Motion Words: A Text-Like Representation of 3D Skeleton Sequences

Jan Sedmidubsky(✉), Petra Budikova, Vlastislav Dohnal, and Pavel Zezula

Masaryk University, Brno, Czechia
{xsedmid,budikova,dohnal,zezula}@fi.muni.cz

Abstract. There is a growing amount of human motion data captured as a continuous 3D skeleton sequence without any information about its semantic partitioning. To make such unsegmented and unlabeled data efficiently accessible, we propose to transform them into a text-like representation and employ well-known text retrieval models. Specifically, we partition each motion synthetically into a sequence of short segments and quantize the segments into motion words, i.e. compact features with similar characteristics as words in text documents. We introduce several quantization techniques for building motion-word vocabularies and propose application-independent criteria for assessing the vocabulary quality. We verify these criteria on two real-life application scenarios.

Keywords: 3D skeleton sequence · Motion word · Motion vocabulary · Quantization · Border problem · Text-based processing

1 Introduction

In recent years, we have witnessed a rapid development of motion capture devices and 3D pose-estimation methods [2] that enable recording human movements as a sequence of *poses*. Each pose keeps the 3D coordinates of important *skeleton joints* in a specific time moment. Effective and efficient processing of such spatio-temporal data is very desirable in many application domains, ranging from computer animation, through sports and medicine, to security [5,7,9].

To illustrate the range of possible tasks over motion data, let us assume that we have the 3D skeleton data from a figure skating competition. Existing research mainly focuses on *action recognition* [23], i.e. categorizing the figure performed in a given, manually selected motion segment. This is typically solved using convolutional [1,17] or recurrent [10,20,22] neural-network classifiers. However, this approach is not applicable to other situations where motion data are captured as long continuous sequences without explicit knowledge of semantic partitioning. In such cases, other techniques need to be applied, e.g., *subsequence search* to find all competitors who performed the triple Axel jump, or *similarity joins* to identify different performances of the same choreography, similar choreographies, or the most common figures. These techniques require identifying query-relevant

J. M. Jose et al. (Eds.): ECIR 2020, LNCS 12035, pp. 527–541, 2020.
https://doi.org/10.1007/978-3-030-45439-5_35

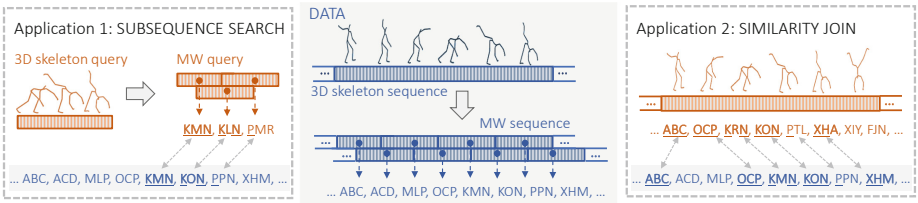

Fig. 1. Representing motions by motion words: both data and queries are transformed into MW sequences and efficiently organized and processed by text-based approaches.

subsequences within the continuous motion data. To allow efficient evaluation of such queries, the data need to be automatically segmented and indexed.

Since a universal semantic segmentation is hardly achievable, we suggest to partition each motion sequence synthetically into short fixed-size *segments* whose length is smaller than the expected size of future queries. In this way, we transform the input motion into an ordered sequence of segments, structurally similar to a text document. To complete the analogy, we quantize the segments into compact representations, denoted as *motion words* (MWs), having similar properties as words in text documents. Individual MWs deal with the spatial variability of the short segments, whereas the temporal variability of longer motions is captured by the MW order and quantified by mature text-retrieval models [12]. We believe that such universal text-based representation is applicable for a wide range of applications that need to process continuous motion data efficiently, as illustrated in Fig. 1.

In this paper, we mainly focus on effective quantization of the motion segments to build a *vocabulary* of motion words. The most desirable MW property is that two MWs *match* each other if their corresponding segments exhibit similar movement characteristics, and do *not match* if the segments are dissimilar. This is challenging with the quantization approach, since it is in general not possible to divide a given space in such way that all pairs of similar objects are in the same partition. Some pairs of similar segments thus get separated by partition borders and become non-matching, which we denote as the *border problem*. We answer this challenge by designing two MW construction techniques that reduce the border problem but still enable efficient organization using text retrieval techniques. Furthermore, we recommend generic (application-independent) criteria for selection of a suitable vocabulary for specific application needs, and verify the usability of such criteria on two real-life applications.

2 Related Work and Our Contributions

Most existing works that process *continuous* 3D skeleton sequences in an *unsupervised* way focus on subsequence search [18], unsupervised segmentation [8], or anticipating future actions based on the past-to-current data [4]. In [18], the continuous sequences are synthetically partitioned into a lot of overlapping and

variable-size segments that are represented by 4,096D deep features. However, indexing a large number of such very high-dimensional features is costly. To move towards more efficient processing, the approaches in [3,11] quantize the segment features using a single k-means clustering. However, with such simple quantization the border problem appears frequently, which decreases the effectiveness of applications with an increasing number of clusters (i.e. the vocabulary size).

In our research, we also take inspiration from image processing where high-dimensional image features are quantized into visual words. There are two lines of research that are important to us: fundamental quantization techniques, and reducing the border problem. The image quantization strategies have evolved from basic k-means clustering used in [21], through cluster hierarchies [14], approximate k-means [16], to recent deep neural-network approaches [24]. The influence of the border problem can be reduced using a weighted combination of the nearest visual words for each feature [16], or by a consensus voting of multiple independent vocabularies [6].

Contributions of This Paper

We propose an effective quantization of unlabeled 3D skeleton data into sequences of motion words that can be efficiently managed by text-retrieval techniques. In contrast to previous works, we give a particular attention to the border problem. Specifically,

- we systematically analyze the process of MW vocabulary construction and discuss possible solutions of the border problem (Sect. 3);
- we propose application-independent criteria that do not require labeled data for selecting a suitable MW vocabulary for a given task (Sect. 3.3);
- we implement three vocabulary construction techniques that differ in dealing with the border problem, and evaluate their quality (Sect. 4);
- we verify the suitability of the proposed criteria by evaluating the best-ranked vocabularies in the context of two real-world applications (Sect. 5).

3 MW Vocabulary Construction

The motion-word (MW) approach assumes that the continuous 3D skeleton data are cut into short, possibly overlapping segments which are consequently transformed into the motion words. The segment and overlap lengths are important parameters of the whole system and have also been studied in our experiments, however their thorough analysis is out of the scope of this paper. Therefore, we assume that a suitable segmentation is available, and focus solely on transforming the segment space into the space of motion words, denoted as the *MW vocabulary*.

The MW vocabulary consists of a finite set of motion words and a Boolean-valued *MW matching function* that determines whether two MWs are considered equal: $match^{MW} : MW \times MW \rightarrow \{0,1\}$. The Boolean matching of words

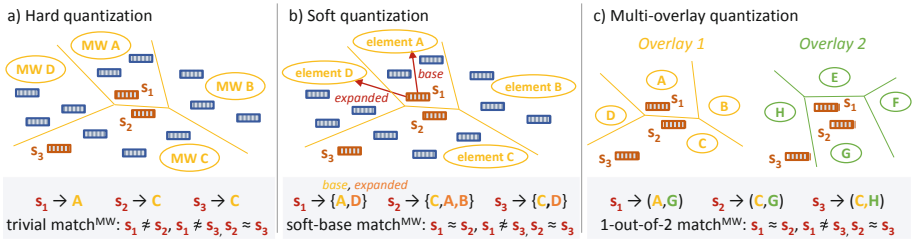

Fig. 2. Comparison of the hard, soft, and multi-overlay quantization of segments.

is a standard text-processing primitive required by most text retrieval techniques [12]. The transformation from segments to MWs has to be *similarity-preserving*: with a high probability, similar segment pairs need to be mapped to matching MWs and dissimilar segment pairs to non-matching MWs. Noticeably, the vocabulary construction can be investigated independently of a particular application, since it only considers the distribution of segments in the segment space. We propose to build the MW vocabulary using *quantization* of the segment space, which can be seen as analogous to the word stemming in text processing.

In the following, we first review the standard quantization approach that leads to a *basic MW model* and discuss its limitations, namely the border problem. Next, we introduce a *generalized MW model* with two techniques for reducing the border problem. Lastly, we present the evaluation methodology that we propose for comparing the quality of different vocabularies.

3.1 Basic MW Model

Basic data quantization is usually performed by the k-means algorithm that divides the segment space into non-overlapping partitions [3,11,21]. Each partition can be assigned a one-dimensional identifier, which constitutes the motion word. Each motion segment is associated with exactly one MW, which we denote as the *hard quantization* (Fig. 2a). To compare two *hard MWs*, a trivial MW matching function is defined: it returns 1 for identical words and 0 otherwise.

Using this approach, the 3D skeleton data are transformed into a sequence of scalar MWs to be readily processed by the standard text-retrieval tools. However, the hard quantization makes it difficult to preserve the similarity between the segments. Unless the input data are inherently well-clustered, which is not likely in the high-dimensional segment space, it is not possible to avoid the border problem, i.e. the situations when two similar segments get assigned to different MWs (s_1 and s_2 in Fig. 2a). Moreover, finding a good clustering is computationally expensive. Therefore, approximate or sampling methods are often used for large data, which makes the border problem even more pronounced.

3.2 Generalized MW Model

We believe that the border problem can be reduced significantly if we allow a given segment to be associated with several partitions of the input space. Therefore, we define the *generalized MW* as a collection of *MW elements*, where each element corresponds to a single partition of the input space. In contrast to the basic model where individual MWs are atomic and mutually exclusive, the generalized MWs may share some MW elements. This allows us to define a more fine-grained MW matching function that better approximates the original similarity between the motion segments.

As illustrated in Fig. 2b and c, we adopt the following two orthogonal principles of selecting the MW elements for a given segment.

- *Soft quantization.* Recall again that the border problem occurs when two similar segments are separated into different partitions. Intuitively, at least one of these segments has to lie near the partition border. Segment s_1 in Fig. 2a lies outside the partition D but is close to its borders, so there is a good chance that some segments similar to s_1 are in D. Therefore, it could be helpful to associate s_1 also with D. Following this idea, we define the *soft MW* for s_1 as an *ordered set* of one or more MW elements, where the first *base element* identifies the partition containing s_1 and the remaining *expanded elements* refer to the partitions that are sufficiently close to s_1 (see Fig. 2b for illustration). A naive MW matching function could return 1 whenever the intersection of two soft MWs is non-empty, however this tends to match even segments that are not so close in the segment space (s_1 and s_3 in Fig. 2b). Therefore, our *soft-base* matching function returns 1 only if the intersection contains at least one base element.
- *Multi-overlay quantization*: So far, we have assumed that the MW elements are taken from a single partitioning of the segment space. However, it is also possible to employ several independent *partitioning overlays* obtained by different methods. A single overlay may incorrectly separate a pair of similar segments, but it is less probable that the same pair will be separated by the other independent overlays. We define the *multi-overlay MW* as an *n-tuple* of MW elements that are assigned to a given segment in the individual overlays. To decide whether two MWs match, the consensus of m out of n MW elements is used. The matching function returns 1 if the multi-overlay MWs agree on at least m positions of the respective n-tuples (see Fig. 2c).

By allowing the MWs to be compound, we improve the quantization quality but create new challenges regarding indexability. The generalized MWs are no longer scalar and cannot be simply treated the same way as text words. However, existing text retrieval tools can be adjusted to index both the soft and multi-overlay MWs, as briefly discussed in Sect. 4.4.

3.3 Evaluation Methodology

For evaluating MW vocabularies, we need to consider two different aspects: (i) *vocabulary quality* – measured by the application-independent ability to

perform a similarity-preserving transformation from the segment space to the MW space, and (ii) *vocabulary usefulness* – measured by effectiveness of the application employing the specific vocabulary. Our objective is to show that both vocabulary quality and vocabulary usefulness are related, so we can choose a suitable vocabulary without evaluating it within the real application, i.e. not needing the application ground truth (GT).

In the following, we introduce the dataset used for both types of evaluation, and describe the application-independent vocabulary quality measures that are examined in Sect. 4. The vocabulary usefulness is discussed in Sect. 5.

Dataset. We adopt the HDM05 dataset [13] of 3D skeleton sequences, which consists of 2,345 labeled actions categorized in 130 classes. The actions capture exercising and daily movement activities with the sampling frequency of 120 Hz and track 31 skeleton joints. The action length ranges from 13 frames (108 ms) to 900 frames (7.5 s). We use this dataset to evaluate the MW usefulness in two applications: a *kNN classification of actions*, and a *similar action search*. These applications do not require complex retrieval algorithms and allow us to clearly show the effect of MWs on application effectiveness.

Both the vocabulary construction and the application-independent quality assessment are designed for completely *unlabeled* segment data, which we extract from the HDM05 dataset as follows. We divide each action synthetically into a sequence of overlapping segments. As recommended in [3], we fix the segment length to 80 frames and the segment overlap to 64 frames, so the segments are shifted by 16 frames. This generates **28 k** segments in total, with 12 segments per action on average. We also down-sample the segments to 12 frames per second. The *similarity* of any two segments is determined by the Dynamic Time Warping (DTW), where the pose distance inside DTW is computed as the sum of Euclidean distances between the 3D coordinates of the corresponding joints.

Estimating GT for Unlabeled Segments. The similarity-preserving property states that similar segments should be mapped to matching MWs, whereas dissimilar segments to non-matching MWs. To be able to check this property for a given vocabulary, we need a ground truth (GT) of similar and dissimilar segment pairs. Since the segments have no semantic labels, we can only use pairwise distances to estimate the GT. Using the distance distribution of all segments from our dataset, we determine two threshold distances that divide the segment pairs into similar pairs, dissimilar pairs, and a grey zone. In particular, the 0.5^{th} percentile distance becomes the similarity threshold T_{sim} and all segment pairs with the mutual distance lower than T_{sim} are the GT's similar pairs. The 40^{th} percentile becomes the dissimilarity threshold T_{dissim} which defines the dissimilar pairs. Both the thresholds are set tightly to eliminate the chance that semantically unrelated segments are considered similar and vice versa. The segment pairs with mutual distance between T_{sim} and T_{dissim} form the grey zone and are ignored in the vocabulary quality evaluations.

Vocabulary Quality Measures. To assess how well a given MW vocabulary manages to match a given segment with similar segments, we use standard IR measures of *precision* (P) and *recall* (R) computed over the above-described GT of similar and dissimilar segment pairs: $P = \frac{tp}{tp+fp}$ and $R = \frac{tp}{tp+fn}$, where the *true positives (tp)* are pairs of similar segments mapped to matching MWs, *false positives (fp)* are dissimilar segments with matching MWs, etc. To quantify the trade-off between P and R, we employ the F_β *score* $= (1+\beta^2) \cdot \frac{P \cdot R}{(\beta^2 \cdot R)+P}$, where the positive real β is used to adjust the importance of the precision and recall according to the target application preferences.

As already mentioned, we test our vocabularies in context of two applications with different needs. The kNN classification requires high precision of retrieved actions for correct decision, but some positives can be missed. On the other hand, action search typically requires high recall. With these two applications in mind, we select the following two F scores for our experiments: $F_{0.25}$ score that emphasizes precision, as required by the classification task, and F_1 score that is the harmonic mean of both precision and recall and complies to the needs of a search-oriented application.

4 Implementation and Evaluation

To create a vocabulary, we use a Voronoi partitioning of the segment space. It assumes a set of sites (*pivots*) is selected beforehand by a particular selection algorithm. The Voronoi cell of pivot p is formed by all segments closer to p than to the other pivots. The pivots' IDs become the motion words or MW elements. Regarding the pivot selection, we must keep in mind that the segment space may not be the Euclidean space, which is our case with DTW that brakes the triangle inequality. So a particular pivot selection algorithm must respect that an artificial data item (e.g., a mean vector) cannot be computed.

In the following, we introduce algorithms implementing the aforementioned MW vocabulary construction principles, and show how the quality measures introduced in Sect. 3.3 can be used to tune the parameters of the algorithms.

4.1 Hard Quantization

Firstly, we analyze the viability of three pivot selection techniques: the *k-medoids*, the *hierarchical k-medoids*, and a *random selection*. We also study the influence of the number of pivots, which determines the cardinality of the vocabulary.

Implementation. The *k-medoids* algorithm is a variation of the k-means clustering that is mostly used for quantization. It works in iterations, gradually moving from a random set of pivots to more optimal ones. With the k-medoids, the pivots must be selected from existing motion segments. The optimization criterion is to minimize the sum of distances to other segments within the cluster. The algorithm does not guarantee to find the global optimum and is very

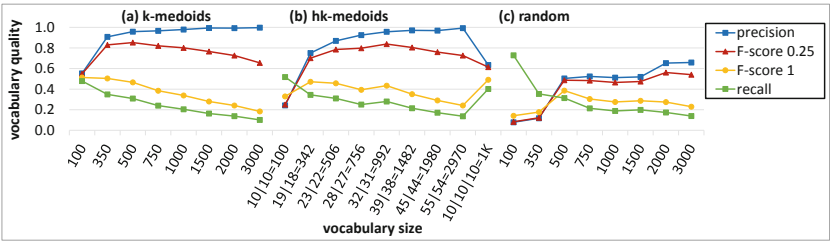

Fig. 3. Vocabulary quality in relation to vocabulary method and varying vocabulary size: (a) *k*-medoids, (b) *hk*-medoids and (c) random pivot selection.

costly since the distances of all pivot-object pairs need to be computed in each iteration. The *hierarchical k-medoids* (*hk-medoids*) seeks the pivots by recursive application of *k*-medoids, which allows using much smaller values of k in each iteration to create a vocabulary of the same size. The pivots for the next level are always selected from the parental cell, so the data locality is preserved. We use a constant number of pivots per level and similar pivot numbers across levels. For example, the set-up 39|38 denotes 39 pivots in the root level and 38 pivots in the second level, which creates 1,482 cells. Finally, we also try a *random* selection of pivots where a pivot too close to another one is omitted. This is the most efficient approach which is known to perform well in permutation-based indexes [15].

Experimental Evaluation. Using the three algorithms, we create vocabularies of sizes ranging from 100 up to 3,000 MWs, and compare their quality. The results presented in Fig. 3 are averages over five runs. In general, the higher the precision is the more pivots are used, and vice-versa for the recall. A good vocabulary is prepared by techniques choosing the pivots in correspondence to the distribution of segments, thus the random selection should be rejected, since its precision is low. Focusing on the vocabulary size, the $F_{0.25}$ score that favors precision guides us to pick the *k*-medoids with 350 or 500 pivots and the *hk*-medoids of the 32|31 breakdown. In the F_1 score, the optimum is 100 or 350 pivots by *k*-medoids, and 19|18 or 10|10|10 by *hk*-medoids.

The *k*-medoids with 350 pivots has been identified as the most promising hard quantization method, therefore we use it in the following trials exclusively. We also experimented with the best settings of the other algorithms and obtained analogous trends, so we do not include them.

4.2 Soft Quantization

Secondly, it is vital for the soft quantization to assign additional MW elements of neighboring cells only to the segments that are close to the cell borders. We limit such closeness by the distance D and bound the number of MW elements to the maximum number K. We study the influence of D and K on the quality measures, which should show that the border problem is reduced.

Implementation. The distance of a segment s located in the Voronoi cell of pivot p_1 to the borderline of the cell of p_2 is estimated as $\frac{|DTW(p_1,s)-DTW(p_2,s)|}{2}$. We gradually check all pivots p_i and expand the segment's MW with the MW element p_i until the estimated distance exceeds D. The value of D must be smaller than the similarity threshold T_{sim} discussed in Sect. 3.3, since it identifies objects that should be assigned the same MW. There can be many neighboring cells, so we constrain the MW elements to the K closest ones.

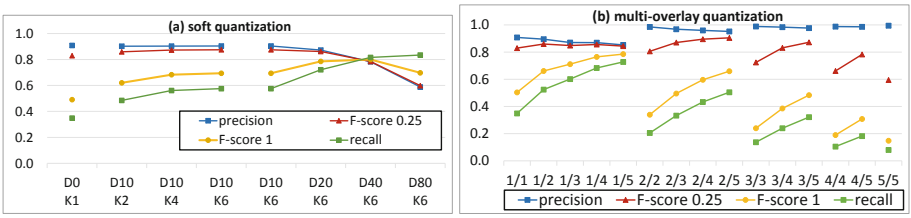

Fig. 4. Quality of vocabulary in relation to vocabulary construction method: (a) soft quantization for 350 pivots: varying K for $D10$ and varying D for $K6$; (b) multi-overlay quantization: 1 to 5 overlays with 350 pivots each, varying number of matching overlays.

Experimental Evaluation. We vary the values of D from 10 ($1/8 \cdot T_{sim}$) to 80 (T_{sim}), and K from 2 to 6. The relevant results are shown in Fig. 4a. Increasing K for a small D ($D10$, $K2$–6) leads to improved recall and nearly constant precision. On the other hand, multiplying D ($D10$–80, $K6$) produces extensive MWs, which reduces the border problem (recall is boosted), but it negatively effects precision. For the classification task ($F_{0.25}$ score), $D10$, $K6$ and $D20$, $K6$ are the best, while $D40$, $K6$ is the optimum for the search (F_1 score).

4.3 Multi-overlay Quantization

Thirdly, independent sets of pivots are likely to provide different Voronoi partitionings, thus increasing a chance of similar segments to share the same cell. We create up to 5 overlays and vary the number of overlays required to agree.

Implementation. Since the k-medoids algorithm provides a locally optimal solution, we ran it five times to obtain different sets of 350 pivots for the multi-overlay quantization. Noticeably, the quality of hard vocabularies created from individual sets of pivots differs up to 5% in both the F scores.

Experimental Evaluation. In Fig. 4b, we present the results for all combinations of the five overlays, where the notation m/n refers to the m-out-of-n matching function. The combination 1/1 corresponds to hard quantization. When we fix m to 1 and add more overlays, the border problem is reduced, as witnessed

by a major improvement of recall and only a marginal drop in precision. Similar trends can be observed also for higher values of m, but the actual values of recall get lower when we require more overlays to agree. The most restrictive combination 5/5 requires all overlays to meet and performs similarly to the hard quantization with more than 3,000 pivots (see Fig. 3a). The best $F_{0.25}$ score is for the 2/5 setup and the best F_1 score is for the 1/5 setting.

4.4 Discussion

By thorough experimentation, we have observed that the k-medoids clustering is the best hard quantization method but its quality can still be significantly improved by the soft and multi-overlay principles. The suppression of the border problem is mainly attributed to the increased number of correctly matched segment pairs (true positives) by both these principles. Although some new false positives are introduced, they decrease the overall precision only marginally.

Since the k-medoids clustering has high computation complexity, we have also considered cheaper techniques, i.e. the random clustering, with the soft and multi-overlay approach. However, the experimental results were not much competitive, so the k-medoids still remains a reasonable choice for quantization.

Our vocabulary construction techniques are universal, but the created vocabulary is clearly data-dependent. Since our evaluation data are relatively small (28,104 segments), the optimal vocabulary size is 350 MWs for the hard quantization. For larger and more diverse data, we expect the quality measures to recommend a larger vocabulary.

Finally, a successful application also requires fast access to the data, which calls for indexes. The hard vocabulary can be directly organized in an inverted file. The soft-assigned vocabulary just expands the query, so the inverted file is sought multiple times (proportional to the number of MW elements in the query). The multi-overlay vocabulary can be managed in separate search indexes (one per overlay) and the query results merged to compute the m-out-of-n matching.

5 Motion Words in Applications

In this section, we experimentally verify that: (i) the MW representation preserves important characteristics of complex 3D skeleton data and causes no drop in application effectiveness (Sect. 5.2), and (ii) the vocabulary quality measures well approximate the usefulness of different vocabularies in applications. Both these aspects are evaluated in context of two applications: the *action classification* that aims at recognizing the correct class of a given action using a kNN classifier, and the *action search* where the goal is to retrieve all actions relevant to a query, i.e. the actions belonging to the same class as the query.

5.1 Evaluation Methodology of Classification/Search Applications

The input for both classification and search applications is the dataset of $2,345$ synthetically-segmented actions discussed in Sect. 3.3. On average, each action

is transformed into a sequence of 12 MWs. To compare two MW sequences, we again adopt the DTW sequence alignment function. Realize that the MW matching function inside DTW deals with the spatial variability of short segments, whereas DTW considers the temporal dimension of the whole actions.

Both applications are evaluated on the basis of k-nearest neighbor (kNN) queries. We use the standard leave-one-out approach to evaluate $2,345$ kNN queries in a sequential way by computing the distance between the specific query action and each of the remaining dataset actions. For the classification task, we fix k to 4 and apply a 4NN classifier (similar to the classifier proposed in [19]). We measure the application *effectiveness* as the average classification accuracy over all $2,345$ queries. For the search task, the value of k is adjusted for each query individually based on the number of available actions belonging to the same class as the query action. Such adaptive value of k allows us to focus on recall as well as precision. The effectiveness of the search application is then determined as the average recall over all the queries. Note that the recall is always the same as the precision in the search task with the adaptive value of k.

5.2 Usefulness and Efficiency of MWs

We quantify the usefulness of the MW concept by evaluating application effectiveness with different vocabularies and comparing it to the baseline case that uses no quantization (i.e. the action segments are represented by original 3D skeleton data). The most interesting results are summarized in Table 1.

For classification, we observe that the baseline case achieves the effectiveness of 77.70%. Worse results have been expected for any MW quantization due to the dimensionality reduction of the original segment data. A standard hard quantization – the single-level k-medoids – indeed achieves the worst result (74.97%). Surprisingly, the soft-assignment $D10$, $K6$ vocabulary reaches basically the same effectiveness (77.61%) as the baseline, and the 2/5 multi-overlay quantization is actually better (80.30%). Thus, the best MW vocabulary not only preserves important motion characteristics but also aggregates many tiny variations in joint positions that confuse DTW on raw 3D skeleton data (the baseline case).

A similar trend can be observed on the search application where the hard quantization has the worst result too. As the recall is very important for the search task, the 1/5 multi-overlay vocabulary is now the best candidate that also outperforms the baseline case. Compared to the state-of-the art approaches [3,11] that employ the hard quantization, the proposed generalized MWs reach much better effectiveness, e.g., about 25% higher recall in the search application (increase from 44.21% to 55.62%).

From the performance point of view, it takes almost 1.5 h to evaluate all the $2,345$ kNN queries with the baseline segment representation. Using any of the MW representations, the evaluation finishes in 30 s, which is an improvement by two orders of magnitude.

5.3 Concordance of Vocabulary Quality and Usefulness

Remember that in Sect. 3.3 we proposed to quantify the vocabulary quality by the F_β score, where the parameter β is set according to the precision/recall preference of the target application. For classification and search, we proposed to use $F_{0.25}$ and F_1, respectively. Here, we verify whether such F_β scores correspond to the actual usefulness of individual vocabularies. To do so, we apply the vocabularies discussed in Sects. 4.1, 4.2 and 4.3 to our real-life applications and measure the application effectiveness.

The results in Fig. 5 confirm that the estimated quality of vocabularies (red line in Fig. 5a and yellow line in Fig. 5b) shares the same *trend* with the actual vocabulary usefulness measured by the real classification (grey dashed line) and search (grey solid line) effectiveness. Therefore, the F_β score can be used for selecting the most suitable vocabulary for a given application, instead of a tedious and costly experimenting with all candidate vocabularies within the application.

Table 1. Effectiveness of classification and search applications with different segment representations (MW representations use the best-ranked vocabularies with 350 pivots).

Application	Segments as raw 3D skel. data	MW segment representations					
		Hard quant.	Soft assignment		Multi overlays		
			$D10$, $K6$	$D20$, $K6$	1/4	1/5	2/5
Classification	77.70%	74.97%	77.61%	76.42%	76.33%	75.69%	**80.30%**
Search	53.84%	44.21%	47.92%	50.26%	54.97%	**55.62%**	50.29%

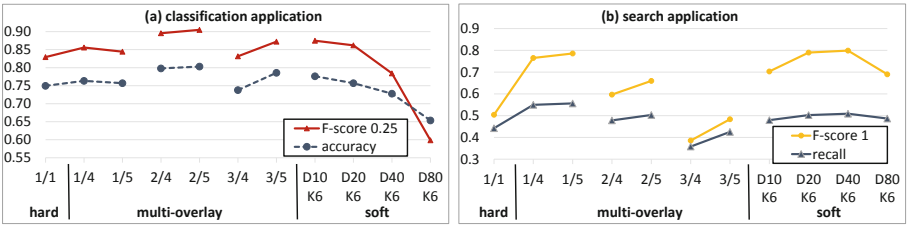

Fig. 5. Comparison of F_β score and actual effectiveness (accuracy, recall) for selected vocabularies in the (a) classification and (b) search applications. (Color figure online)

6 Conclusions

This paper studies the possibility of transforming unlabeled 3D skeleton data into text-like representations that allow efficient processing. In particular, we

focused on quantizing short synthetic motion segments into compact, similarity-preserving motion words (MWs). In contrast to existing works on motion quantization, we recognize the border problem and try to minimize it using the soft-assignment and multi-overlay partitioning principles. We also proposed a methodology for application-independent evaluation of the MW vocabulary quality. The experimental results on two real-world motion processing tasks confirm that we are able to construct MW vocabularies which preserve or even slightly increase application effectiveness and significantly improve processing efficiency.

We believe that these achievements open new possibilities for efficient analysis of 3D motion data. In the future, we will study more thoroughly the preparation and preprocessing of the short segments, and develop scalable indexing and search algorithms for the MW data. In particular, we plan to enrich the segmentation process to include several segment sizes, which should help us deal with possible speed variability of semantically related motions. Before the actual quantization, the segments can be replaced by characteristic features extracted, e.g., by state-of-the-art neural networks. To index and search MW sequences, we intend to employ the shingling technique and adapted inverted files.

Acknowledgements. This research has been supported by the GACR project No. GA19-02033S.

References

1. Ahmad, Z., Khan, N.M.: Towards improved human action recognition using convolutional neural networks and multimodal fusion of depth and inertial sensor data. In: 20th International Symposium on Multimedia (ISM), pp. 223–230. IEEE (2018)
2. Alldieck, T., Magnor, M.A., Bhatnagar, B.L., Theobalt, C., Pons-Moll, G.: Learning to reconstruct people in clothing from a single RGB camera. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE (2019)
3. Aristidou, A., Cohen-Or, D., Hodgins, J.K., Chrysanthou, Y., Shamir, A.: Deep motifs and motion signatures. ACM Trans. Graph. **37**(6), 187:1–187:13 (2018). https://doi.org/10.1145/3272127.3275038
4. Butepage, J., Black, M.J., Kragic, D., Kjellstrom, H.: Deep representation learning for human motion prediction and classification. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 6158–6166. IEEE (2017)
5. Demuth, B., Röder, T., Müller, M., Eberhardt, B.: An information retrieval system for motion capture data. In: Lalmas, M., MacFarlane, A., Rüger, S., Tombros, A., Tsikrika, T., Yavlinsky, A. (eds.) ECIR 2006. LNCS, vol. 3936, pp. 373–384. Springer, Heidelberg (2006). https://doi.org/10.1007/11735106_33
6. Dohnal, V., Homola, T., Zezula, P.: MDPV: metric distance permutation vocabulary. Inf. Retr. J. **18**(1), 51–72 (2015)
7. Kabary, I.A., Schuldt, H.: Using hand gestures for specifying motion queries in sketch-based video retrieval. In: de Rijke, M., et al. (eds.) ECIR 2014. LNCS, vol. 8416, pp. 733–736. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-06028-6_84
8. Krüger, B., Vögele, A., Willig, T., Yao, A., Klein, R., Weber, A.: Efficient unsupervised temporal segmentation of motion data. IEEE Trans. Multimed. **19**(4), 797–812 (2017)

9. Liu, B., Cai, H., Ju, Z., Liu, H.: RGB-D sensing based human action and interaction analysis: a survey. Pattern Recogn. **94**, 1–12 (2019)
10. Liu, J., Wang, G., Duan, L., Hu, P., Kot, A.C.: Skeleton based human action recognition with global context-aware attention LSTM networks. IEEE Trans. Image Process. **27**(4), 1586–1599 (2018)
11. Liu, X., He, G., Peng, S., Cheung, Y., Tang, Y.Y.: Efficient human motion retrieval via temporal adjacent bag of words and discriminative neighborhood preserving dictionary learning. IEEE Trans. Hum.-Mach. Syst. **47**(6), 763–776 (2017). https://doi.org/10.1109/THMS.2017.2675959
12. Manning, C.D., Raghavan, P., Schütze, H.: Introduction to Information Retrieval. Cambridge University Press, Cambridge (2008). https://doi.org/10.1017/CBO9780511809071
13. Müller, M., Röder, T., Clausen, M., Eberhardt, B., Krüger, B., Weber, A.: Documentation Mocap Database HDM05. Technical Report CG-2007-2, Universität Bonn (2007)
14. Nistér, D., Stewénius, H.: Scalable recognition with a vocabulary tree. In: International Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2161–2168 (2006)
15. Novak, D., Zezula, P.: PPP-codes for large-scale similarity searching. In: Hameurlain, A., Küng, J., Wagner, R., Decker, H., Lhotska, L., Link, S. (eds.) Transactions on Large-Scale Data- and Knowledge-Centered Systems XXIV. LNCS, vol. 9510, pp. 61–87. Springer, Heidelberg (2016). https://doi.org/10.1007/978-3-662-49214-7_2
16. Philbin, J., Chum, O., Isard, M., Sivic, J., Zisserman, A.: Object retrieval with large vocabularies and fast spatial matching. In: International Conference on Computer Vision and Pattern Recognition (CVPR) (2007)
17. Sedmidubsky, J., Elias, P., Zezula, P.: Effective and efficient similarity searching in motion capture data. Multimed. Tools Appl. **77**(10), 12073–12094 (2017). https://doi.org/10.1007/s11042-017-4859-7
18. Sedmidubsky, J., Elias, P., Zezula, P.: Searching for variable-speed motions in long sequences of motion capture data. Inf. Syst. **80**, 148–158 (2019). https://doi.org/10.1016/j.is.2018.04.002
19. Sedmidubsky, J., Zezula, P.: Probabilistic classification of skeleton sequences. In: Hartmann, S., Ma, H., Hameurlain, A., Pernul, G., Wagner, R.R. (eds.) DEXA 2018. LNCS, vol. 11030, pp. 50–65. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-98812-2_4
20. Sedmidubsky, J., Zezula, P.: Augmenting spatio-temporal human motion data for effective 3D action recognition. In: 21st IEEE International Symposium on Multimedia (ISM), pp. 204–207. IEEE Computer Society (2019). https://doi.org/10.1109/ISM.2019.00044
21. Sivic, J., Zisserman, A.: Video Google: a text retrieval approach to object matching in videos. In: 9th International Conference on Computer Vision (ICCV), pp. 1470–1477. IEEE (2003)
22. Zhao, R., Wang, K., Su, H., Ji, Q.: Bayesian graph convolution LSTM for skeleton based action recognition. In: IEEE International Conference on Computer Vision (ICCV), pp. 6882–6892. IEEE (2019)

23. Zheng, W., Li, L., Zhang, Z., Huang, Y., Wang, L.: Relational network for skeleton-based action recognition. In: International Conference on Multimedia and Expo (ICME), pp. 826–831. IEEE (2019)
24. Zhu, H., Long, M., Wang, J., Cao, Y.: Deep hashing network for efficient similarity retrieval. In: 30th Conference on Artificial Intelligence (AAAI), pp. 2415–2421. AAAI Press (2016)