 130–147, May 2016.

[11] C. Wan, L. Wang, and V. V. Phoha, "A survey on gait recognition," *ACM Comput. Surv.*, vol. 51, no. 5, p. 89, 2019.

[12] C. Chen, R. Jafari, and N. Kehtarnavaz, "A survey of depth and inertial sensor fusion for human action recognition," *Multimedia Tools Appl.*, vol. 76, no. 3, pp. 4405–4425, Feb. 2017.

[13] W. M. R. Wan Idris, A. Rafi, A. Bidin, A. A. Jamal, and S. A. Fadzli, "A systematic survey of martial art using motion capture technologies: The importance of extrinsic feedback," *Multimedia Tools Appl.*, vol. 78, no. 8, pp. 10113–10140, Apr. 2019.

[14] D. Webster and O. Celik, "Systematic review of kinect applications in elderly care and stroke rehabilitation," *J. Neuroeng. Rehabil.*, vol. 11, no. 1, p. 108, 2014.

[15] B. Pueo and J. Jimenez-Olmedo, "Application of motion capture technology for sport performance analysis," in *Retos: Nuevas Tendencias en Educación Física, Deporte y Recreación*. Madrid, Spain: Federación Española de Asociaciones de Docentes de Educación Física, 2017, pp. 241–247. [Online]. Available: https://dialnet.unirioja.es/servlet/revista?codigo=7258

[16] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2D pose estimation using part affinity fields," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1302–1310.

[17] S. Kreiss, L. Bertoni, and A. Alahi, "PifPaf: Composite fields for human pose estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 11977–11986.

[18] Y. Xu, Z. Shen, X. Zhang, Y. Gao, S. Deng, Y. Wang, Y. Fan, and E.-C. Chang, "Learning multi-level features for sensor-based human action recognition," *Pervasive Mobile Comput.*, vol. 40, pp. 324–338, Sep. 2017.

[19] A. Aristidou, D. Cohen-Or, J. K. Hodgins, Y. Chrysanthou, and A. Shamir, "Deep motifs and motion signatures," *ACM Trans. Graph.*, vol. 37, no. 6, p. 187, 2018.

[20] P. Elias, J. Sedmidubsky, and P. Zezula, "A real-time annotation of motion data streams," in *Proc. IEEE Int. Symp. Multimedia (ISM)*, Dec. 2017, pp. 154–161.

[21] J. Sedmidubsky, P. Elias, and P. Zezula, "Searching for variable-speed motions in long sequences of motion capture data," *Inf. Syst.*, vol. 80, pp. 148–158, Feb. 2019.

[22] A. Sharaf, M. Torki, M. E. Hussein, and M. El-Saban, "Real-time multi-scale action detection from 3D skeleton data," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, Jan. 2015, pp. 998–1005.

[23] L. Zhou, Z. Lu, H. Leung, and L. Shang, "Spatial temporal pyramid matching using temporal sparse representation for human motion retrieval," *Vis. Comput.*, vol. 30, nos. 6–8, pp. 845–854, Jun. 2014.

[24] R. Lan and H. Sun, "Automated human motion segmentation via motion regularities," *Vis. Comput.*, vol. 31, no. 1, pp. 35–53, Jan. 2015.

[25] M. Müller, A. Baak, and H.-P. Seidel, "Efficient and robust annotation of motion capture data," in *Proc. ACM SIGGRAPH/Eurographics Symp. Comput. Animation (SCA)*, 2009, pp. 17–26.

[26] K. Papadopoulos, E. Ghorbel, R. Baptista, D. Aouada, and B. E. Ottersten, "Two-stage RGB-based action detection using augmented 3D poses," in *Proc. 18th Int. Conf. Comput. Anal. Images Patterns (CAIP)*. Cham, Switzerland: Springer, 2019, pp. 26–35.

[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 *Proc. Graph. Interface Conf.*, 2004, pp. 185–194.

[28] A. Vögele, B. Krüger, and R. Klein, "Efficient unsupervised temporal segmentation of human motion," in *Proc. ACM SIGGRAPH/Eurographics Symp. Comput. Animation (SCA)*, 2014, pp. 167–176.

[29] S. Y. Boulahia, E. Anquetil, F. Multon, and R. Kulpa, "CuDi3D: Curvilinear displacement based approach for online 3D action detection," *Comput. Vis. Image Understand.*, vol. 174, pp. 57–69, Sep. 2018.

[30] M. Field, D. Stirling, Z. Pan, M. Ros, and F. Naghdy, "Recognizing human motions through mixture modeling of inertial data," *Pattern Recognit.*, vol. 48, no. 8, pp. 2394–2406, Aug. 2015.

[31] B. Kruger, A. Vogele, T. Willig, A. Yao, R. Klein, and A. Weber, "Efficient unsupervised temporal segmentation of motion data," *IEEE Trans. Multimedia*, vol. 19, no. 4, pp. 797–812, Apr. 2017.

[32] X. Yu, W. Liu, and W. Xing, "Behavioral segmentation for human motion capture data based on graph cut method," *J. Vis. Lang. Comput.*, vol. 43, pp. 50–59, Dec. 2017.

[33] M. Barnachon, S. Bouakaz, B. Boufama, and E. Guillou, "Ongoing human action recognition with motion capture," *Pattern Recognit.*, vol. 47, no. 1, pp. 238–247, Jan. 2014.

[34] H. Kadu and C.-C.-J. Kuo, "Automatic human mocap data classification," *IEEE Trans. Multimedia*, vol. 16, no. 8, pp. 2191–2202, Dec. 2014.

[35] J. Sedmidubsky and P. Zezula, "Augmenting spatio-temporal human motion data for effective 3D action recognition," in *Proc. IEEE Int. Symp. Multimedia (ISM)*, Dec. 2019, pp. 204–207.

[36] J. Sedmidubsky, P. Elias, and P. Zezula, "Effective and efficient similarity searching in motion capture data," *Multimedia Tools Appl.*, vol. 77, no. 10, pp. 12073–12094, May 2018.

[37] S. Yan, Y. Xiong, and D. Lin, "Spatial temporal graph convolutional networks for skeleton-based action recognition," in *Proc. 32nd AAAI Conf. Artif. Intell.*, 2018, pp. 7444–7452.

[38] Y. Wang and M. Neff, "Deep signatures for indexing and retrieval in large motion databases," in *Proc. 8th ACM SIGGRAPH Conf. Motion Games*, Nov. 2015, pp. 37–45.

[39] J. Sedmidubsky, P. Budikova, V. Dohnal, and P. Zezula, "Motion words: A text-like representation of 3D skeleton sequences," in *Proc. 42nd Eur. Conf. Inf. Retr. (ECIR)*. Springer, 2020, pp. 1–14.

[40] P. Budikova, J. Sedmidubsky, J. Horvath, and P. Zezula, "Towards scalable retrieval of human motion episodes," in *Proc. IEEE Int. Symp. Multimedia (ISM)*, Dec. 2020, pp. 49–56.

[41] J. Baumann, R. Wessel, B. Krüger, and A. Weber, "Action graph a versatile data structure for action recognition," in *Proc. Int. Conf. Comput. Graph. Theory Appl. (GRAPP)*, Jan. 2014, pp. 1–10.

[42] M. Barnachon, S. Bouakaz, B. Boufama, and E. Guillou, "A real-time system for motion retrieval and interpretation," *Pattern Recognit. Lett.*, vol. 34, no. 15, pp. 1789–1798, Nov. 2013.

[43] C. Ren, X. Lei, and G. Zhang, "Motion data retrieval from very large motion databases," in *Proc. Int. Conf. Virtual Reality Vis.*, Nov. 2011, pp. 70–77.

[44] J. Sedmidubsky, J. Valcik, and P. Zezula, "A key-pose similarity algorithm for motion data retrieval," in *Proc. 15th Int. Conf. Adv. Concepts Intell. Vis. Syst.* Berlin, Germany: Springer, 2013, pp. 669–681.

[45] J. Wang, Z. Liu, Y. Wu, and J. Yuan, "Mining actionlet ensemble for action recognition with depth cameras," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 1290–1297.

[46] A. Shahroudy, J. Liu, T.-T. Ng, and G. Wang, "NTU RGB+D: A large scale dataset for 3D human activity analysis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1010–1019.

[47] Y. Du, W. Wang, and L. Wang, "Hierarchical recurrent neural network for skeleton based action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1110–1118.

[48] J. Liu, G. Wang, L.-Y. Duan, K. Abdiyeva, and A. C. Kot, "Skeleton-based human action recognition with global context-aware attention LSTM networks," *IEEE Trans. Image Process.*, vol. 27, no. 4, pp. 1586–1599, Apr. 2018.

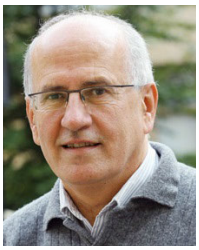
[49] W. Zhu, C. Lan, J. Xing, W. Zeng, Y. Li, L. Shen, and X. Xie, "Co-occurrence feature learning for skeleton based action recognition using regularized deep LSTM networks," in *Proc. 30th AAAI Conf. Artif. Intell. (AAAI)*, 2016, pp. 3697–3703.
[50] H. Wang and L. Wang, "Modeling temporal dynamics and spatial configurations of actions using two-stream recurrent neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3633–3642.
[51] D. Dotti, E. Ghaleb, and S. Asteriadis, "Temporal triplet mining for personality recognition," in *Proc. 15th IEEE Int. Conf. Autom. Face Gesture Recognit. (FG)*, Nov. 2020, pp. 171–178.
[52] S. Laraba, M. Brahimi, J. Tilmanne, and T. Dutoit, "3D skeleton-based action recognition by representing motion capture sequences as 2D-RGB images," *Comput. Animation Virtual Worlds*, vol. 28, nos. 3–4, p. e1782, 2017.
[53] M. Liu, H. Liu, and C. Chen, "Enhanced skeleton visualization for view invariant human action recognition," *Pattern Recognit.*, vol. 68, pp. 346–362, Aug. 2017.
[54] P. Wang, W. Li, C. Li, and Y. Hou, "Action recognition based on joint trajectory maps with convolutional neural networks," *Knowl.-Based Syst.*, vol. 158, pp. 43–53, Oct. 2018.
[55] T. Huynh-The, C.-H. Hua, N. A. Tu, and D.-S. Kim, "Learning geometric features with dual–stream CNN for 3D action recognition," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2020, pp. 2353–2357.
[56] C. Li, Z. Cui, W. Zheng, C. Xu, and J. Yang, "Spatio-temporal graph routing for skeleton-based action recognition," in *Proc. 32nd Int. Conf. Artif. Intell. (AAAI)*, 2018, pp. 7444–7452.
[57] H. Yang, Y. Gu, J. Zhu, K. Hu, and X. Zhang, "PGCN-TCA: Pseudo graph convolutional network with temporal and channel-wise attention for skeleton-based action recognition," *IEEE Access*, vol. 8, pp. 10040–10047, 2020.
[58] R. Liu, C. Xu, T. Zhang, W. Zhao, Z. Cui, and J. Yang, "Si-GCN: Structure-induced graph convolution network for skeleton-based action recognition," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2019, pp. 1–8.
[59] G. Zhu, L. Zhang, H. Li, P. Shen, S. A. A. Shah, and M. Bennamoun, "Topology-learnable graph convolution for skeleton-based action recognition," *Pattern Recognit. Lett.*, vol. 135, pp. 286–292, Jul. 2020.
[60] J. C. Nuñez, R. Cabido, J. J. Pantrigo, A. S. Montemayor, and J. F. Vélez, "Convolutional neural networks and long short-term memory for skeleton-based human activity and hand gesture recognition," *Pattern Recognit.*, vol. 76, pp. 80–94, Apr. 2018.
[61] Y. Wu, L. Wei, and Y. Duan, "Deep spatiotemporal LSTM network with temporal pattern feature for 3D human action recognition," *Comput. Intell.*, vol. 35, no. 3, pp. 535–554, Aug. 2019.
[62] J. Li, X. Xie, Q. Pan, Y. Cao, Z. Zhao, and G. Shi, "SGM-Net: Skeleton-guided multimodal network for action recognition," *Pattern Recognit.*, vol. 104, Aug. 2020, Art. no. 107356.
[63] S. Zagoruyko and N. Komodakis, "Learning to compare image patches via convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 4353–4361.
[64] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 815–823.
[65] Q. Ke, M. Bennamoun, S. An, F. Sohel, and F. Boussaid, "A new representation of skeleton sequences for 3D action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3288–3297.
[66] D. Wu, J. Chen, N. Sharma, S. Pan, G. Long, and M. Blumenstein, "Adversarial action data augmentation for similar gesture action recognition," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2019, pp. 1–8.
[67] M. Espadoto, R. M. Martins, A. Kerren, N. S. T. Hirata, and A. C. Telea, "Toward a quantitative survey of dimension reduction techniques," *IEEE Trans. Vis. Comput. Graphics*, vol. 27, no. 3, pp. 2153–2173, Mar. 2019.
[68] Q. Lei, H.-B. Zhang, J.-X. Du, T.-C. Hsiao, and C.-C. Chen, "Learning effective skeletal representations on RGB video for fine-grained human action quality assessment," *Electronics*, vol. 9, no. 4, p. 568, Mar. 2020.
[69] M. Meshry, M. E. Hussein, and M. Torki, "Linear-time online action detection from 3D skeletal data using bags of gesturelets," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2016, pp. 1–9.
[70] S. Wu, Z. Wang, and S. Xia, "Indexing and retrieval of human motion data by a hierarchical tree," in *Proc. 16th ACM Symp. Virtual Reality Softw. Technol. (VRST)*, 2009, pp. 207–214.
[71] B. Krüger, J. Tautges, A. Weber, and A. Zinke, "Fast local and global similarity searches in large motion capture databases," in *Proc. Eurographics/ACM SIGGRAPH Symp. Comput. Animation (SCA)*, 2010, pp. 1–10.
[72] M.-W. Chao, C.-H. Lin, J. Assa, and T.-Y. Lee, "Human motion retrieval from hand-drawn sketch," *IEEE Trans. Vis. Comput. Graphics*, vol. 18, no. 5, pp. 729–740, May 2012.
[73] T. Ren, W. Li, Z. Jiang, X. Li, Y. Huang, and J. Peng, "Video-based human motion capture data retrieval via MotionSet network," *IEEE Access*, vol. 8, pp. 186212–186221, 2020.
[74] 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.
[75] M. Müller and T. R. O. der, "Motion templates for automatic classification and retrieval of motion capture data," in *Proc. ACM SIGGRAPH/Eurographics Symp. Comput. Animation (SCA)*, 2006, pp. 137–146.
[76] Z. Deng, Q. Gu, and Q. Li, "Perceptually consistent example-based human motion retrieval," in *Proc. Symp. Interact. 3D Graph. Games (I3D)*, 2009, pp. 191–198.
[77] M. Kapadia, I.-K. Chiang, T. Thomas, N. I. Badler, and J. T. Kider, "Efficient motion retrieval in large motion databases," in *Proc. ACM SIGGRAPH Symp. Interact. 3D Graph. Games (I3D)*, 2013, pp. 19–28.
[78] J. Sedmidubsky, P. Zezula, and J. Svec, "Fast subsequence matching in motion capture data," in *21st Eur. Conf. Adv. Databases Inf. Syst. (ADBIS)*. Cham, Switzerland: Springer, 2017, pp. 50–72.
[79] M. G. Choi and T. Kwon, "Motion rank: Applying page rank to motion data search," *Vis. Comput.*, vol. 35, no. 2, pp. 289–300, Feb. 2019.
[80] J. Bernard, N. Wilhelm, B. Kruger, T. May, T. Schreck, and J. Kohlhammer, "MotionExplorer: Exploratory search in human motion capture data based on hierarchical aggregation," *IEEE Trans. Vis. Comput. Graphics*, vol. 19, no. 12, pp. 2257–2266, Dec. 2013.
[81] N. Numaguchi, A. Nakazawa, T. Shiratori, and J. K. Hodgins, "A puppet interface for retrieval of motion capture data," in *Proc. ACM SIGGRAPH/Eurographics Symp. Comput. Animation (SCA)*, 2011, pp. 157–166.
[82] L. Chen, Y. Gao, X. Song, Z. Li, X. Miao, and C. S. Jensen, "Indexing metric spaces for exact similarity search," 2020, *arXiv:2005.03468*. [Online]. Available: https://arxiv.org/abs/2005.03468
[83] M. Devanne, S. Berretti, P. Pala, H. Wannous, M. Daoudi, and A. D. Bimbo, "Motion segment decomposition of RGB-D sequences for human behavior understanding," *Pattern Recognit.*, vol. 61, pp. 222–233, Jun. 2017.
[84] H. Wu, J. Shao, X. Xu, Y. Ji, F. Shen, and H. Tao Shen, "Recognition and detection of two-person interactive actions using automatically selected skeleton features," *IEEE Trans. Human-Mach. Syst.*, vol. 48, no. 3, pp. 304–310, Jun. 2018.
[85] M. Zanfir, M. Leordeanu, and C. Sminchisescu, "The moving pose: An efficient 3D kinematics descriptor for low-latency action recognition and detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 2752–2759.
[86] X. Zhao, X. Li, C. Pang, X. Zhu, and Q. Z. Sheng, "Online human gesture recognition from motion data streams," in *Proc. 21st ACM Int. Conf. Multimedia*, 2013, pp. 23–32.
[87] Y. Li, C. Lan, J. Xing, W. Zeng, C. Yuan, and J. Liu, "Online human action detection using joint classification-regression recurrent neural networks," in *Proc. 14th Eur. Conf. Comput. Vis. (ECCV)*. Springer, 2016, pp. 203–220. [Online]. Available: https://www.springer.com/gp/book/9783319464770
[88] F. Patrona, A. Chatzitofis, D. Zarpalas, and P. Daras, "Motion analysis: Action detection, recognition and evaluation based on motion capture data," *Pattern Recognit.*, vol. 76, pp. 612–622, Apr. 2018.
[89] S. Song, C. Lan, J. Xing, W. Zeng, and J. Liu, "Spatio-temporal attention-based LSTM networks for 3D action recognition and detection," *IEEE Trans. Image Process.*, vol. 27, no. 7, pp. 3459–3471, Jul. 2018.
[90] F. Carrara, P. Elias, J. Sedmidubsky, and P. Zezula, "LSTM-based real-time action detection and prediction in human motion streams," *Multimedia Tools Appl.*, vol. 78, no. 19, pp. 27309–27331, Oct. 2019.
[91] S. Li, K. Li, and Y. Fu, "Early recognition of 3D human actions," *ACM Trans. Multimedia Comput., Commun., Appl.*, vol. 14, no. 1, p. 20, 2018.
[92] A. Jain, A. R. Zamir, S. Savarese, and A. Saxena, "Structural-RNN: Deep learning on spatio-temporal graphs," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 5308–5317.

[93] C. Wu, J. Zhang, S. Savarese, and A. Saxena, "Watch-n-patch: Unsupervised understanding of actions and relations," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 4362–4370.
[94] D. Holden, J. Saito, and T. Komura, "A deep learning framework for character motion synthesis and editing," *ACM Trans. Graph. (TOG)*, vol. 35, no. 4, p. 138, 2016.
[95] S. Starke, Y. Zhao, T. Komura, and K. Zaman, "Local motion phases for learning multi-contact character movements," *ACM Trans. Graph.*, vol. 39, p. 54, Jun. 2020.
[96] W. Geng and G. Yu, "Reuse of motion capture data in animation: A review," in *Computational Science and Its Applications (ICCSA)*. Berlin, Germany: Springer, 2003, pp. 620–629.
[97] M. J. Kyan, G. Sun, H. Li, L. Zhong, P. Muneesawang, N. Dong, B. Elder, and L. Guan, "An approach to ballet dance training through ms kinect and visualization in a cave virtual reality environment," *ACM Trans. Intell. Syst. Technol.*, vol. 6, no. 2, p. 23, 2015.
[98] Y. Yan, O. M. Omisore, Y. Xue, H. Li, Q. Liu, Z. Nie, J. Fan, and L. Wang, "Classification of neurodegenerative diseases via topological motion analysis—A comparison study for multiple gait fluctuations," *IEEE Access*, vol. 8, pp. 96363–96377, 2020.
[99] Y. Zhang and Y. Ma, "Application of supervised machine learning algorithms in the classification of sagittal gait patterns of cerebral palsy children with spastic diplegia," *Comput. Biol. Med.*, vol. 106, pp. 33–39, Mar. 2019.
[100] R. Watari, D. Kobsar, A. Phinyomark, S. Osis, and R. Ferber, "Determination of patellofemoral pain sub-groups and development of a method for predicting treatment outcome using running gait kinematics," *Clin. Biomech.*, vol. 38, pp. 13–21, Oct. 2016.
[101] W. R. Johnson, A. Mian, C. J. Donnelly, D. Lloyd, and J. Alderson, "Predicting athlete ground reaction forces and moments from motion capture," *Med. Biol. Eng. Comput.*, vol. 56, no. 10, pp. 1781–1792, Oct. 2018.
[102] L. Pogrzeba, T. Neumann, M. Wacker, and B. Jung, "Analysis and quantification of repetitive motion in long-term rehabilitation," *IEEE J. Biomed. Health Informat.*, vol. 23, no. 3, pp. 1075–1085, May 2019.
[103] S. Schez-Sobrino, D. Vallejo, D. N. Monekosso, C. Glez-Morcillo, and P. Remagnino, "A distributed gamified system based on automatic assessment of physical exercises to promote remote physical rehabilitation," *IEEE Access*, vol. 8, pp. 91424–91434, 2020.
[104] A. Aristidou, A. Shamir, and Y. Chrysanthou, "Digital dance ethnography: Organizing large dance collections," *Journal on Computing and Cultural Heritage*, vol. 12, no. 4, p. 29, 2019.
[105] C. Joyce, A. Burnett, J. Cochrane, and K. Ball, "Three-dimensional trunk kinematics in golf: Between-club differences and relationships to clubhead speed," *Sports Biomech.*, vol. 12, no. 2, pp. 108–120, Jun. 2013.
[106] H. Pirsiavash, C. Vondrick, and A. Torralba, "Assessing the quality of actions," in *Proc. 11th Eur. Conf. Comput. Vis. (ECCV)*. Cham, Switzerland: Springer, 2014, pp. 556–571.
[107] T. Shimizu, R. Hachiuma, H. Saito, T. Yoshikawa, and C. Lee, "Prediction of future shot direction using pose and position of tennis player," in *Proc. 2nd Int. Workshop Multimedia Content Anal. Sports (MMSports)*, New York, NY, USA, 2019, pp. 59–66.
[108] D. Zecha, C. Eggert, and R. Lienhart, "Pose estimation for deriving kinematic parameters of competitive swimmers," *Electron. Imag.*, vol. 2017, no. 16, pp. 21–29, Jan. 2017.
[109] M. Einfalt, C. Dampeyrou, D. Zecha, and R. Lienhart, "Frame-level event detection in athletics videos with pose-based convolutional sequence networks," in *Proc. 2nd Int. Workshop Multimedia Content Anal. Sports (MMSports)*, 2019, pp. 42–50.
[110] J. C. P. Chan, H. Leung, J. K. T. Tang, and T. Komura, "A virtual reality dance training system using motion capture technology," *IEEE Trans. Learn. Technol.*, vol. 4, no. 2, pp. 187–195, Apr. 2011.
[111] F. Anderson, T. Grossman, J. Matejka, and G. W. Fitzmaurice, "YouMove: Enhancing movement training with an augmented reality mirror," in *Proc. 26th Annu. ACM Symp. User Interface Softw. Technol. (UIST)*, 2013, pp. 311–320.
[112] M. M. Islam, A. Lam, H. Fukuda, Y. Kobayashi, and Y. Kuno, "An intelligent shopping support robot: Understanding shopping behavior from 2D skeleton data using GRU network," *ROBOMECH J.*, vol. 6, no. 1, p. 18, Dec. 2019.
[113] T. Hu, X. Zhu, S. Wang, and L. Duan, "Human interaction recognition using spatial-temporal salient feature," *Multimedia Tools Appl.*, vol. 78, no. 20, pp. 28715–28735, Oct. 2019.
[114] P. Woznowski *et al.*, *SPHERE: A Sensor Platform for Healthcare in a Residential Environment*. Cham, Switzerland: Springer, 2017, pp. 315–333.
[115] R. Henschel, Y. Zou, and B. Rosenhahn, "Multiple people tracking using body and joint detections," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2019, pp. 1–10.
[116] E. Barsoum, J. Kender, and Z. Liu, "HP-GAN: Probabilistic 3D human motion prediction via GAN," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2018, pp. 1418–1427.
[117] A. Aristidou, D. Cohen-Or, J. K. Hodgins, and A. Shamir, "Self-similarity analysis for motion capture cleaning," *Comput. Graph. Forum*, vol. 37, no. 2, pp. 297–309, May 2018.
[118] P. Zezula, G. Amato, V. Dohnal, and M. Batko, *Similarity Search: The Metric Space Approach* (Advances in Database Systems), vol. 32. Springer, 2006. [Online]. Available: https://www.springer.com/gp/book/9780387291468
[119] B. Morschheuser, J. Hamari, J. Koivisto, and A. Maedche, "Gamified crowdsourcing: Conceptualization, literature review, and future agenda," *Int. J. Hum.-Comput. Stud.*, vol. 106, pp. 26–43, Oct. 2017.
[120] E. Chávez, G. Navarro, R. Baeza-Yates, and J. L. Marroquin, "Searching in metric spaces," *ACM Comput. Surv.*, vol. 33, no. 3, pp. 273–321, Sep. 2001.
[121] V. Mic, D. Novak, and P. Zezula, "Binary sketches for secondary filtering," *ACM Trans. Inf. Syst.*, vol. 37, no. 1, pp. 1–28, Jan. 2019.
[122] N. Dasgupta, M. R. Banaji, and R. P. Abelson, "Group entitativity and group perception: Associations between physical features and psychological judgment," *J. Personality Social Psychol.*, vol. 77, no. 5, p. 991, 1999.
[123] D. Lakens, "Movement synchrony and perceived entitativity," *J. Exp. Social Psychol.*, vol. 46, no. 5, pp. 701–708, Sep. 2010.
[124] V. J. Kok, M. K. Lim, and C. S. Chan, "Crowd behavior analysis: A review where physics meets biology," *Neurocomputing*, vol. 177, pp. 342–362, Feb. 2016.

JAN SEDMIDUBSKY received the Ph.D. degree from Masaryk University, Czech Republic, in 2011, along with the dean's and rector's prize for a distinguished dissertation thesis. He is currently a Researcher of computer science with Masaryk University. His research activities are primarily concentrated on developing effective and efficient similarity-based processing techniques, with a special emphasis on the domain of 3-D motion capture data.

PETR ELIAS received the Ph.D. degree from Masaryk University, Czech Republic, in 2020. He is currently a Postdoctoral Researcher in computer science with Masaryk University. His research interests include similarity-based human motion data understanding, including action recognition, searching, and filtering in 2-D and 3-D skeleton data.

PETRA BUDIKOVA received the Ph.D. degree from Masaryk University, Czech Republic, in 2013. She is currently a Researcher of computer science with Masaryk University. In her research, she focuses on algorithms for efficient and scalable processing of multimedia data. After a period of work on image search and annotations, she also applies her skills in the field of human motion understanding.



PAVEL ZEZULA is currently a Professor of computer science with Masaryk University, Czech Republic. He is a coauthor of the famous metric-based similarity search structure M-Tree and book *Similarity Search: The Metric Space Approach*. His professional research interests include content-based retrieval, large-scale similarity search, and big data analysis.