Partial Alignment of Time Series for Action and Activity Prediction

Victoria Manousaki[1](✉) and Antonis Argyros[1,2]

[1] Computer Science Department, University of Crete, Heraklion, Greece
{vmanous,argyros}@ics.forth.gr
[2] Institute of Computer Science, Foundation for Research and Technology - Hellas (FORTH), Heraklion, Greece

Abstract. The temporal alignment of two complete action/activity sequences has been the focus of interest in many research works. However, the problem of partially aligning an incomplete sequence to a complete one has not been sufficiently explored. Very effective alignment algorithms such as Dynamic Time Warping (DTW) and Soft Dynamic Time Warping (S-DTW) are not capable of handling incomplete sequences. To overcome this limitation the Open-End DTW (OE-DTW) and the Open-Begin-End DTW (OBE-DTW) algorithms were introduced. The OE-DTW has the capability to align sequences with common begin points but unknown ending points, while the OBE-DTW has the ability to align unsegmented sequences. We focus on two new alignment algorithms, namely the Open-End Soft DTW (OE-S-DTW) and the Open-Begin-End Soft DTW (OBE-S-DTW) which combine the partial alignment capabilities of OE-DTW and OBE-DTW with those of Soft DTW (S-DTW). Specifically, these algorithms have the segregational capabilities of DTW combined with the soft-minimum operator of the S-DTW algorithm that results in improved, differentiable alignment in the case of continuous, unsegmented actions/activities. The developed algorithms are well-suited tools for addressing the problem of action prediction. By properly matching and aligning an on-going, incomplete action/activity sequence to prototype, complete ones, we may gain insight in what comes next in the on-going action/activity. The proposed algorithms are evaluated on the MHAD, MHAD101-v/-s, MSR Daily Activities and CAD-120 datasets and are shown to outperform relevant state of the art approaches.

Keywords: Segregational soft dynamic time warping · Temporal alignment · Action prediction · Activity prediction · Duration prognosis · Graphs

1 Introduction

The visual observations of different executions of the same activity may vary considerably when performed by different or even the same subject. Variations are further attributed to changes in the environment, the manipulated objects the observation viewpoints and several other causes. A video showing an action/activity execution can be represented as a time series of frames, each of which is represented in a point in a multidimensional feature space. Given time-series representations of certain

A. A. de Sousa et al. (Eds.): VISIGRAPP 2022, CCIS 1815, pp. 89–107, 2023.
https://doi.org/10.1007/978-3-031-45725-8_5

actions/activities, temporal alignment algorithms have been used for matching different executions in time in order to support the solution of problems such as action quality assessment [25], action co-segmentation [20], fine-grained frame retrieval [11] etc.

The Dynamic Time Warping (DTW) [26] algorithm is a commonly employed time series alignment algorithm. DTW requires that the test and reference action/activity executions need to be trimmed in order to be aligned. The alignment of two sequences is achieved by finding the minimum-cost warping path between them. The alignment path provides a one-to-one alignment between the frames of the two sequences. The Soft Dynamic Time Warping (S-DTW) [7] algorithm is a variant of the baseline DTW algorithm which finds their alignment by calculating the soft-minimum cost of all possible path-based alignments.

Both DTW and S-DTW are able to align only trimmed/segmented sequences. When the starting and ending points of the time series are known, these algorithms can provide meaningful alignments. But there are cases where the executions are not trimmed or there can be prefix or suffix noise. In such scenarios, the DTW and S-DTW algorithms fail to achieve satisfactory alignment of the input sequences. Such situations may occur, for example, when an ongoing, incomplete action needs to be matched with a completed, reference one, or when the two actions to be matched occur between other actions. Such untrimmed/unsegmented inputs can be aligned by two other DTW variants, the Open-End DTW [28] and Open-Begin-End DTW [28]. The OE-DTW is designed for aligning two sequences with known starting point but have unknown ending points. On the contrary, the OBE-DTW algorithm is not anchored to either points thus is able to align the unsegmented input without any boundary constraints.

In this paper we explore the use of two new S-DTW variants, the Open-End Soft DTW (OE-S-DTW) and Open-Begin-End Soft DTW (OBE-S-DTW). These alignment algorithms were proposed by Manousaki et al. [16] for the problem of aligning segmented and unsegmented action executions. These two variants combine the merits of OE-DTW and OBE-DTW to that of S-DTW. Specifically, similarly to OE-DTW and OBE-DTW they have reduced requirements on the knowledge of the sequence endpoints (i.e., relaxed sequence boundary constraints) and similarly to S-DTW they are differentiable. Thus, these DTW variants can be used for aligning unsegmented sequences and also as a loss function for training deep neural networks. Currently, they have been used in the framework of Manousaki et al. [16] as a tool for solving the problem of action prediction.

In this paper we build on top of the OE-S-DTW and OBE-S-DTW algorithms and we provide the following additional investigations and contributions:

- We extend the experimental evaluation of the framework of Manousaki et al. [16] by providing additional evaluations of the OE-S-DTW and OBE-S-DTW alignment algorithms on the problem of short-term human action prediction and action duration prognosis in standard datasets and in comparison to existing state of the art methods. More specifically, we present results on the duration prediction problem on the MSR Daily Activities [29] and CAD120 [13] datasets. We also evaluate the efficiency of the alignment algorithms on a new challenging action prediction scenario involving a reversed observation of the input actions.

– Differently from [16] where the evaluation of the OBE-S-DTW and OE-S-DTW algorithms is focused on the prediction of actions, we evaluate these algorithms using activities, too, that are composed of long (and therefore more complex) sequences of actions. In that direction, we evaluate OBE-S-DTW and OE-S-DTW not only using trimmed actions of the MHAD101-s/-v [22] and CAD120 [13] datasets, but also activities of the CAD120 dataset.
– In [16], the OE-S-DTW and OBE-S-DTW algorithms have been used for action prediction in a closest match-based action prediction framework. In this paper we extend the experimental evaluation of these two variants by utilizing and comparing them also in a graph-based framework for action and activity prediction [18].

2 Related Work

The temporal alignment of sequences is a problem that has been explored for many years and remains of interest until today. A classical approach is the Dynamic Time Warping [26] algorithm which is capable of aligning segmented sequences by finding the minimum-cost warping path between them. The warping path is calculated upon the distance matrix which contains all the frame-wise distances between the two sequences to be aligned. The DTW score is based on the summation of all path-related values in the distance matrix. The DTW algorithm poses boundary constraints on the warping path which means that the sequences to be aligned must start and end at known frames i.e. the first frame of the first sequence will be matched to the first frame of the second sequence. DTW has been used in a variety of problems such as action cosegmentation [22], representation learning [10], etc.

The boundary constraints of DTW have been relaxed by the work of Tormene et al. [28] who proposed the Open-End DTW (OE-DTW) algorithm. The OE-DTW variant is capable of aligning sequences that have a known common start point but unknown endpoints. This relaxation of the endpoint constraint is useful when the sequences to be matched are partially observed or when other actions appear after the end of the sequence. The OE-DTW score is provided by the summation of all values of the minimum-cost alignment path. The difference to the DTW algorithm is that the alignment path that starts at the top-left point of the distance matrix should not necessarily end at the bottom-right cell of that matrix, thus permitting a certain sequence to match with a part of a reference one. OE-DTW has been used to compare motion curves for the rehabilitation of post-stroke patients [27] as well as for the evaluation of the user's motion in visual observations of humans with Kinect [32].

Tormene et al. [28] also proposed the Open-Begin-End (OBE-DTW) [28] that aligns two unsegmented sequences, i.e., two sequences of unknown starting and ending points. The matching path defined by OBE-DTW does not necessarily have to start and end at the top-left and bottom-right cells of the distance matrix. OBE-DTW has been used in many contexts for unsegmented sequence alignment e.g., for the problem of classifying motion from depth cameras [12].

While the DTW algorithm considers the minimum-cost alignment path of the sequences, the Soft Dynamic Time Warping (S-DTW) [7] variant considers the soft-minimum of the distribution of all costs spanned by all possible alignments between two segmented sequences. This alignment score contains the summation of all path-based values. The S-DTW algorithm has been used by [11] as temporal alignment loss for training a neural network to learn better video representations. The differentiable alignment of S-DTW has also been used by Chang et al. [6] for the alignment and segmentation of actions by using the videos and the transcripts of the actions.

Segmental DTW [23] seeks for the minimum-cost sub-sequence alignment of pairs of unsegmented inputs. Segmental DTW decomposes the distance matrix in sets of overlapping areas and finds the local end-to-end alignments in these areas resulting in sub-sequence matching. Segmental DTW has been used in the context of action co-segmentation [19] in motion-capture data or video between pairs of actions for the detecting of commonalities of varying length, different actors, etc.

The Ordered Temporal Alignment Module (OTAM) [5] aligns segmented sequences of fixed length by using the soft-minimum operator and calculating all possible path-based alignments. The alignment score is given by aligning the sequences end-to-end using S-DTW, while the alignment path is retrieved by an OBE-DTW approximation. Cao et al. [5] used the OTAM alignment for few-shot video classification of fixed-length trimmed videos.

Finally, the Drop Dynamic Time Warping (Drop-DTW) [8] algorithm is a variant of DTW based on images where outliers are dropped during the alignment of sequences. Differently from OBE-DTW where the unrelated parts can be at the prefix or the suffix of an action, this DTW approximation is very useful in cases where the sequences to be aligned have unrelated parts anywhere inside the sequences. By eliminating all the irrelevant parts Drop-DTW results in more meaningful alignments.

3 Temporal Alignment of Action/Activity Sequences

Let $Q = (q_1, \dots q_l) \in \mathbb{R}^{n \times l}$ represent a test action/activity sequence that needs to be aligned with a reference sequence $Y = (y_1, \dots, y_m) \in \mathbb{R}^{n \times m}$. The distance matrix $D(Q,Y) = [d(q_i, y_i)]_{ij} \in \mathbb{R}^{l \times m}$ contains all Euclidean pair-wise frame distances $d(q,y)$ of frames q and y. The cumulative matrix that is based on D and represents all path-based alignments P of Q and Y, is denoted as $C(Q,Y) = \{\langle p, D(Q,Y)\rangle, p \in P_{l,m}\}$ where P represents all the alignments connecting the upper-left to the lower-right of the distance matrix. Using this notation, we proceed with presenting the employed action/activity sequence alignment methods.

3.1 Alignment Methods - Segmented Sequences

Dynamic Time Warping (DTW) [26]**:** The DTW algorithm aligns two sequences in their entirety by finding their minimum alignment cost. The distance matrix $D(Q,Y)$ is used to create the cumulative matrix $C(Q,Y)$. The minimum alignment cost is provided at the bottom-right cell of the cumulative matrix. Due to the variability of the sequence sizes the alignment score needs to be normalized by the length of the test sequence. The alignment cost provided by DTW is defined as:

$$DTW(Q,Y) = min_{p \in P} C(Q,Y). \quad (1)$$

Soft Dynamic Time Warping (S-DTW) [7]**:** The DTW algorithm has some limitations such as not being differentiable and getting stuck in local minima due to the min operator. S-DTW is a powerful extension of the original DTW algorithm which is differentiable and introduces a smoothing parameter γ that can help avoid local minima depending on the values of the smoothing parameter. In order to do so, the S-DTW takes into account all possible alignment paths contrary to DTW which calculates only the minimum cost alignment path. The limitations are alleviated by changing the minimum operator with the soft-minimum operator (see Eq. (3)). The cumulative matrix $C(x_i, y_j) = D(x_i, y_j) + min^{\gamma}(C(x_{i-1}, y_j), C(x_{i-1}, y_{j-1}), C(x_i, y_{j-1}))$ is calculated as in DTW by allowing horizontal, diagonal and vertical moves. The cumulative matrix is padded at the top with a row and at the left with a column so that $C_{i,0} = C_{0,j} = \infty$ for all $i, j \neq 0$ and $C_{0,0} = 0$. The S-DTW alignment cost between two sequences is defined as:

$$SDTW_{\gamma}(X, Y) = min^{\gamma}_{p \in P}\, C(X, Y), \tag{2}$$

with

$$\min \gamma(p_1, \ldots, p_k) = \begin{cases} \min_{i \leq k} p_i, & \gamma = 0, \\ -\gamma \log \sum_{i=1}^{k} e^{p_i/\gamma} & \gamma > 0, \end{cases} \tag{3}$$

where $\gamma \geq 0$ is a smoothing hyper-parameter. When γ is equal to zero, the DTW score is calculated.

3.2 Alignment Methods - Unsegmented Sequences

Open-End Dynamic Time Warping (OE-DTW) [28]**:** DTW [26] is designed and used for aligning two sequences from start to finish. When the sequences have unknown end points, DTW produces poor alignment results. A DTW variant was created to address this problem called OE-DTW. The cumulative matrix C is calculated as

$$C(x_i, y_j) = D(x_i, y_j) + min(C(x_{i-1}, y_j), C(x_{i-1}, y_{j-1}), C(x_{i-1}, y_{j-2})). \tag{4}$$

Essentially, the alignment cost becomes the minimum value of the last row of the cumulative matrix. As explained earlier, the values need to be normalized by the size of the test sequence. Thus, the alignment cost of OE-DTW is defined as:

$$OE\text{-}DTW(X, Y) = min_{j=1,\ldots,m} DTW(X, Y_j). \tag{5}$$

Open-Begin-End Dynamic Time Warping (OBE-DTW) [28]**:** There can be sequences that do not share the same beginning and ending or one sequence appears anywhere inside the other. To solve this alignment problem a variant of the DTW algorithm was created namely OBE-DTW. To calculate the alignment cost based on this variant, a row with zero values is appended at the beginning of the distance matrix and the computations are performed as in OE-DTW. The computed cumulative matrix is denoted as $C'(X, Y)$ and the alignment cost is the minimum value of the last row which were previously normalized by the length of the test sequence. The back-tracing

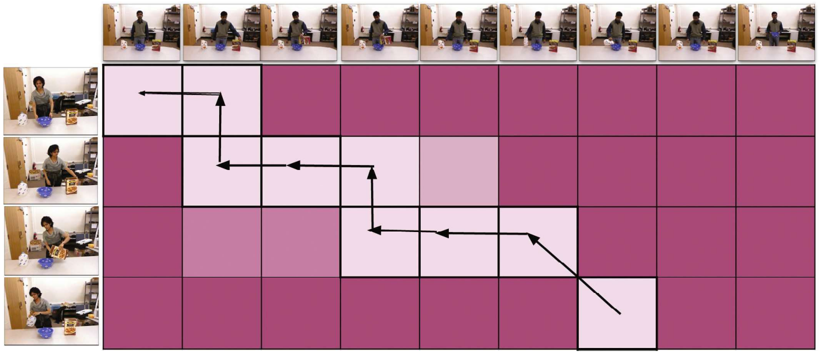

Fig. 1. Graphical illustration of the OE-S-DTW algorithm. On the horizontal axis we can observe a man performing an activity. On the vertical axis a woman is performing the same activity which is not yet completed. The light pink boxes represent the possible alignment paths while the black arrows represent a possible path. The two sequences share the same starting point but end at different points. The OE-S-DTW algorithm is able to match the partially observed activity with a part of the completely observed one. (Color figure online)

of the minimum-cost path starts from the minimum value of the last row and ends at the first zero-valued row. The OBE-DTW alignment cost is denoted as:

$$OBE\text{-}DTW(X, Y) = min_{j=1,\ldots,m} C'(X, Y_j). \tag{6}$$

Open-End Soft DTW (OE-S-DTW) [16]**:** The OE-S-DTW is a newly proposed algorithm that combines the merits of the OE-DTW and S-DTW algorithms. This is a differentiable alignment algorithm that can align sequences that share the same beginning but do not share the same ending points. The distance matrix is calculated by using the pairwise distances of the reference and test sequences X and Y, respectively. The cumulative matrix is calculated as in S-DTW by using the min^{γ} operator as follows:

$$C(x_i, y_j) = D(x_i, y_j) + min^{\gamma}(C(x_{i-1}, y_j), C(x_{i-1}, y_{j-1}), C(x_i, y_{j-1})). \tag{7}$$

The scores at the last row are normalized by the query's size and the cost of alignment is the minimum of the last row. As in OE-DTW, the alignment path may terminate at any point of the last row of the cumulative matrix. Finally, the gradient is calculated from that point backwards to the common start point to find the alignment between the two sequences. The final OE-S-DTW score is also normalised by the size of the matched reference as follows:

$$OE\text{-}S\text{-}DTW(X, Y) = min^{\gamma}_{j=1,\ldots,m} SDTW_{\gamma}(X, Y_j). \tag{8}$$

A graphical illustration of the OE-S-DTW algorithm is presented in Fig. 1.

Open-Begin-End Soft Dynamic Time Warping (OBE-S-DTW) [16]**:** OBE-S-DTW is an alignment algorithm that combines the beneficial properties of the OBE-DTW and S-DTW. The distance matrix $D'(X, Y)$ is created by appending the distance matrix

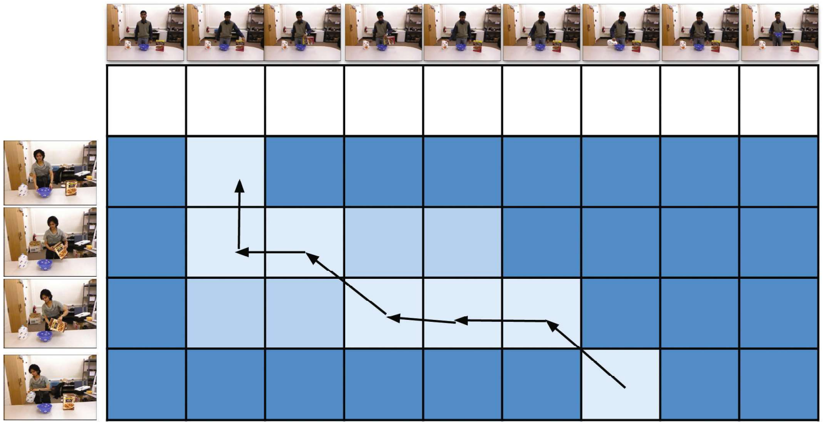

Fig. 2. Graphical illustration of the OBE-S-DTW algorithm. The activity illustrated at the left (rows) matches a part of the activity illustrated on the top (columns). At the top a zero-valued row is added. The light blue boxes represent all possible alignments while the black arrows show a possible warping path. (Color figure online)

$D(X, Y)$ with a row of zero values at the beginning. Following that, the cumulative matrix denoted as C' is calculated by using the soft-minimum operator as follows:

$$C'(x_i, y_j) = D'(x_i, y_j) + min^{\gamma}(C'(x_{i-1}, y_j), C'(x_{i-1}, y_{j-1}), C'(x_i, y_{j-1})). \quad (9)$$

The last row of the cumulative matrix is normalized by the test sequence size and the alignment cost is found at the minimum value of the last row. Then, the gradient is computed from that point towards the zero-valued row. The size of the matched reference corresponds to that range. The calculated gradient gives the possibility to consider all possible alignments (see Fig. 2). Once the alignment path is obtained, we normalize the alignment cost with the size of the matching part of the reference sequence. The OBE-S-DTW alignment cost is defined as:

$$OBE\text{-}S\text{-}DTW(X, Y) = min^{\gamma}_{j=1,\ldots,m} C'(X, Y_j). \quad (10)$$

An illustration of the OBE-S-DTW algorithm is provided in Fig. 2.

3.3 Action and Activity Prediction

Alignment-Based Action and Activity Prediction. The action/activity prediction problem is denoted as the problem of predicting the correct label of a partially observed action/activity execution. Our intention is to solve the problem of action and activity prediction by matching reference and test action executions. To do so, prototype executions are aligned with incomplete executions and the inference of the label is done by reporting the label of the closest matching prototype execution to the incomplete one as in Manousaki et al. [17]. More specifically, we fuse the human and object representations by calculating the weighted sum of the respective distance matrices. If the actions

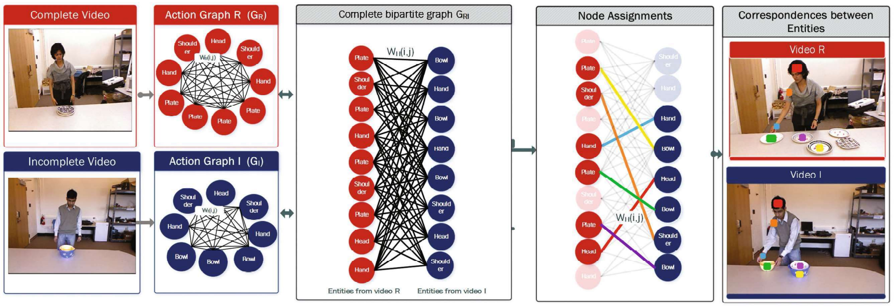

Fig. 3. Illustration of the graph-based framework for action/activity prediction. We assume a complete video (reference) and an incomplete/partially observed (test) one. First, the fully connected graphs of each video are created based on the video entities. On the basis of these graphs, a complete bipartite graph between the action graphs is constructed where the new edges describe the semantic and motion dissimilarity between the nodes of the two graphs. By calculating the Graph Edit Distance (GED), we are able to quantify their dissimilarity and to correspond nodes between the two original action graphs.

do not contain objects then only the human pose representations are used. The weights depend on the manipulated/closest object class. Further details on how to transform an input video into a multidimensional times series can be found in [17].

Having represented the action/activity executions as multi-dimensional time series, our goal is to align the Z prototype actions/activities with labels $L(P_i)$ with the incomplete action/activity A and infer the label $L(A)$. More specifically, a set of prototype actions/activities will be aligned temporally with the incomplete action/activity. The minimum alignment cost denoted as $MAC(P, A)$ will determine which prototype action/activity P^i has the minimum alignment cost with A. Formally,

$$L(A) = L\left(arg\ min_{1 \leq i \leq Z}\left(MAC(A, P^i)\right)\right). \tag{11}$$

Graph-Based Action and Activity Prediction. The OE-S-DTW and OBE-S-DTW alignment algorithms can be exploited in another, graph-based framework for action and activity prediction. Specifically, we explore the work of [18] where they use graphs to solve the activity and next-active-object prediction problem. According to this approach, each activity is represented as a graph. The nodes of the graph represent the entities that participate in an activity, i.e., the joints of the human body of the acting person and the visible scene objects. The nodes hold the semantic label of the corresponding entity and its motion information (2D/3D trajectory). The semantic dissimilarity of the entities as well their motion dissimilarity are encoded on the edges of the graph. More specifically, the semantic dissimilarity of two entities is calculated based on the Wordnet [9] lexical database and the Natural Language Toolkit [14] to calculate the Wu-Palmer distance metric [31]. The motion dissimilarity is calculated using either the OE-S-DTW or the OBE-S-DTW algorithms based on the 2D/3D trajectories of the corresponding entities. Thus, the edges carry the weighted sum of the motion and the semantic dissimilarity that they connect.

Then, matching and aligning actions amounts to matching the graphs representing them. To do so, a complete bipartite graph is created between the graph nodes of a prototype and an incomplete video. As illustrated in Fig. 3, in this new graph the edges hold the semantic and motion dissimilarities of the connected entities between the two graphs. The dissimilarity of these graphs is calculated using the Graph Edit Distance (GED) [1]. In order to disregard action-irrelevant objects, the BP-GED is normalized by the number of pairs of matched objects (MO). Thus, the dissimilarity $D(G_I, G_R)$ of the graph of incomplete video G_I and the graph of the reference video G_R is defined as:

$$D(G_I, G_R) = BP\text{-}GED(G_I, G_R)/MO. \tag{12}$$

It is noted that while [17] can handle up to only one object per activity, this graph-based approach does not pose any relevant constraint.

The BP-GED is calculated thus providing us with the node correspondences between the pair of graphs. For activity prediction, the above process is performed between the test activity and all the reference activities. The test activity takes the label of the reference activity that gives rise to the smallest BP-GED. A detailed explanation of this graph-based action/activity prediction algorithm can be found in [18].

4 Experimental Results

4.1 Datasets

The assessment of the considered methods is performed on benchmark datasets for action/activity prediction. These datasets contain actions and activities in trimmed and untrimmed sequences performed by different subjects manipulating a variety of objects. Each action is represented by the poses of human/object and the class of the manipulated objects using various features, as described in the following.

CAD-120 Dataset [13]**:** The CAD-120 dataset contains long and complex activities of human-object interactions. The activities are performed by male and female subjects and filmed from varying viewpoints. Moreover, the same actions are performed with different objects in order to induce variability in the executions. These activities can be trimmed in actions based on the provided ground truth data. The dataset provides annotations regarding the activity and action labels, object labels, affordance labels and temporal segmentation of activities. The activities are: arranging objects, cleaning objects, having meal, making cereal, microwaving food, picking objects, stacking objects, taking food, taking medicine and un-stacking objects. The action labels are: reach, move, pour, eat, drink, open, place, close, clean. The comparative evaluation is done based on the experimental split adopted in [18,30].

In each video, we consider the upper body joints of the acting person as well as the manipulated objects as in [18,30]. Specifically, the 3D locations of the 8 upper body joints are employed, as well as the distance moved by each joint and their displacement. As object features we employ their 3D centroid location, the distance between the object centroid and each of the 8 upper body human joints. Also, the distance moved by the object (i.e., the displacement of the object's centroid).

MSR Daily Activity 3D Dataset [29]**:** The MSR Daily Activity 3D Dataset contains actions performed by different subjects in an indoor environment. A small part of these trimmed actions do not contain human-object interactions but the majority does. The actions are performed twice by all subjects, the first time by standing up while the second by sitting on a sofa. The actions contained in the dataset are: cheering up, sitting still, playing a game, walking, lie down on the sofa, playing the guitar, reading a book, standing up, drinking, sitting down, eating, tossing paper, speaking on cellphone, writing on paper, using a laptop and using a vacuum cleaner. Again, we follow the evaluation split used in [17, 18, 24].

In the MSR Daily Activity dataset the estimation of the lower body positions are very noisy, so following the related works [16, 18] we take into consideration only the upper body. The representation consists of a 45-dimensional feature vector containing the 3D joint angles and the 3D skeletal joint position relative to the body center. The dataset does not provide object classes and object positions, so we acquired them using YoloV4 [4].

MHAD101-s/-v Datasets [21]**:** The MHAD101-s/-v Datasets are constructed based on actions in the MHAD dataset. The MHAD dataset contains 11 trimmed human actions which mainly do not contain human-object interactions with the exception of one action. The actions are performed with different execution styles and speeds. The actions are: waving two hands, clapping, throwing a ball, sit down and stand up, sit down, stand up, jumping in place, jumping jacks, bending, punching, waving one hand. These actions are used in the MHAD101-s/-v to form longer sequences of actions starting from sequences containing from 3 to 7 actions in a row. The concatenated actions exclude the action sit down/stand up as the combination of these two actions results in ambiguities and confusion. From the 101 pairs of action sequences contained in the dataset we use only the first 50 paired sequences where each sequence consists of 3 concatenated action clips (triplets) and the paired sequences have exactly 1 in common. We used only the first 50 pairs of action sequences. By splitting these 50 pairs, we obtained 100 action sequences where each of them contains 3 concatenated actions. An important aspect is that the style and duration variability are enforced by using different subjects in forming different triplets.

The MHAD101-s is constructed using skeletal data and features. We employ the same human body representation as in [15, 17] based on the 3D skeletal data of the 30 human joints provided from a motion capture system. The employed features are body-centered and camera-centered providing a 60-dimensional vector plus four angles representing the angles encoding the fore- and the back- arms and upper- and lower legs. The MHAD101-v dataset contains the RGB videos of the same triplets as in MHAD101-s. We then extract features from the VGG-16 network as in [3]. We took into account all the available frames without down-sampling to 30fps.

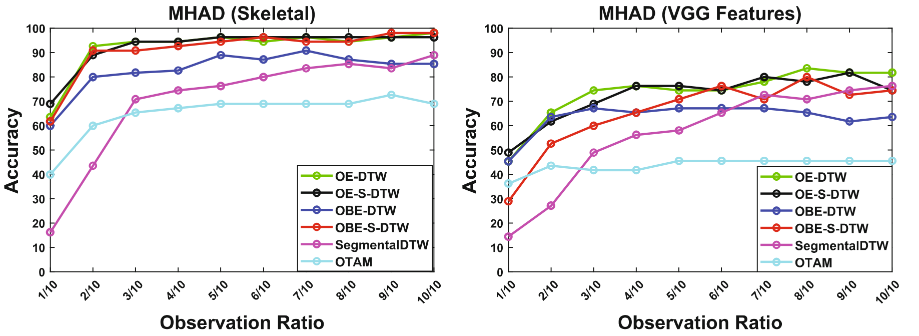

Fig. 4. Action prediction accuracy in trimmed videos as a function of observation ratio involving skeletal (left) and VGG (right) features on the MHAD dataset.

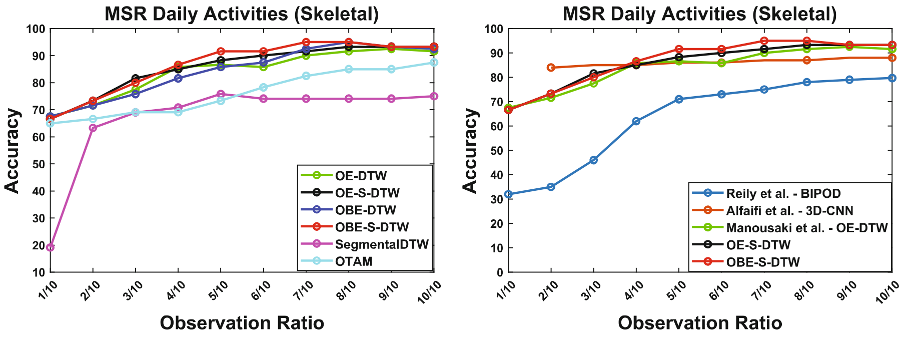

Fig. 5. Action prediction accuracy in trimmed videos as a function of the observation ratio involving skeletal features in the MSR Daily Activities dataset. (Left) a comparison of the OBE-S-DTW and OE-S-DTW with other alignment algorithms, (right) a comparative evaluation of the proposed algorithms with the state-of-art on action prediction on the MSR dataset.

4.2 Alignment-Based Prediction in Segmented Sequences

Action Prediction: The OE-S-DTW and OBE-S-DTW algorithms are employed for solving the problem of action and activity prediction in trimmed sequences. For this reason we evaluate them on the trimmed action executions of the MHAD (skeletal & VGG features), MSR (skeletal features) and CAD120 (skeletal features) datasets and on the activities of the CAD120 (skeletal features) dataset. These algorithms are evaluated by employing the framework of Manousaki et al. [16] as described in Sect. 3.3 which aligns test and reference action sequences and classifies them based on the closest match.

Performance Metrics: To assess our method we use the standard observation split of a certain test action/activity in parts of 10% in the range $[0, 100\%]$. At every successive such split, we match the (partially) observed test sequence to the reference ones and we calculate the agreement between the predicted activity/action label and the ground truth label. An observation ratio of 100% means that the whole test sequence

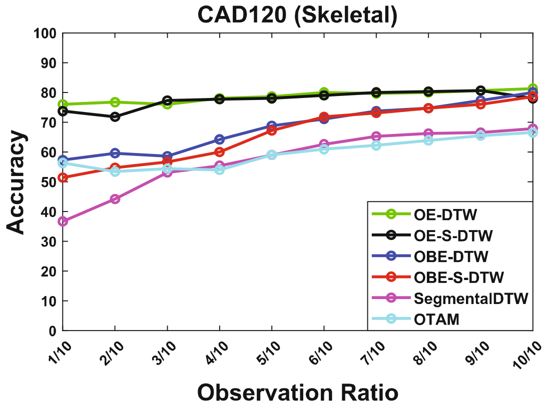

Fig. 6. Action prediction results on the CAD120 dataset.

has been observed, therefore this is equivalent to solving the problem of action/activity classification.

Action Prediction Results: Figure 4 shows the results of action prediction on the MHAD dataset using skeletal and VGG features. For this segmented input it is expected that the Open-End DTW and S-DTW variants will have the best results. Also, these algorithms perform better than the SOTA algorithms. Figure 5 shows the action prediction results on the MSR dataset. The left plot illustrates a comparison with existing alignment algorithms. We observe that the OE-S-DTW and OBE-S-DTW algorithms have the best results across all observation ratios. The right plot of the same figure shows a comparison of OE-S-DTW and OBE-S-DTW with the state-of-art work on action prediction on the MSR dataset of Alfaifi et al. [2], Manousaki et al. [17] and Reily et al. [24]. For an observation ratio greater than 40% the OBE-S-DTW outperforms all other algorithms. Finally, Fig. 6 shows a comparison of the OBE-S-DTW and OE-S-DTW algorithms with the OE-DTW [28], OBE-DTW [28], Segmental-DTW [19] and OTAM [5] algorithms on the actions of the CAD120 dataset.

Activity Prediction Results: On top of the results presented in [16], we also evaluate the framework of Manousaki et al. [16] on the activities of the CAD120 dataset. To do so, we use compare the performance of the OE-S-DTW and the OBE-S-DTW methods to several competitive methods [2,2,5,19,24] and the works of Wu et al. [30] and Manousaki et al. [18] that hold the state-of-art results. As it can be observed in Fig. 7 the OE-S-DTW alignment algorithm performs generally better than the OBE-S-DTW. This happens due to the fact that most of the activities start from the same pose so the OE-S-DTW that has the starting point constraint aligns the sequences more accurately. Generally, the alignment of activities using the framework of [16] has low performance compared to the state-of-art. This happens because the activities are more complex compared to actions. Additionally this method has the drawback of taking into account only one object while in these activities several objects are used.

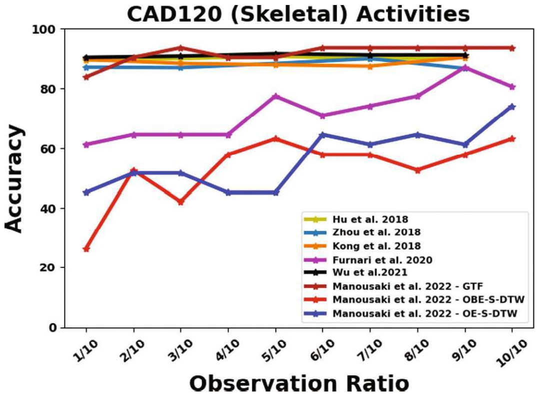

Fig. 7. Activity prediction results on the CAD120 dataset.

4.3 Alignment-Based Action Prediction in Unsegmented Sequences

Action Prediction: The OBE-S-DTW algorithm is capable of finding an action anywhere inside a long sequence of actions as shown in [16] for the MHAD101-s/-v datasets. This was achieved by checking whether the OBE-S-DTW can recognize an unsegmented action that appears between some other prefix and suffix actions. We complement those experiments here by observing action triplets not only from start to finish as in [16] but also backwards (from the suffix to the prefix). It is noted that in each observed triplet, the prefix/suffix actions are excluded from the set of the prototype actions.

Performance Metrics: The prefix and suffix actions are progressively observed in thirds while the middle action is observed in tenths. This protocol is used to acquire finer performance measures for the action of interest. Accuracy is used as the metric for action prediction of the middle action. F1-score, precision, recall and Intersection-over-Union (IoU) are calculated for all observation ratios.

Action Prediction Results: Figure 8 shows the F1-score, recall, precision and IoU scores for the OBE-DTW and OBE-S-DTW algorithms which are the better algorithms overall for aligning unsegmented sequences. From this plot we can observe that the OBE-S-DTW provides better alignments.

A comparison of the OBE-S-DTW and OE-S-DTW with the OE-DTW [28], OBE-DTW [28], SegmentalDTW [23] and OTAM [5] is provided in Fig. 9 (left) for the MHAD101-s and in Fig. 9 (right) for the MHAD101-v dataset. As it can be observed, the OBE-S-DTW has the best performance overall.

As mentioned earlier, on top of the results presented in Manousaki et al. [16], we evaluate the performance of the OE-DTW and OBE-S-DTW algorithms on reversed action triplets. In Fig. 10 (left) we observe how the algorithms performs for the MHAD101-s dataset while observing the triplet from start to end (deep red and deep blue lines) and how they perform while observing the triplet from the end to the start (light red and light blue lines). In Fig. 10, the two vertical black lines denote the

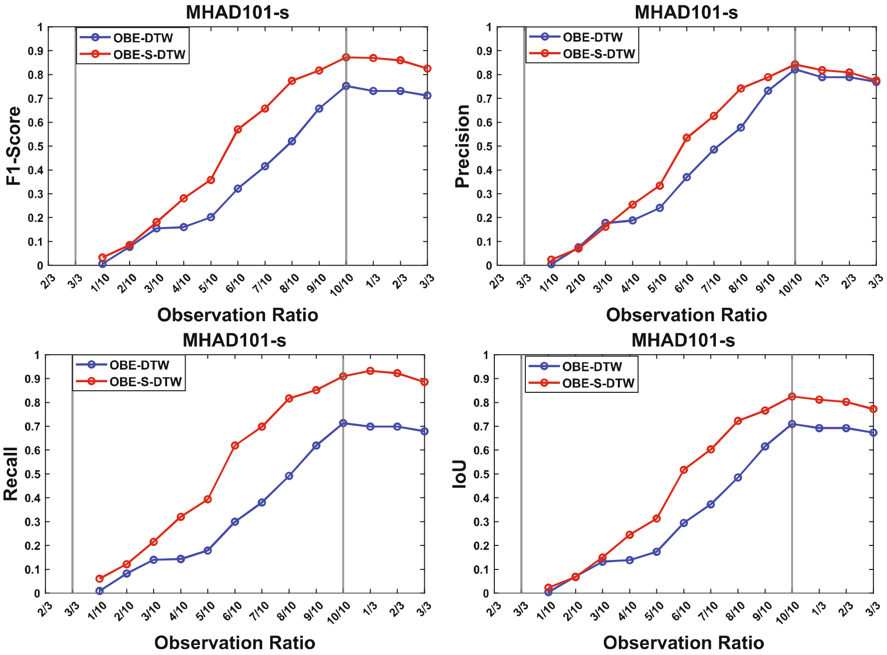

Fig. 8. Performance metrics for OBE-DTW and OBE-S-DTW on the MHAD101-s dataset.

ground-truth start and end of the middle action. High accuracy during the prefix denotes the ability of the algorithm to recognize that the algorithm correctly identifies that the sought action has not yet started. High accuracy during the suffix denotes the successful recognition of the middle action inside the triplet. We can observe a symmetrical effect in the results which means that observing the triplet from start to end or vice versa does not have a significant impact on the algorithms. Also, the OBE-S-DTW algorithm consistently outperforms the OBE-DTW algorithm. The same holds for the MHAD101-v dataset as it can be observed in Fig. 10 (right).

4.4 Graph-Based Activity Prediction

We evaluate the performance of OE-S-DTW and OBE-S-DTW on the problem of activity prediction when employed in the graph-based framework of Manousaki et al. [18]. As it can be observed in Fig. 11 the OBE-S-DTW algorithm helps the GTF method achieve much better results compared to the use of the OE-S-DTW algorithm. This method considers multiple objects for the task of activity prediction. The motion weights with the use of OE-S-DTW and OBE-S-DTW are calculated for each pair of nodes in the action graphs and the bipartite graph. The boundary constraint of OE-S-DTW is not an advantage in this setting where objects can be used in different points in time and in mixed order.

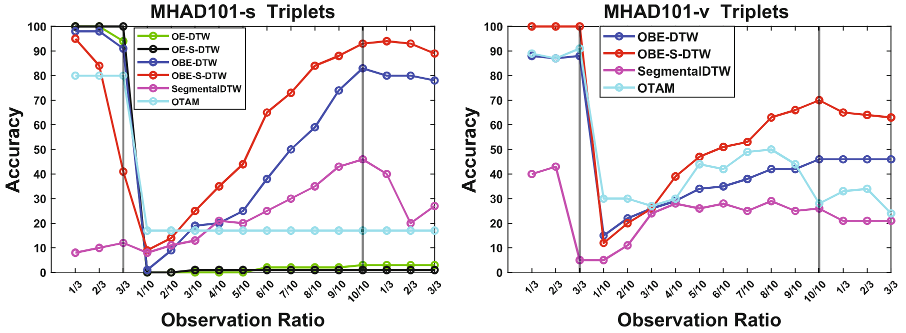

Fig. 9. Aligning unsegmented action sequences on the MHAD101-s and MHAD101-v datasets by comparing the OBE-S-DTW and OBE-DTW algorithms to other pre-existing alignment algorithms while observing the triplets from the prefix to the suffix. Vertical lines depict the limits of the middle action.

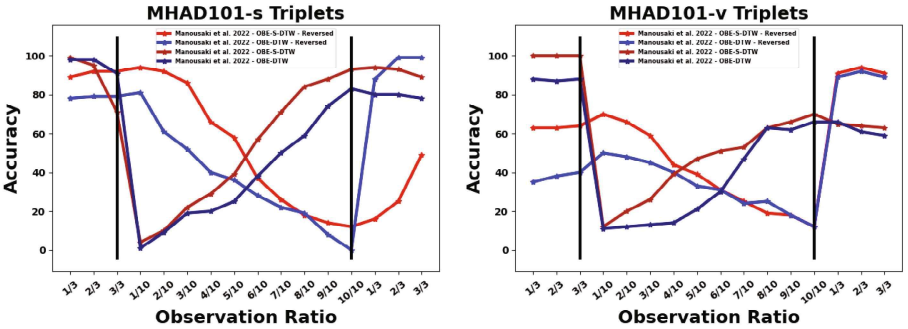

Fig. 10. Aligning unsegmented action sequences on the MHAD101-s/-v datasets using the OBE-S-DTW and OBE-DTW algorithms. Dark red and blue lines depict the accuracy of the alignment algorithms while observing the triplets from the prefix to the suffix. The light (red and blue) lines depict the observation of the triplet from the suffix to the prefix. (Color figure online)

4.5 Duration Prognosis

Knowing the label of an ongoing action before its completion is a very useful capability. In certain situations, it is equally important to be able to predict the time at which the currently observed action/activity will end. As proposed in [16], action duration prognosis is defined as the prediction of the time remaining until the completion of the currently observed action. Currently in the framework of [16] the duration prognosis has been evaluated only on the MHAD101-s dataset (see also Fig. 12). We extend this evaluation by performing duration prognosis of actions and activities on the MSR and CAD120 datasets, respectively.

Performance Metrics: For a given observation ratio, we report the end-frame prediction error which is defined as the discrepancy of the estimated end of a certain action/activity from its ground truth end, as a percentage of the test action length. When

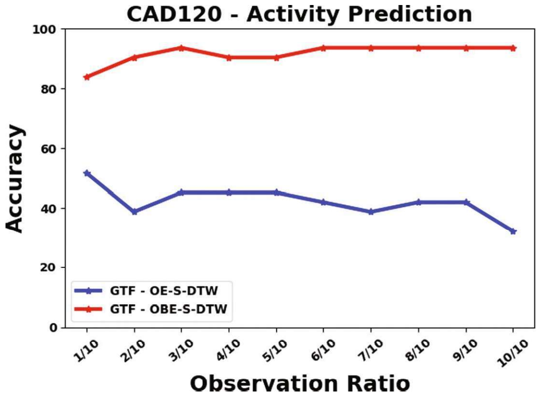

Fig. 11. Activity prediction results on the CAD120 dataset using the graph based method presented in [18]. The OBE-S-DTW and OE-S-DTW algorithms were used to quantify the motion dissimilarity of the entities involved in the considered activities.

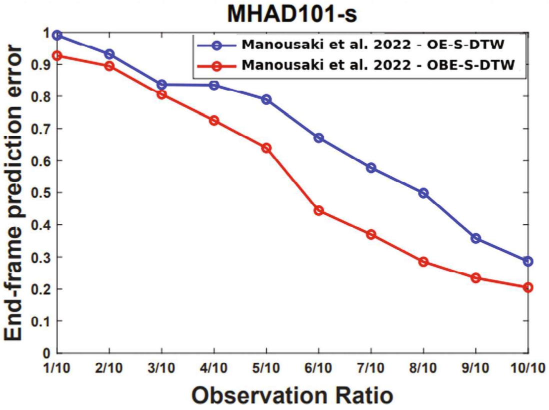

Fig. 12. End-frame prediction error calculated for all observation ratios of the middle actions of the triplets of MHAD101-s.

an action/activity is wrongly classified by the algorithm, then a prediction error of 1.0 (100%) is added.

Action Duration Prognosis Results: Figure 12 shows the results of duration prognosis on the triplets of the MHAD101-s dataset, as presented in [16]. We can observe that the OBE-S-DTW has smaller error rates across all observation ratios compared to OBE-DTW.

We also report the evaluation of duration prognosis on the actions of the MSR dataset. Figure 13 (left) show the relevant action prediction results while in Fig. 13 (right) the results of the duration prognosis are presented. The OBE-S-DTW is the best choice for this dataset across the different frameworks. The higher the action prediction accuracy, the lower the duration prognosis error. As the observation ratios increase

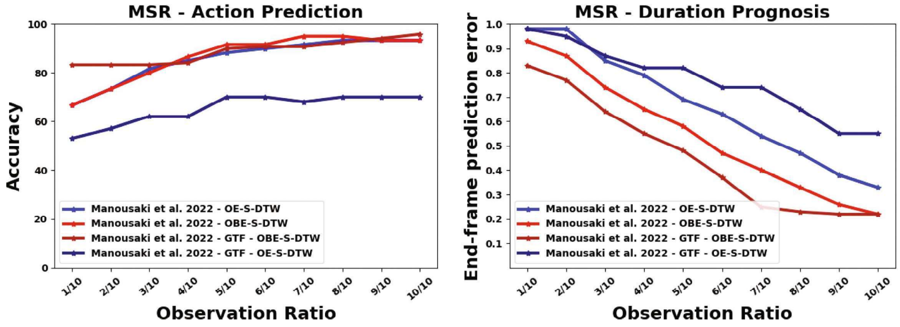

Fig. 13. (Left) Action prediction results for the MSR Daily Activities dataset. (Right) Prognosis of the duration of the partially observed actions for the MSR Daily Activities dataset.

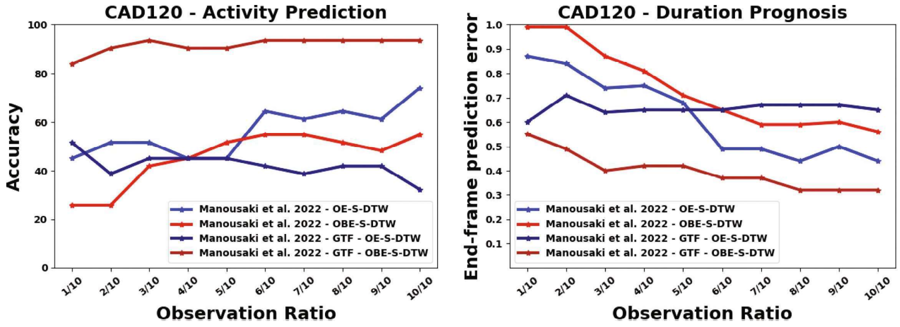

Fig. 14. (Left) Activity prediction results for the CAD120 dataset. (Right) Prognosis of the duration of the partially observed activities of the CAD120 dataset.

more meaningful alignments are established thus matching with actions of similar temporal duration. While the prediction accuracy of the different frameworks is similar, we observe differences in the end-frame prediction error. This happens because the test actions get classified to different reference actions thus having variability in their predicted temporal duration.

Activity Duration Prognosis Results: Complementary to the duration prognosis for actions, we extend the experimental evaluation of duration prognosis on the activities of the CAD120 dataset. In Fig. 14 we present the results of activity prediction and duration prognosis of the framework of [16] and the GTF framework [18] on the CAD120 dataset. As it can be observed, the OE-S-DTW performs better than the OBE-S-DTW in the framework of [16] while the opposite holds true for the GTF framework. As explained earlier, these frameworks handle different numbers of objects. Thus, depending on the framework, the dataset and its characteristics, different alignment algorithms should be employed. Moreover, moving forward onto the timeline we can see that the end-frame prediction error is decreasing for both algorithms, as they observe a larger portion of the test activity.

5 Conclusions

In this paper we presented an extensive evaluation of the OBE-S-DTW and OE-S-DTW alignment algorithms on the task of human action prediction and duration prediction on the MHAD, MSR Daily Activities, CAD-120 and MHAD101-s/-v datasets. These algorithms were deployed for the alignment and matching of segmented and unsegmented action executions. We opted to extend the use of these algorithms for the task of activity prediction by incorporating them in different frameworks and testing them on the activities of the CAD120 dataset. Our experimental evaluation showed that the OE-S-DTW algorithm is better suited for the alignment of segmented sequences while the OBE-S-DTW for the alignment of unsegmented sequences. The OBE-S-DTW is a powerful algorithm that can identify actions as parts of unsegmented sequences. We showed that the OBE-S-DTW algorithm can align/match sequences with similar duration thus having low end-frame prediction error compared to the OBE-DTW algorithm. Additionally, the OE-S-DTW and OBE-S-DTW algorithms are proven to be better than several competitive algorithms. Moving forward, we will exploit the properties of these alignment algorithms in a deep neural network to assess their effectiveness and performance in a learning-based framework for action and activity prediction.

Acknowledgements. This research was co-financed by Greece and the European Union (European Social Fund-ESF) through the Operational Programme "Human Resources Development, Education and Lifelong Learning" in the context of the Act "Enhancing Human Resources Research Potential by undertaking a Doctoral Research" Sub-action 2: IKY Scholarship Programme for PhD candidates in the Greek Universities. The research work was supported by the Hellenic Foundation for Research and Innovation (HFRI) under the HFRI PhD Fellowship grant (Fellowship Number: 1592) and by HFRI under the "1st Call for H.F.R.I Research Projects to support Faculty members and Researchers and the procurement of high-cost research equipment", project I.C.Humans, number 91.

References

1. Abu-Aisheh, Z., Raveaux, R., Ramel, J.Y., Martineau, P.: An exact graph edit distance algorithm for solving pattern recognition problems. In: ICPRAM (2015)
2. Alfaifi, R., Artoli, A.: Human action prediction with 3D-CNN. SN Comput. Sci. **1**, 1–15 (2020)
3. Bacharidis, K., Argyros, A.: Improving deep learning approaches for human activity recognition based on natural language processing of action labels. In: IJCNN. IEEE (2020)
4. Bochkovskiy, A., Wang, C., Liao, H.: Yolov4: optimal speed and accuracy of object detection. arXiv:2004.10934 (2020)
5. Cao, K., Ji, J., Cao, Z., Chang, C.Y., Niebles, J.C.: Few-shot video classification via temporal alignment. In: CVPR (2020)
6. Chang, C.Y., Huang, D.A., Sui, Y., Fei-Fei, L., Niebles, J.C.: D3TW: discriminative differentiable dynamic time warping for weakly supervised action alignment and segmentation. In: CVPR (2019)
7. Cuturi, M., Blondel, M.: Soft-DTW: a differentiable loss function for time-series. arXiv:1703.01541 (2017)
8. Dvornik, N., Hadji, I., Derpanis, K.G., Garg, A., Jepson, A.D.: Drop-DTW: aligning common signal between sequences while dropping outliers. arXiv preprint arXiv:2108.11996 (2021)

9. Fellbaum, C.: Wordnet and wordnets (2005)
10. Hadji, I., Derpanis, K.G., Jepson, A.D.: Representation learning via global temporal alignment and cycle-consistency. arXiv preprint arXiv:2105.05217 (2021)
11. Haresh, S., et al.: Learning by aligning videos in time. arXiv preprint arXiv:2103.17260 (2021)
12. Kim, D., Jang, M., Yoon, Y., Kim, J.: Classification of dance motions with depth cameras using subsequence dynamic time warping. In: SPPR. IEEE (2015)
13. Koppula, H., Gupta, R., Saxena, A.: Learning human activities and object affordances from RGB-D videos. Int. J. Robot. Res. **32**(8), 951–970 (2013)
14. Loper, E., Bird, S.: NLTK: the natural language toolkit. arXiv preprint CS/0205028 (2002)
15. Manousaki, V., Papoutsakis, K., Argyros, A.: Evaluating method design options for action classification based on bags of visual words. In: VISAPP (2018)
16. Manousaki, V., Argyros, A.A.: Segregational soft dynamic time warping and its application to action prediction. In: VISIGRAPP (5: VISAPP), pp. 226–235 (2022)
17. Manousaki, V., Papoutsakis, K., Argyros, A.: Action prediction during human-object interaction based on DTW and early fusion of human and object representations. In: Vincze, M., Patten, T., Christensen, H.I., Nalpantidis, L., Liu, M. (eds.) ICVS 2021. LNCS, vol. 12899, pp. 169–179. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-87156-7_14
18. Manousaki, V., Papoutsakis, K., Argyros, A.: Graphing the future: activity and next active object prediction using graph-based activity representations. In: 17th International Symposium on Visual Computing (2022)
19. Panagiotakis, C., Papoutsakis, K., Argyros, A.: A graph-based approach for detecting common actions in motion capture data and videos. Pattern Recognit. **79**, 1–11 (2018)
20. Papoutsakis, K., Panagiotakis, C., Argyros, A.: Temporal action co-segmentation in 3D motion capture data and videos (2017)
21. Papoutsakis, K., Panagiotakis, C., Argyros, A.A.: Temporal action co-segmentation in 3D motion capture data and videos. In: CVPR 2017. IEEE (2017)
22. Papoutsakis, K., Panagiotakis, C., Argyros, A.A.: Temporal action co-segmentation in 3D motion capture data and videos. In: CVPR (2017)
23. Park, A.S., Glass, J.R.: Unsupervised pattern discovery in speech. IEEE Trans. Audio Speech Lang. Process. **16**(1), 186–197 (2007)
24. Reily, B., Han, F., Parker, L., Zhang, H.: Skeleton-based bio-inspired human activity prediction for real-time human-robot interaction. Auton. Robots **42**, 1281–1298 (2018)
25. Roditakis, K., Makris, A., Argyros, A.: Towards improved and interpretable action quality assessment with self-supervised alignment (2021)
26. Sakoe, H., Chiba, S.: Dynamic programming algorithm optimization for spoken word recognition. IEEE Trans. Acoust. Speech Signal Process. **26**(1), 43–49 (1978)
27. Schez-Sobrino, S., Monekosso, D.N., Remagnino, P., Vallejo, D., Glez-Morcillo, C.: Automatic recognition of physical exercises performed by stroke survivors to improve remote rehabilitation. In: MAPR (2019)
28. Tormene, P., Giorgino, T., Quaglini, S., Stefanelli, M.: Matching incomplete time series with dynamic time warping: an algorithm and an application to post-stroke rehabilitation. Artif. Intell. Med. **45**(1), 11–34 (2009)
29. Wang, J., Liu, Z., Wu, Y., Yuan, J.: Mining actionlet ensemble for action recognition with depth cameras. In: IEEE CVPR (2012)
30. Wu, X., Wang, R., Hou, J., Lin, H., Luo, J.: Spatial-temporal relation reasoning for action prediction in videos. Int. J. Comput. Vision **129**(5), 1484–1505 (2021)
31. Wu, Z., Palmer, M.: Verb semantics and lexical selection. arXiv preprint CMP-LG/9406033 (1994)
32. Yang, C.K., Tondowidjojo, R.: Kinect V2 based real-time motion comparison with retargeting and color code feedback. In: IEEE GCCE (2019)