Unsupervised and Explainable Assessment of Video Similarity

Konstantinos Papoutsakis
papoutsa@ics.forth.gr
Antonis Argyros
argyros@ics.forth.gr

Computer Science Department,
University of Crete, Greece
Computational Vision & Robotics Lab,
Institute of Computer Science,
FORTH, Greece

Abstract

We propose a novel unsupervised method that assesses the similarity of two videos on the basis of the estimated relatedness of the objects and their behavior, and provides arguments supporting this assessment. A video is represented as a complete undirected action graph that encapsulates information on the types of objects and the way they (inter)act. The similarity of a pair of videos is estimated based on the bipartite Graph Edit Distance (GED) of the corresponding action graphs. As a consequence, on-top of estimating a quantitative measure of video similarity, our method establishes spatiotemporal correspondences between objects across videos if these objects are semantically related, if/when they interact similarly, or both. We consider this an important step towards explainable assessment of video and action similarity. The proposed method is evaluated on a publicly available dataset on the tasks of activity classification and ranking and is shown to compare favorably to state of the art supervised learning methods.

1 Introduction

Video understanding and human action analysis have been gaining increasing interest in computer vision in both academia and industry. Considerable milestones have been achieved on challenging recognition problems [21, 61] ranging from coarse action classification, segmentation and prediction in short clips, to fine-grained understanding and reasoning on human-object interactions in long videos.

Humans have a remarkable ability to analyze and reason about the spatio-temporal relationships of actors and objects in images and videos. Consider, for example, the three snapshots from videos in Figure 1. In video A a person stirs a cup of coffee using a spoon; in B a person stirs a bowl with a spatula while pouring liquid from a jug and in C a person pours cereals into a bowl. Humans can analyze effortlessly such videos and assess the semantic similarity between videos A and B relative to A and C or B and C based on the affinity of the motion patterns of stirring that occurs in both activities in A and B and the meaningful associations between the used tools (spoon, spatula, used for stirring) and the objects (cup, bowl are containers). Similar analysis applies for the pouring action in videos B and C.

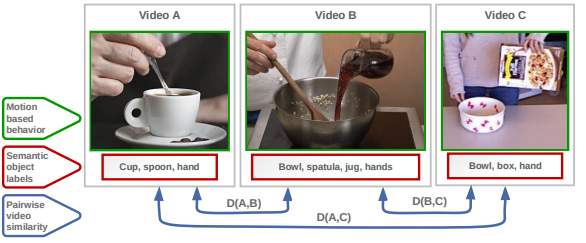

Figure 1: Video A: stirring a cup of coffee using a spoon, B: stirring a bowl using a spatula while pouring liquid using a jug, C: pouring cereals into a bowl. Humans can reason about these actions and assess the (dis)similarity $D(X,Y)$ of videos X,Y and provide argumentation on this by establishing spatiotemporal correspondences between the actors, the manipulated objects and their behavior. We propose an unsupervised method that aims at these goals.

Motivated by the above observations, we present an unsupervised method that achieves an *explainable* assessment of the similarity[1] of two videos. Our method represents each video as an action graph, i.e. a complete, undirected, labeled graph whose nodes correspond to tracked objects. The edges of the graph represent object relations and interactions. The weight of an edge aggregates the dissimilarity of the objects it connects with respect to (i) their semantic similarity (estimated based on the semantic affinity of their labels) and (ii) the dissimilarity of their behavior in time (i.e. estimated based on temporal co-segmentation of their trajectories). Then, the distance of a pair of videos is estimated based on an approximation of the Graph Edit Distance (GED) between the corresponding action graphs. GED establishes meaningful correspondences between the two compared videos, i.e., identifies object pairs that are semantically related, exhibit similar interactions with other objects, or both. Similar actions/interactions are also localized in time. Thus, on-top of estimating a quantitative measure of video similarity, the proposed method provides explanations of why and when two videos are similar.

Our methodology is evaluated using the CAD-120 dataset [20] on the tasks of nearest-neighbor action classification and pairwise action matching and ranking, and is shown to compare favorably to state-of-art supervised and unsupervised learning methods.

2 Related Work

Action similarity: Action similarity or action matching in images or videos comprises a set of sub-problems in action recognition, where a test action should be matched with the k most similar actions (possibly of the same category) from a given dataset of actions. Kliper-Gross *et al.* [25, 26] defined action pair-matching, as the task of determining whether actors in two given videos are performing the sane same activity or not. A series of methods have been proposed for distance metric learning towards (a) one-shot recognition [25, 58, 60, 63, 67, 68, 69], where only a single example of each possible class is available and (b) supervised learning based on SVM-based multi-class or ranking models [9, 30, 43], ensembles of non-linear weak classifiers [46], or Dynamic Time Warping [31, 32]. Moreover, Celiktutan *et al.* [10] presented an action recognition method that matches hyper-graphs generated based on interest points in videos. Contemporary methods

[1] Our method computes a distance of two videos. Given that a distance measure can be turned into a similarity measure, we use these two terms interchangeably.

also assess action similarity in the context of automatic skill determination [12, 13, 33] or action quality assessment [9, 44], where the goal is to evaluate how well an actor performs a given task compared to an instructional video as well as to provide interpretable feedback on how performers can improve their actions. Finally, in the context of video/action alignment, Baraldi [3] recently proposed a learnable approach for video comparing and aligning, integrating temporal match kernels within neural networks. Evaluation on the tasks of video alignment, copy detection and event retrieval was carried out.

Unsupervised human-object interaction understanding: Recently, several unsupervised methods have been proposed to learn temporal structure and human-object interactions in videos. Girdhar *et al*. [18] presented a novel deep model to learn spatio-temporal context of human actions based on tracked humans and objects in a video clip using self-attention towards action localization and classification. Other non-parametric probabilistic methods proposed models based on topic models [64] or the hierarchical Dirichlet process prior [41] to discover representative co-occurrences and temporal relations between atomic actions to recognize human-object interactions. In a similar spirit, the work presented by Duckworth *et al*. [14] aims to understand human activities based on a generative probabilistic method, while Mici *et al*. [36] presents an unsupervised, hierarchical framework of self-organizing neural networks for human-object interaction recognition from RGB-D videos. Finally, a recently proposed method by Chang *et al*. [11] deals with the problem of weakly-supervised action alignment and segmentation. A novel approach of Dynamic Time Warping is introduced based on a differentiable loss function and a deep network based architecture. The method uses an ordered list of actions labels occurring in a training video as supervision.

Graph-based action recognition/visual reasoning: A series of graph-based neural networks approaches have recently been proposed for action recognition and object-level visual reasoning in videos using as input the detected human and object masks across time. Jain *et al*. [23] combines spatio-temporal graphs and sequence learning based on RNNs. Baradel *et al*. [2] introduced the Object Relation Network, Wang *et al*. [62] treats videos as space-time region graphs and Qi *et al*. [45] introduced the Graph Parsing Neural Network (GPNN) based on an actor-object graph structure. Guo *et al*. [19] introduced the Neural Graph Matching Networks (NGM) to recognize previous unseen 3D action classes by learning based on only a few examples. The method consists of two stage, namely (i) the interaction graph generation, where graph nodes represent physical entities in a 3D scene (e.g. body parts, objects) and edges to represent the interactions between the entities for each action, and (ii) the graph-based matching metric learning, in order to enable few-shot training on the generated interaction graph. It also uses the CAD-120 dataset, as our proposed work, but focus on learning the sub-activity labels and their combinations with the interacting objects. Overall, the reported graph-based methods achieve high-level comprehension of human-object interactions by learning intuitive and interpretable patterns in terms of pairwise spatio-temporal co-occurrences and transformations of objects that characterize different activities. Finally, we refer to deep graph similarity learning method by Li [66] that proposes an extension of Graph Neural Networks [24], noted as the Graph Matching Networks (GMNs). The GMNs compute a similarity score through a cross-graph attention mechanism to associate nodes across graphs and identify differences. Making the graph representation computation dependent on the pair of graphs makes the matching model more powerful than an embedding model. The method has also been applied on a synthetic graph edit-distance learning task.

Graph Edit Distance (GED): GED is a widely used method for error-tolerant graph matching [7, 54] being under active exploration in pattern recognition [1, 17, 51, 55], but to a much

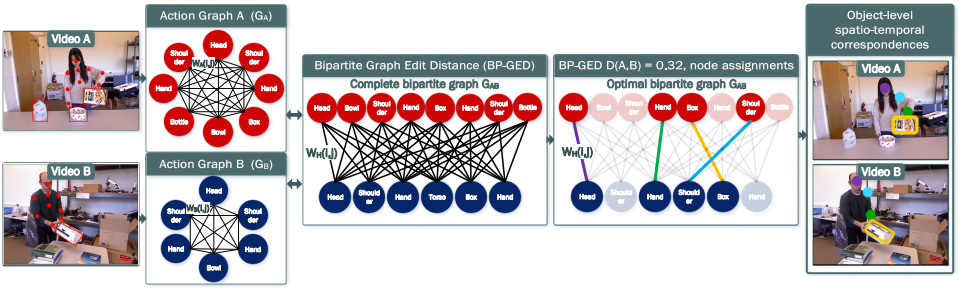

Figure 2: An overview of the proposed method for explainable action similarity assessment. The two compared videos are represented as fully connected action graphs from which a complete bipartite graph is constructed. The Bipartite Graph Edit Distance [50] on this graph quantifies the dissimilarity of the corresponding videos and provides meaningful spatiotemporal correspondences between objects that are semantically similar, behave similarly, or both. Matched objects appear in the same color in the last column.

lesser extend in computer vision [5, 6]. The main idea of the graph edit distance is that of finding the dissimilarity of two graphs by a set of operations (substitution, deletion and insertion of nodes and edges) of minimum total cost that are required to transform one graph to the other. The cost functions and parameters of GED and can be user-defined or learned from a set of labeled graphs [38, 40]. Neuhaus and Bunke [39] provided evidence that the distance quality of approximated GED is sufficiently accurate for nearest-neighbor-based classifiers or edit distance based kernel machines.

Our contributions: Two are the most significant contributions of this work: (i) it proposes a novel *unsupervised* method for the quantitative assessment of the similarity of two videos that performs comparably to contemporary state of the art *supervised* methods, (ii) based on the bipartite Graph Edit Distance, the proposed method establishes semantically meaningful spatio-temporal correspondences between objects of the compared videos contributing to the explainability of the suggested quantitative video similarity assessment. The code and dataset used in this work will become publicly available.

3 Explainable Assessment of the Similarity of Activities

Our approach consists of (a) an object-level graph-based representation of a video as an action graph (Section 3.1), and (b) a method for assessing the similarity of two videos based on the bipartite Graph Edit Distance of the corresponding action graphs (Section 3.2).

3.1 Graph-based Video Representation

We represent a video V of T frames with an undirected, complete graph, noted as $G = (V, E)$. We assume that throughout the video, a total of N entities can be detected, localized and tracked [43] at any time point and for various time intervals. Such entities include objects, but also localized human body joints. We associate a graph node $v \in V$ with each such entity, therefore $|V| = N$. Graph edges $e_{ij} = (v_i, v_j) \in E$, represent pairwise relations of the connected vertices $v_i, v_j \in V$. Each edge e_{ij} is associated with a weight w_{ij} defined as

$$w_{ij} = (1 - \lambda) \cdot m_{ij} + \lambda \cdot s_{ij}. \quad (1)$$

In Eq.(1), w_{ij} aggregates the dissimilarity of the objects it connects, i.e., their semantic dissimilarity s_{ij} and the dissimilarity m_{ij} of their motion/behavior in time. $\lambda \in [0,1]$ is a user-defined parameter that tunes the relative contribution of the motion-based and semantic information. In our experiments, we test $\lambda = 0.5$ to allow for equal contribution of both types of information and $\lambda = 0$ to test the role of the motion information, alone. The $|N| \times |N|$ cost matrix $W = \{w_{ij}\}$ denotes all the pairwise dissimilarities of the nodes of a graph.

Computation of s_{ij} based on semantic features: Given two graph nodes v_i, v_j, their semantic dissimilarity s_{ij} is computed based on the semantic similarity of their recognized labels, l_i and l_j. The computation of these labels is performed based on object recognition methods [47, 48] or provided as ground truth annotations of the video. The semantic similarity $L(l_i, l_j)$ of these labels is estimated using the WordNet [15] lexical database and the Natural Language Toolkit [35] to compute the path-based Wu-Palmer scaled metric [65]. This depends on the depth of l_i and l_j and that of their Least Common Subsumer (most specific ancestor node) in the hierarchical tree-based taxonomy of WordNet. $L(l_i, l_j)$ is in the range (0, 1] (1 for identical concepts). Thus,

$$s_{ij} = 1 - L(l_i, l_j). \tag{2}$$

Computation of m_{ij} based on motion features: Each entity v_i in a frame of a video is represented as a d-D feature vector. Such a vector can be as simple as the entity's $2D$ image position, or a higher dimensional vector encoding position, shape, appearance, e.t.c. Assuming that v_i appears in the video for T_i consecutive frames, a time series F_i of length t_i, $0 \leq t_i \leq T$, of d-D vectors represents the evolution/behavior of v_i in time. To compute the dissimilarity m_{ij} of the behavior of v_i and v_j in time, we need to establish a comparison between F_i and F_j. To this end, we opt for an unsupervised temporal co-segmentation method [42] which can discover multiple commonalities in times series of multidimensional data. In particular, given two time series F_i and F_j, the EVACO co-segmentation method [42] performs a stochastic optimization to compute an automatically identified number n_c of commonalities $J_k(i,j)$, $0 \leq k \leq n_c$. For each of them, EVACO computes a discrepancy $\Delta(J_k(i,j))$ representing the normalized DTW-based alignment cost of the co-segmented sub-sequences. Moreover, the temporal coverage $C(i,j) \in (0,1]$ of the discovered commonalities is defined as the ratio of the number of frames of F_i and F_j that are part of any of the discovered commonalities. Given the above, the motion-based dissimilarity of v_i and v_j is defined as:

$$m_{ij} = \frac{1}{C(i,j)} \cdot \frac{1}{n_c} \sum_{k=1}^{n_c} \Delta(J_k(i,j)). \tag{3}$$

Intuitively, the dissimilarity score m_{ij} for a pair of objects v_i and v_j is inversely proportional to the length of the discovered commonalities and proportional to their average discrepancy. Our approach is able to discover pairwise, common motion-based patterns between objects detected at varying different moments and tracked for largely variable duration in time.

3.2 Graph Edit Distance-based Video Comparison

Having represented two videos with action graphs, the goal is now to assess their distance. To do so, we capitalize on the Graph Edit Distance (GED) [7, 17, 54] which is appropriate for assessing the structural dissimilarity between attributed graphs [8]. GED is defined based on the notion of optimal edit path, that is, a sequence of edit operations of minimum total

cost, transforming a source graph into a target one. Such edit operations regard the insertion, deletion and substitution of graph nodes or edges that suffice to transform one graph to the other. All edit operations are performed on the source graph. For example, in order to transform the "red" graph to the "blue" graph in Figure 2, one needs to remove the red faded-out nodes, maintain those in correspondence and insert the blue faded-out ones. The corresponding operations on edges are then deduced from each node's connectivity in the original graphs. Real, non-negative costs need to be defined for each such operation. Several edit paths may exist between two given graphs. Selecting the optimal one for the exact computation of the GED is NP-hard. Therefore, we employ BP-GED [50], a method that estimates an approximate GED based on a fast bipartite optimization procedure. Given two graphs G_A and G_B, BP-GED constructs a special, complete bipartite graph G_{AB}. Estimating the GED of G_A and G_B amounts to solving an assignment problem on G_{AB}. This is achieved by the Kuhn-Munkres algorithm [37]. In the worst case, the maximum number of operations required by this method is $O(n^3)$, where n is the total number of nodes of G_A and G_B.

Let A and B be two videos and G_A=(V_A, E_A), G_B=(V_B, E_B), $|V_A|$=N_A, $|V_B|$=N_B, be their respective action graphs, as shown in Figure 2. The corresponding cost/dissimilarity matrices are W_A ($N_A \times N_A$) and W_B (size $N_B \times N_B$), and are computed based on Eq.(1). In order to compute GED, besides W_A and W_B, BP-GED requires the definition of the complete bipartite graph $G_{AB} = (V_{AB}, E_{AB})$. This is defined as follows. We set V_{AB}=$V_A \cup V_B$ and $E_{AB} = H$, where $H = \{(u_i, v_j) | u_i \in V_A, v_j \in V_B\}$ is the set of edges of the complete bipartite graph with one end in V_A and the other in V_B. W_{AB} is a $(N_A + N_B) \times (N_A + N_B)$ weight matrix of the form: $W_{AB} = \begin{bmatrix} 0_{N_A,N_A} & W_H \\ W_H^T & 0_{N_B,N_B} \end{bmatrix}$. $0_{x,y}$ stands for an $x \times y$ matrix of zeros. W_H is a $N_A \times N_B$ matrix, called the bi-adjacency matrix of G_{AB}. The weights of W_H are also established based on Eq.(1) and represent the dissimilarities (semantic + motion-based) of all pairs of objects in videos A and B. Finally, BP-GED requires the definition of the costs of the edit operations (node/edge insertion, deletion and substitution), which are described below.

Costs of node edit operations in G_{AB}: The costs of the deletion, insertion and substitution of nodes are defined as follows, where ε denotes the empty node:

$$c(u_i \to \varepsilon) = \tau, \quad c(\varepsilon \to v_j) = \tau, \quad c(u_i \to v_j) = \left[\frac{1}{2\tau} + exp(-\alpha \cdot W_H(i,j) + \sigma) \right]^{-1}. \quad (4)$$

According to Eq.(4), the cost $c(u \to \varepsilon)$ of deleting a node u is constant (equal to τ) and equal to the cost $c(\varepsilon \to v)$ of inserting a node v. The cost $c(u \to v)$ of substituting the node $u_i \in V_A$ by node $v_j \in V_B$ is defined via the sigmoid function of the weighted dissimilarity value $W_H(i,j)$, computed based on Eq.(1). The parameters of these costs were empirically set equal to $\tau = 3$, $\alpha = 5$, $\sigma = 0$ based on experimental tests carried out. Methods for learning these parameters automatically from a set of sample graphs are also available [17]. The substitution function $c(u_i \to v_j)$ in Eq.(4) takes values in the range $[0, 2\tau]$. This secures that the set of cost functions satisfies the necessary conditions (non-negativity, triangular inequality) for GED to be a distance metric [49, 56].

Costs of edge edit operations G_{AB}: The costs of the deletion and insertion of edges are defined as it is for nodes. The edge substitution cost is defined as

$$c(e_{ij}^A \to e_{kl}^B) = \left[\frac{1}{2\tau} + exp\left(-\alpha \cdot (W_A(i,j) + W_B(k,l))/2 + \sigma\right) \right]^{-1}. \quad (5)$$

The cost $c(e_{ij}^A \to e_{kl}^B)$ of substituting the edge $e_{ij}^A \in E_A$ by edge $e_{kl}^B \in E_B$ is defined via the

Co-segmentation variant	S-EVACO-GT		S-EVACO		EVACO	
Methods / Feature types	$\lambda=0$	$\lambda=0.5$	$\lambda=0$	$\lambda=0.5$	$\lambda=0$	$\lambda=0.5$
(A) Naive matching	58.7	61.9	41.3	53.4	47.1	48.6
(B) GreedyED [16]	59.9	68.3	54.7	66.0	53.7	58.2
(C) DTW+GreedyED	-	-	-	-	55.2	64.2
(D) DTW+GED	-	-	-	-	67.3	72.9
(E) DTW	-	-	-	-	63.5	-
Ours	**85.9**	**90.9**	**82.2**	**88.2**	**81.5**	**87.5**

Table 1: Results for the task of action matching in triplets of videos based on the CAD-120 dataset [20]. The classification accuracy (%) scores are reported for the case of using motion ($\lambda=0$) / motion+semantics ($\lambda=0.5$) features and for three co-segmentation variants (S-EVACO-GT, S-EVACO, EVACO) see Section 4 for details). Empty cells correspond to meaningless combinations of methods and experimental settings.

sigmoid function of the weighted mean of the affinity values $W_A(i,j), W_B(k,l)$, computed based on Eq.(1).The parameters τ, α and σ are the same as for the edit costs for nodes.

Action distance: The GED of two graphs G_A and G_B establishes the sequence of graph edit operations (edit path) of minimum cost that transforms G_A into G_B. The bipartite GED between these two graphs computed by BP-GED and noted as $\mathcal{B}_{GED}(G_A,G_B)$, approximates the exact GED. The dissimilarity $\mathcal{D}(A,B)$ of the two videos A and B represented by the action graphs G_A and G_B is $\mathcal{B}_{GED}(G_A,G_B)$, normalized by the total number of objects involved, i.e.:

$$\mathcal{D}(A,B) = \mathcal{B}_{GED}(G_A,G_B)/(N_A+N_B). \tag{6}$$

The near-optimal edit path computed by BP-GED provides an assignment between the nodes of the original graphs that justifies the assessed quantitative measure of action similarity.

4 Experimental Results

We evaluate the proposed method on the tasks of (i) pairwise matching of actions in triplets, (ii) ranking of action pairs in triplets and (iii) nearest-neighbor action classification. Moreover, the cross-video object associations identified by our method are qualitatively assessed.

The evaluation employs the Cornell Activity Dataset (CAD-120)[2] [20, 29] that provides annotation data for 120 RGB-D videos of long daily activities, including 10 high-level activities and 10 sub-activity labels, temporal segmentation of each sequence into sub-activities, 12 object affordance labels and 8 categorical object labels. Each type of activity is performed 3 times by each of 4 different individuals in different environments, employing different classes of objects. Videos have been recorded from a variety of camera views. We use the ground truth annotation regarding the semantic labels, the bounding boxes, the 3D poses of the scene objects as well as the labels and the 3D positions of the tracked human body joints. We encode the trajectory of each body joint and of each object in a video with its relative 3D position per frame with respect to the torso joint which acts as a reference point.

In our implementation of the co-segmentation method of [42], the fast DTW algorithm [53] of linear time and space complexity is used. We employed both the unsupervised (EVACO) and the supervised (S-EVACO) variants. EVACO identifies automatically the number of common sub-sequences of the time series it compares. S-EVACO requires this as input,

[2] http://pr.cs.cornell.edu/humanactivities/data.php

Classification task	Action matching		Action pair ranking	
Our method / Co-segmentation variant	S-EVACO	EVACO	S-EVACO	EVACO
Random selection	33.3	33.3	16.6	16.6
Motion only ($\lambda = 0$)	48.5	48.0	25.5	24.3
Motion+semantics ($\lambda = 0.5$)	**64.0**	**63.2**	**34.2**	**36.0**

Table 2: Classification accuracy for the tasks of action matching in 6.480 triplets and ranking of action pairs in 1.920 triplets of CAD-120 videos. We consider supervised (S-EVACO) and unsupervised (EVACO) estimation of the common segments in a pair of videos.

which we provide based on the available labels of the action units per video in the CAD-120 dataset. In a 3rd setting (S-EVACO-GT), S-EVACO does not operate over the whole duration of the videos but within the time intervals resulting from the ground-truth-based temporal segmentation of the videos. In all experiments, both EVACO and S-EVACO use 32 generations of 64 PSO particles to search for common sub-sequences in the behavior of tracked objects. The search range for the sub-sequences length was set to $[25..500]$ frames.

Action matching in triplets of videos: We consider the task of action matching in a triplet of videos and evaluate the performance of our method against a set of baseline methods. Towards this end, the 120 action video clips of the CAD-120 dataset are combined to generate a set of $6,480$ distinct video triplets. Each triplet comprises three videos $\{A, A+, B\}$. A and $A+$ belong to the same action class a and are performed by two different subjects, while B is of any action class b except a and is performed by any of those two subjects. We considered the cases of $\lambda = 0$ (motion information, only) and $\lambda = 0.5$ (balanced combination of motion and semantic information) in Eq.(1) (definition of edge costs in the action graphs). If the value of λ was set equal to 1, action similarity in two videos would only rely on the semantic relatedness of the labels of the tracked body joints and the interacting objects.

We compared the proposed approach against the following baseline methods. *(A) Naive matching:* We perform bipartite matching by selecting the minimum bipartite edge weights row-wise in the bipartite adjacency matrix of the two action graphs. The average value of the selected edge weights is normalized by the number of nodes in the bipartite graph and is used as a measure of dissimilarity between the videos. *(B) GreedyED:* We apply the Greedy Edit Distance approach [16], instead of the BP-GED that computes another approximation of the exact GED in quadratic time (BP-GED runs in cubic-time). *(C) DTW-GreedyED:* We use the Dynamic Time Warping (DTW) algorithm (also used as a core component in [42]), to populate the bipartite adjacency matrix by comparing the full trajectories of each pair of objects in the two videos. The GreedyED approach is applied to score action dissimilarity between videos, normalized by the number of nodes of the bipartite graph. *(D) DTW-GED:* The same as (C), but using the BP-GED approach [50]. *(E) DTW:* We use the DTW to directly compare two videos based on their pairwise distance matrix (use the Euclidean distance between their frame-wise feature vectors generated based only on body joint features). The normalized DTW path score is used as a distance measure between videos.

We measure the classification accuracy, defined as the percentage of triplets in which $A+$ is ranked higher in similarity than B, given A as query. The results are summarized in Table 1, verifying that the performance of the proposed method is considerably better than any of the baseline methods, under any experimental setting. For the proposed method, the difference between using supervised co-segmentation (S-EVACO) compared to unsupervised (EVACO) is less than 1% - this means that temporal unsupervised co-segmentation operates very effectively within the proposed framework. Moreover, the use of GED in (D) compared to GreedyED in (C) provides a notable improvement of accuracy. Finally, the use of semantic

Method	(S) Koppula [22]	(S) Koppula [23]	(S) Wang [32]	(S) Hu [20]	(S) Rybok [52]	(S) Tayyub [57]	(S) Koperski [27]	(S) Lin [34]	(S) Duckworth [19]	(U) Mici [36]	**Ours** $\lambda=0$	**Ours** $\lambda=0.5$
GT	84.7	93.5	81.2	93.0	-	-	-	-	-	-	80.8	**88.3**
No GT	75.0	83.1	-	85.2	78.2	95 / 76	85.5	90	81	79	78.7	**84.2**

Table 3: Comparison of our 1-NN action classification with state of the art (S)upervised and (U)nsupervised methods. Two cases are considered: (a) In the first row (noted as GT) the ground truth annotations regarding temporal segmentation of the CAD-120 videos is considered as known, thus exploited by our method and (b) in the second row (noted as NO GT) where ground truth temporal segmentation is not used.

information ($\lambda = 0.5$ in Eq.(1)) improves the motion-based-only ($\lambda = 0$) results by 6%.

Ranking of action pairs in triplets of videos: Given a triplet of videos, three pairs can be defined. We are interested in ranking these three pairs according to their similarity, as this is computed by the proposed method. We consider the case where each of the videos in a triplet, shows actions from different classes. To the best of our knowledge, there is no available ground-truth-annotated dataset that is tailored to this task. Therefore, we generated 10 templates of action triplets. For each triplet, 3 different action classes were randomly selected from the CAD-120 dataset (each action class is selected in at least two triplets). We then asked 10 individuals to annotate these triplets by identifying the rank-1 and rank-2 pair of videos according to action similarity, and to justify their selection. Based on the 10 templates of action triplets, we consider combinations of subjects and action instances in the CAD-120 to generate a set of $1,920$ triplets on which the proposed method was evaluated.

We measured the percentage of triplets for which (a) only the top-ranked pair (action matching) and (b) the total rank ordering (action ranking) per triplet are correctly estimated. The obtained results are presented in Table 2. On average, the supervised co-segmentation (S-EVACO) results in marginal (1%) improvement compared to unsupervised (EVACO). However, the use of semantic information ($\lambda = 0.5$ in Eq.(1)) is crucial, improving the accuracy from 9% to 16%, over the motion-based-only ($\lambda = 0$) results.

Nearest-neighbor action classification: To demonstrate the capability of the proposed method in recognizing actions in an unsupervised manner, we perform nearest-neighbor classification of actions in the CAD-120 dataset. Thus, each action video is used as a query and compared to any other of the dataset, excluding the action instances recorded by the same subject, similar to the leave-one-person-out cross validation protocol of the various supervised schemes reported in Table 3. The GED-based scores between the query and the compared action videos is used as a metric to classify the former. The experimental results and a comparison with state of the art supervised and unsupervised approaches (not necessarily following the 1-NN scheme) are presented in Table 3. It can be observed that when motion-only information is used ($\lambda = 0$), our approach performs comparably to the performance of the recently proposed unsupervised method in [36]. By integrating semantic information of the object ($\lambda = 0.5$) improves the performance of our unsupervised framework by a great margin and brings it close to the top performing supervised methods.

Action similarity explained: Figure 3 shows node assignments and the BP-GED scores. In the top left, we consider a video triplet. In two of the videos, $(A, A+)$ two persons are "having

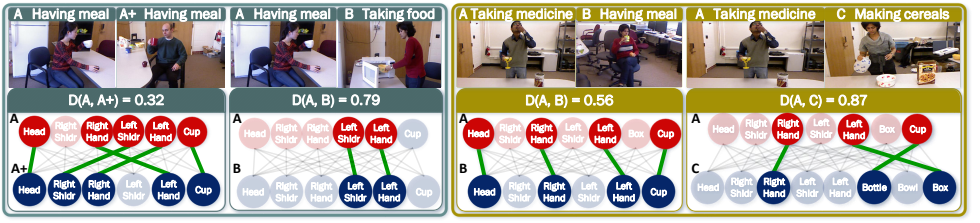

Figure 3: Qualitative analysis on the node assignments and the BP-GED results of the proposed method, for the task of action matching (left) and of ranking action pairs (right).

meal" while in the 3rd video (B) "a person pulls a dish out of the oven". Using motion information only ($\lambda = 0$), our method estimates that $\mathcal{D}(A,A+) = 0.32$ and $\mathcal{D}(A,B) = 0.79$, while $\mathcal{D}(A+,B)$ (not shown) is larger. Although no semantic information is used, in the most similar pair of $(A,A+)$, our method establishes meaningful correspondences (head/head, left-shoulder/right-shoulder, left-hand/right-hand, cup/object) as the two persons handle, at different times of the video, different objects, in the same manner, but with different hands. In the (A,B) pair, the established correspondences are still meaningful, but fewer.

The second example regards the task of ranking action pairs in a triplet of videos from different action classes, using $\lambda = 0.5$ (motion+semantics). In this example, the three videos are A:"taking medication", B:"having meal" and C:"preparing cereals". Our method estimated $\mathcal{D}(A,B) = 0.56$ and $\mathcal{D}(A,C) = 0.87$, as the rank-1 and rank-2 pairs of videos, respectively, according to action similarity. Indeed, the extracted result that "taking medication" is closer to "having meal" than it is to "preparing cereals" in the CAD-120 dataset and also the overall ranking order of the three action pairs in this triplet of videos is in agreement with the information conveyed by the human annotators for this action similarity task. More qualitative results which also showcase the temporal aspects of the established object correspondences are available online [3]. Moreover, details on the action ranking annotation task, datasets and source code are available in the online project page [4].

5 Summary

We proposed an unsupervised method for assessing the similarity of videos by estimating the relatedness of object classes based on WordNet and of their behavior based on temporal action co-segmentation. This permits the modeling of a video as an action graph. The computation of the Graph Edit Distance between two such graphs assesses the (dis)similarity of the corresponding videos. As a byproduct, this also establishes meaningful statio-temporal associations of objects in the compared videos that serve as an explanation of the quantitative assessment of video similarity. Extensive experiments on three action recognition tasks based on the CAD-120 dataset and in comparison with existing methods demonstrate that our unsupervised approach performs favorably to state of the art supervised methods.

Acknowledgments: This work was supported by EU H2020 project Co4Robots (Grant No 731869).

[3] https://youtu.be/QUHYa72fFt0
[4] https://www.ics.forth.gr/cvrl/video_similarity/

References

[1] Z. Abu-Aisheh, B. Gauzere, S. Bougleux, J. Y. Ramel, L. Brun, R. Raveaux, P. Heroux, and S. Adam. Graph edit distance contest: Results and future challenges. *Pattern Recognition Letters*, 100:96 – 103, 2017.

[2] F. Baradel, N. Neverova, C. Wolf, J. Mille, and G. Mori. Object Level Visual Reasoning in Videos. In *Proc. ECCV*, 2018.

[3] L. Baraldi, M. Douze, R. Cucchiara, and H. Jégou. Lamv: Learning to align and match videos with kernelized temporal layers. *Proc. IEEE CVPR*, pages 7804–7813, 2018.

[4] G. Bertasius, H. Soo Park, S. X Yu, and J. Shi. Am i a baller? basketball performance assessment from first-person videos. In *Proc. IEEE ICCV*, 2017.

[5] E. Z. Borzeshi, R. Xu, and M. Piccardi. Automatic human action recognition in videos by graph embedding. In *International Conference on Image Analysis and Processing*. Springer, 2011.

[6] E. Z. Borzeshi, M. Piccardi, K. Riesen, and H. Bunke. Discriminative prototype selection methods for graph embedding. *Pattern Recognition*, 46(6):1648–1657, 2013.

[7] H. Bunke and G. Allermann. Inexact graph matching for structural pattern recognition. *Pattern Recognition Letters*, 1(4):245–253, 1983.

[8] H. Bunke and X. Jiang. *Graph Matching and Similarity*. Springer US, 2000.

[9] E. F. Can and R. Manmatha. Formulating action recognition as a ranking problem. In *Proc. IEEE CVPR Workshops*, 2013.

[10] O. Çeliktutan, C. Wolf, B. Sankur, and E. Lombardi. Fast exact hyper-graph matching with dynamic programming for spatio-temporal data. *J. Math. Imaging Vis.*, 51(1): 1–21, 2015.

[11] C. Y. Chang, D. A. Huang, Y. Sui, L. Fei-Fei, and J. C. Niebles. D3tw: Discriminative differentiable dynamic time warping for weakly supervised action alignment and segmentation. In *Proc. IEEE CVPR*, 2019.

[12] H. Doughty, D. Damen, and W. Mayol-Cuevas. Who's better? who's best? pairwise deep ranking for skill determination. In *Proc. IEEE CVPR*, 2018.

[13] H. Doughty, W. Mayol-Cuevas, and D. Damen. The pros and cons: Rank-aware temporal attention for skill determination in long videos. In *Proc. IEEE CVPR*, 2019.

[14] P. Duckworth, D. C. Hogg, and A. G. Cohn. Unsupervised human activity analysis for intelligent mobile robots. *Artificial Intelligence*, 270:67 – 92, 2019.

[15] C. Fellbaum. Wordnet and wordnets. In Alex Barber, editor, *Encyclopedia of Language and Linguistics*, pages 2–665. Elsevier, 2005.

[16] A. Fischer, K. Riesen, and H. Bunke. Improved quadratic time approximation of graph edit distance by combining hausdorff matching and greedy assignment. *Pattern Recognition Letters*, 87, 2017. Advances in Graph-based Pattern Recognition.

[17] X. Gao, B. Xiao, D. Tao, and X. Li. A survey of graph edit distance. *Pattern Analysis and Applications*, 13(1):113–129, 2010.

[18] R. Girdhar, J. Carreira, C. Doersch, and A. Zisserman. Video Action Transformer Network. In *Proc. IEEE CVPR*, 2019.

[19] M. Guo, E. Chou, D. A. Huang, S. Song, S. Yeung, and L. Fei-Fei. Neural graph matching networks for fewshot 3d action recognition. In *Proc. ECCV*, 2018.

[20] R. Gupta H. S. Koppula and A. Saxena. Learning human activities and object affordances from rgb-d videos. *The International Journal of Robotics Research*, 32(8): 951–970, 2013.

[21] S. Herath, M. T. Harandi, and F. M. Porikli. Going deeper into action recognition: A survey. *Image Vision Computing*, 60:4–21, 2017.

[22] N. Hu, G. Englebienne, Z. Lou, and B. Krose. Latent hierarchical model for activity recognition. *IEEE Transactions on Robotics*, 31:1472–1482, 2015.

[23] A. Jain, A. R. Zamir, S. Savarese, and A. Saxena. Structural-rnn: Deep learning on spatio-temporal graphs. In *Proc. IEEE CVPR*, 2016.

[24] T. N. Kipf and M. Welling. Semi-supervised classification with graph convolutional networks. *ArXiv*, abs/1609.02907, 2017.

[25] O. Kliper-Gross, T. Hassner, and L. Wolf. One shot similarity metric learning for action recognition. In *Proc. SIMBAD'11*, 2011.

[26] O. Kliper-Gross, T. Hassner, and L. Wolf. The action similarity labeling challenge. *IEEE Trans. Pattern Anal. Mach. Intell.*, 34(3):615–621, 2012.

[27] M. Koperski and F. Bremond. Modeling spatial layout of features for real world scenario rgb-d action recognition. In *Proc. IEEE AVSS*, 2016.

[28] H. S. Koppula and A. Saxena. Learning spatio-temporal structure from rgb-d videos for human activity detection and anticipation. In *Proc. ICML*. JMLR.org, 2013.

[29] H. S. Koppula and A. Saxena. Anticipating human activities using object affordances for reactive robotic response. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(1), 2016.

[30] I. Kotsia and I. Patras. Exploring the similarities of neighboring spatiotemporal points for action pair matching. In *Proc. ACCV*, 2013.

[31] R. Krishnan and S. Sarkar. Similarity measure between two gestures using triplets. In *Proc. IEEE CVPR Workshops*, 2013.

[32] R. Krishnan and S. Sarkar. Conditional distance based matching for one-shot gesture recognition. *Pattern Recognition*, 48(4):1302 – 1314, 2015.

[33] Z. Li, Y. Huang, M. Cai, and Y. Sato. Manipulation-skill assessment from videos with spatial attention network. *arXiv preprint arXiv:1901.02579*, 2019.

[34] L. Lin, K. Wang, W. Zuo, M. Wang, J. Luo, and L. Zhang. A deep structured model with radius–margin bound for 3d human activity recognition. *International Journal of Computer Vision*, 118(2):256–273, 2016.

[35] E. Loper and S. Bird. Nltk: the natural language toolkit. *arXiv preprint cs/0205028*, 2002. URL https://www.nltk.org/.

[36] L. Mici, G. I. Parisi, and S. Wermter. A self-organizing neural network architecture for learning human-object interactions. *Neurocomputing*, 307:14 – 24, 2018.

[37] J. Munkres. Algorithms for the assignment and transportation problems. *Journal of the Society for Industrial and Applied Mathematics*, 5(1):32–38, 1957.

[38] M. Neuhaus and H. Bunke. Self-organizing maps for learning the edit costs in graph matching. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 35:503–514, 2005.

[39] M. Neuhaus and H. Bunke. *Bridging the Gap Between Graph Edit Distance and Kernel Machines*. World Scientific Publishing Co., Inc., 2007.

[40] M. Neuhaus and H. Bunke. Automatic learning of cost functions for graph edit distance. *Information Sciences*, 177(1):239–247, 2007.

[41] B. Ni and P. Moulin. Manipulation pattern discovery: A nonparametric bayesian approach. In *Proc. IEEE ICCV*, 2013.

[42] K. Papoutsakis, C. Panagiotakis, and A. A Argyros. Temporal action co-segmentation in 3d motion capture data and videos. In *Proc. IEEE CVPR*, 2017.

[43] X. Peng, Y. Qiao, Q. Peng, and Q. Wang. Large margin dimensionality reduction for action similarity labeling. *IEEE Signal Processing Letters*, 21(8):1022–1025, 2014.

[44] H. Pirsiavash, C. Vondrick, and A. Torralba. Assessing the quality of actions. In *Proc. ECCV*. Springer, 2014.

[45] S. Qi, W. Wang, B. Jia, J. Shen, and S. C. Zhu. Learning human-object interactions by graph parsing neural networks. In *Proc. ECCV*, 2018.

[46] J. Qin, L. Liu, Z. Zhang, Y. Wang, and L. Shao. Compressive sequential learning for action similarity labeling. *IEEE Transactions on Image Processing*, 25(2):756–769, 2016.

[47] J. Redmon and A. Farhadi. Yolo9000: Better, faster, stronger. *Proc. IEEE CVPR*, 2017.

[48] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(06):1137–1149, 2017.

[49] K. Riesen. Structural pattern recognition with graph edit distance. *Advances in computer vision and pattern recognition*, 2015.

[50] K. Riesen and H. Bunke. Approximate graph edit distance computation by means of bipartite graph matching. *Image Vision Computing*, 27:950–959, 2009.

[51] K. Riesen and H. Bunke. Improving bipartite graph edit distance approximation using various search strategies. *Pattern Recognition*, 48(4):1349 – 1363, 2015.

[52] L. Rybok, B. Schauerte, Z. Al-Halah, and R. Stiefelhagen. "important stuff, everywhere!" activity recognition with salient proto-objects as context. In *Proc. IEEE WACV*, 2014.

[53] S. Salvador and P. Chan. Toward accurate dynamic time warping in linear time and space. *Intell. Data Anal.*, 11(5):561–580, 2007.

[54] A. Sanfeliu and K. Fu. A distance measure between attributed relational graphs for pattern recognition. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-13 (3):353–362, 1983.

[55] F. Serratosa. Fast computation of bipartite graph matching. *Pattern Recognition Letters*, 45:244 – 250, 2014.

[56] F. Serratosa. Graph edit distance: Restrictions to be a metric. *Pattern Recognition*, 90: 250 – 256, 2019.

[57] J. Tayyub, A. Tavanai, Y. Gatsoulis, A. G Cohn, and D. C Hogg. Qualitative and quantitative spatio-temporal relations in daily living activity recognition. In *Proc. ACCV*, 2014.

[58] J. Wan, Q. Ruan, W. Li, and S. Deng. One-shot learning gesture recognition from rgb-d data using bag of features. *Journal of Machine Learning Research*, 14:2549–2582, 2013.

[59] K. Wang, X. Wang, L. Lin, M. Wang, and W. Zuo. 3d human activity recognition with reconfigurable convolutional neural networks. In *Proceedings of the 22nd ACM International Conference on Multimedia*, MM '14, pages 97–106, 2014.

[60] P. Wang, L. Liu, C. Shen, Z. Huang, A. v. d. Hengel, and H. T. Shen. Multi-attention network for one shot learning. In *Proc. IEEE CVPR*, 2017.

[61] P. Wang, W. Li, P. Ogunbona, J. Wan, and S. Escalera. Rgb-d-based human motion recognition with deep learning: A survey. *Computer Vision and Image Understanding*, 171:118 – 139, 2018.

[62] X. Wang and A. Gupta. Videos as space-time region graphs. In *Proc. ECCV*, 2018.

[63] L. Wolf, T. Hassner, and Y. Taigman. The one-shot similarity kernel. In *Proc. IEEE ICCV*, 2009.

[64] C. Wu, J. Zhang, O. Sener, B. Selman, S. Savarese, and A. Saxena. Watch-n-patch: Unsupervised learning of actions and relations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(2), 2018.

[65] Z. Wu and M. Palmer. Verb semantics and lexical selection. In *ACL*, 1994.

[66] T. Dullien O. Vinyals Y. Li, C. Gu and P. Kohli. Graph matching networks for learning the similarity of graph structured objects. In *Proc. ICML*.

[67] B. Yao, A. Khosla, and L. Fei-Fei. Classifying actions and measuring action similarity by modeling the mutual context of objects and human poses. *a) A*, 1, 2011.

[68] M. Ye and Y. Guo. Deep triplet ranking networks for one-shot recognition. *arXiv preprint arXiv:1804.07275*, 2018.

[69] S. P. Zafeiriou and I. Kotsia. On one-shot similarity kernels: Explicit feature maps and properties. *Proc. IEEE ICCV*, 2013.