A New Adaptive Segmental Matching Measure for Human Activity Recognition

Shahriar Shariat, Vladimir Pavlovic
Computer Science Department, Rutgers University
Piscataway, New Jersey, USA 08854
{sshariat, vladimir}@cs.rutgers.edu

Abstract

The problem of human activity recognition is a central problem in many real-world applications. In this paper we propose a fast and effective segmental alignment-based method that is able to classify activities and interactions in complex environments. We empirically show that such model is able to recover the alignment that leads to improved similarity measures within sequence classes and hence, raises the classification performance. We also apply a bounding technique on the histogram distances to reduce the computation of the otherwise exhaustive search.

1. Introduction

Human activity recognition is an important yet difficult problem that has attracted the attention of many researchers in the field of computer vision (see [17] for a recent review). Human activity recognition is the central part of many applications such as video surveillance, human computer interfaces based on activity, robotics and so forth. There have been many instances of successful works in this area particularly when recognizing simple tasks such as walking and running [13, 4]. As the field matures, researchers have turned their attention on activities within more complex environments [20, 10]

Local spatio-temporal features have been widely and successfully used for activity recognition tasks [4, 13, 10, 9]. Invariance to affine transformation and robustness against noise and slight changes in environmental factors such as lighting are among the reasons that have made these features effective and popular. Different approaches have been built upon such features to classify and recognize simple and complex activities. Dollar et al [4] introduced a feature descriptor called 'Cuboid' which encompasses spatial and temporal features within small patches and then represents an activity using bag-of-words representation [16]. They used support vector machines (SVM) to classify videos containing each activity based on this representation. Niebles et al [7] have proposed a probabilistic latent semantic analysis coupled with cuboids to classify and recognize activities.

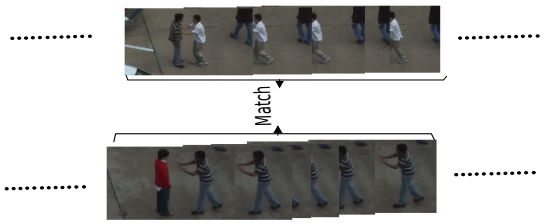


Figure 1. Similar activities can be recognized by matching the warped versions of their sub-sequences

Similar activities can be reasonably accurately characterized as different warped instances of basic activity patterns, provided that the feature extraction is robust to noise and slight changes in environmental factors (Figure 1). Alignment-based methods have been used in activity recognition especially for MoCap datasets, where the amount of noise is in general low and there is little to no ambiguity introduced by visual projections or visual clutter. The most common alignment algorithm, Dynamic Time Warping (DTW), has been successfully used in many applications [3]. In [21] authors propose and extension of DTW by introducing an spatial embedding through canonical correlation analysis (CTW) so that sequences of different modalities can be aligned and thus a better performance comparing to DTW in aligning MoCap sequences is achieved. The authors extend CTW in [22] by introducing Generalized Time Warping (GTW) to be able to align multiple sequences of different modalities efficiently by solving the objective function using Gauss-Newton algorithm.

In practice, alignment models are sensitive to noise which limits their application in real-world computer vision problems. In [14] we formulate the alignment problem as a monotonic canonical correlation analysis and introduce a segmental alignment model which is robust against noise when applied to MoCap data. In the followup work,[15], we propose a probabilist segmental alignment model (SPHMM) which exhibited good performance in the

presence of significant artificially added noise. Both works are based on the idea that aligning segments of sequences instead of sample-by-sample alignment can be more robust. These proposed models however, suffers from high computational requirements. Chu et al. in [2] propose a branch an bounding method to find common subsequences of two time-series and thus they extract matching segments by applying the algorithm repeatedly without respecting time monotonicity. Ryoo in [9] proposed a method for matching short intervals of sequences for classifying various visual activities. Due to computational cost the author proposed to approximate this matching by fixed segmentation of the sequences. Since the segmentation is fixed in [9], it is crucial to have a very good estimate of the boundaries of such segments but the paper does not provide such insight. Such approach may potentially reduce the computational cost in expense of loosing classification accuracy.

The contributions of this paper are two-fold:

- We propose a simplified segmental alignment model which consist of a single match operation and empirically show that it is able to approximate the true alignment of a pair of sequences.
- Using a bounding technique for histogram distances we reduce the computation time by a factor of two.

We build upon the idea in [15] for probabilistic adaptive segmental alignment and simplify it by introducing a gapless alignment model. The adaptive segmental alignment model is able to realize the boundaries of segments of the contrasting sequences and efficiently match them. We show in the experimental results that such model is able to achieve accurate alignment performance while significantly reducing the computational cost. The gap-less model enables us to further reduce the computation time by employing a bounding technique on histogram distances to prune many segmentations that may yield inferior alignments and thereby eliminate unnecessary computation. Essentially, we propose a bad-of-words model where the bags are inferred so as to maximize the sequence similarity.

The rest of the paper is as follows. We first explain our methodology in Section 2 where we detail our signal representation in Subsection 2.1 and our matching model in Subsection 2.2. Then the histogram distance bounding is described in Subsection 2.3 and its application in our segmental match model is explained in Subsection 2.4. Experimental results are discussed in Section 3 and the paper is finally concluded by Section 4.

2. Methodology

In this section we describe our approach towards segmental matching for activity recognition. We first detail our representation scheme and then describe the segmental matching and the bounding technique to reduce the computation time.

2.1. Representation

In this paper we adopt a common Bag of Temporal Words (BOTW) [16] representation for videos. BOTW is a popular representation that has been successfully used by researchers [8, 2]. In this representation extracted features are clustered into several codewords using a clustering method such as k-means. Similar features described by the same codeword are then counted together and form a histogram for a single or a collection of frames. Therefore, given a histogram map $\phi^H_{b_i:e_i}(.)$ and F, corresponding codewords of features extracted from contiguous segment of frames $b_i : e_i = (b_i, b_{i+1}, \ldots, e_{i-1}, e_i)$, we denote an H-bin histogram of such contiguous segment as $X_{b_i:e_i} = \phi^H_{b_i:e_i}(F)$. Throughout the paper we may refer to segments by their starting point, X_{b_i}, ending point X_{e_i} or their index X_i to simplify the notation.

2.2. Segmental Matching (SM)

Assume that given a video containing an action, proper features are extracted and the associated BOTW is represented by X as described in Sec. 2.1. Furthermore, $\mathcal{D} = \{(X^n, z^n)_{n=1}^N\}$ is a given training set of sequences, such that X^n contains the BOTW representation of videos containing activity label z^n. The objective is then to label X with the appropriate activity. More specifically, the most probable activity label, z^* is

$$z^*|X = \arg\max_{(z \in Z)} \max_{(X^n, z^n) \in \mathcal{D}} P(X, X^n) I(z, z^n) \quad (1)$$

where Z is the label set and $I(\cdot, \cdot)$ is the indicator function. In the rest of the paper we refer to the training sequence as Y to simplify the notation.

In [15] we proposed an effective way to maximize the joint likelihood of two contrasting sequences using an extension of a pair-HMM to construct an adaptive probabilistic segmental alignment model. The proposed model allows for aligning segments of sequences which perfectly maps to the problem of activity recognition where one seeks to find a collection or consecutive frames in the query video to match with a similar set of frames in the training set. The model however, is computationally demanding. Therefore, in this paper, we propose to remove the gap states (insertion and deletion) and thus obtain a single state HMM consisting of only a match state. In fact, we claim that a single match operation coupled with adaptive segmentation is able to approximate a full operation alignment model. This reduction not only removes the computation needed for gap states, but also enables us to bound the likelihood of alignment (c.f Sec. 2.3) and thus improve the performance even further.

Segmentation in this paper is defined as a tight partitioning of the sequence. Assume a fixed partitioning of a given sequence X into L intervals is provided. In the context of human activity recognition, we consider X to be the sequence of H-bin unnormalized histograms resulted from the mapping of the extracted features of each frame using ϕ^H from Sec. 2.1. This partitioning then defines $X = (X_1, X_2, ..., X_L)$. That is, $X_i = \sum_{f=b_i}^{e_i} x_f$ where x_f is the unnormalized histogram associated with the BOW representation of frame f. Throughout the paper, we denote the histogram of individual frames with lower case letters indexed by the frame number. Also note that we assume tight and non-overlapping segmentation. That is, assuming the input video has T frames, then for $X \in \mathbb{R}^{H\times T}$, we require $b_1 = 1$, $e_L = T$ and $b_{i+1} = e_i + 1, \forall i = [1 \dots L-1]$. Thus, given two sequences of histograms X and Y, we define segmentation $\mathbf{S} = (\mathbf{S}(X), \mathbf{S}(Y)) = ((X_1, X_2, \dots X_L), (Y_1, Y_2, \dots Y_L))$. For a clarification on notation look at Figure 2. Note that even thought we assume fixed segmentation and one-to-one matching, we do not need the boundaries to be the same and they will be determined through the estimation.

For a fixed segmentation the likelihood of matching two sequences is defined as

$$P(X, Y|\mathbf{S}) = \prod_{i=1}^{L} \exp\left(-\frac{1}{\sigma} D(X_i, Y_i)\right) \Psi(X_i, Y_i). \quad (2)$$

where $D(\cdot,\cdot)$ is a suitable distance metric such as l_1 or χ^2 and σ is a scaling parameter which in our experiments is set to 1. $\Psi(\cdot,\cdot)$ is a prior on segments. A non-uniform prior on segment matching can result into different alignments by ,for instance, favouring longer or shorter segments and their matching.

Our objective is thus to maximize the joint likelihood by iterating over all possible segmentations. That is,

$$P^*(X, Y) = \max_{\mathbf{S}} P(X, Y|\mathbf{S}) P(\mathbf{S}). \quad (3)$$

Consequently one may obtain the optimal segmentation as

$$\mathbf{S}^* = \arg\max_{\mathbf{S}} P(X, Y|\mathbf{S}) P(S) \quad (4)$$

where we assume uniform prior on segmentation. Using $P(S)$ one can define various types of bands usually used in alignment such as the Sakao-Chiba band [12] by relating the prior to the segment length and the position in the sequence.

To find such optimal segmentation one may search over all permissible segment lengths. This exhaustive search however, is very expensive and thus we propose a pruning technique inspired by [6, 2]. Such pruning is not possible on the full alignment model since the gap operations remove parts of either of the sequences and can affect any estimated or determined bounds on the matching.

2.3. Bounding Histogram Distances

Given the maximum segment length l_{max}, the minimum segment length l_{min}, and two segments of sequence X and Y, ending at e_i and e_j, respectively, we denote the maximum length segments by $\overline{X}_{e_i} = X_{e_i - l_{max}:e_i}$ and $\overline{Y}_{e_j} = Y_{e_j - l_{max}:e_j}$. Likewise, the minimum length segments are denoted by $\underline{X}_{e_i} = X_{e_i - l_{min}:e_i}$ and $\underline{Y} = Y_{e_j - l_{min}:e_j}$. We are aiming to bound the distance of the histogram features of any possible segment starting from $X_{b_i - l_{max}}$ extending to $X_{e_i}$and $Y_{e_j - l_{max}}$ extending maximally to Y_{e_i}. Note that even though we use the same l_{min} and l_{max} for both sequences, it is not a requirement of our method and is used only to simplify the notation. The bin counts of X_{e_i} and Y_{e_j} are bounded as

$$\underline{X}^h_{e_i} \le X^h_{e_i-k:e_i} \le \overline{X}^h_{e_i}, (l_{min} \le k \le l_{max}) \quad (5)$$

$$\underline{Y}^h_{e_j} \le Y^h_{e_j-z:e_j} \le \overline{Y}^h_{e_j}, (l_{min} \le z \le l_{max}) \quad (6)$$

where $X^h_{\cdot}$ and $Y^h_{\cdot}$ denote the histogram bin h.

One can easily extend (5, 6) to normalized histogram noting that $|\underline{X}_{e_i}| \le X_{e_i-k:e_i} \le |\overline{X}_{e_i}|$. That is,

$$\frac{\underline{X}^h_{e_i}}{|\overline{X}_{e_i}|} \le \hat{X}^h_{e_i-k:e_i} \le \frac{\overline{X}^h_{e_i}}{|\underline{X}_{e_i}|}, (l_{min} \le k \le l_{max}) \quad (7)$$

$$\frac{\underline{Y}^h_{e_j}}{|\overline{Y}_{e_j}|} \le \hat{Y}^h_{e_j-z:e_j} \le \frac{\overline{Y}^h_{e_j}}{|\underline{Y}_{e_i}|}, (l_{min} \le z \le l_{max}) \quad (8)$$

It is straightforward to observe

$$\begin{aligned} \min(\underline{X}^h_{e_i}, \underline{Y}^h_{e_j}) &\le \min(X^h_{e_i-k:e_i}, Y^h_{e_j-z:e_j}) \\ &\le \min(\overline{X}^h_{e_i}, \overline{Y}^h_{e_j}) \quad (9) \\ \max(\underline{X}^h_{e_i}, \underline{Y}^h_{e_j}) &\le \max(X^h_{e_i-k:e_i}, Y^h_{e_j-z:e_j}) \\ &\le \max(\overline{X}^h_{e_i}, \overline{Y}^h_{e_j}) \quad (10) \end{aligned}$$

for $l_{min} \le k, z \le l_{max}$. Following [2] one may construct the bounds on popular histogram distances. For completeness of presentation these bounds are included below.

Bounding l_1 distance: Noting that $|a-b| = \max(a,b) - \min(a,b)$ and a simple reordering of (9, 10) one can observe that

$$\begin{aligned} &\max(\underline{X}^h_{e_i}, \underline{Y}^h_{e_j}) - \min(\overline{X}^h_{e_i}, \overline{Y}^h_{e_j}) \\ &\le |X^h_{e_i-k:e_i} - Y^h_{e_j-z:e_j}| \le \\ &\max(\overline{X}^h_{e_i}, \overline{Y}^h_{e_j}) - \min(\underline{X}^h_{e_i}, \underline{Y}^h_{e_j}) \quad (11) \end{aligned}$$

for $l_{min} \le k, z \le l_{max}$. The bounds on l_1 distance are then the summation over all bins. That is,

$$l_b^{l_1}(X_{e_i}, Y_{e_j}, m, l) = \sum_{h=1}^{H} \max(\underline{X}^h_{e_i}, \underline{Y}^h_{e_j}) - \min(\overline{X}^h_{e_i}, \overline{Y}^h_{e_j}) \quad (12)$$

$$u_b^{l_1}(X_{e_i}, Y_{e_j}, m, l) = \sum_{h=1}^{H} \max(\overline{X}^h_{e_i}, \overline{Y}^h_{e_j}) - \min(\underline{X}^h_{e_i}, \underline{Y}^h_{e_j}) \quad (13)$$

and for normalized histograms

$$\hat{l}_b^{l_1}(X_{e_i}, Y_{e_j}, l_{min}, l_{max}) = \sum_{h=1}^{H} \left(\max\left(\frac{\underline{X}_{e_i}^h}{|\underline{X}_{e_i}^h|}, \frac{\underline{Y}_{e_j}^h}{|\underline{Y}_{e_j}^h|} \right) - \min\left(\frac{\overline{X}_{e_i}^h}{|\underline{X}_{e_i}^h|}, \frac{\overline{Y}_{e_j}^h}{|\underline{Y}_{e_j}^h|} \right) \right) \tag{14}$$

$$\hat{u}_b^{l_1}(X_{e_i}, Y_{e_j}, l_{min}, l_{max}) = \sum_{h=1}^{H} \left(\max\left(\frac{\overline{X}_{e_i}^h}{|\underline{X}_{e_i}^h|}, \frac{\overline{Y}_{e_j}^h}{|\underline{Y}_{e_j}^h|} \right) - \min\left(\frac{\overline{X}_{e_i}^h}{|\underline{X}_{e_i}^h|}, \frac{\overline{Y}_{e_j}^h}{|\underline{Y}_{e_j|}^h|} \right) \right). \tag{15}$$

Histogram intersection and χ^2 distances can also be derived in the same way.

Bounding histogram intersection distance: Histogram intersection distance is defined as

$$d_{\cap}(\phi_X^H, \phi_Y^H) = -\sum_{h=1}^{H} \min(\hat{X}^h, \hat{Y}^h) \tag{16}$$

using (7), (8) the corresponding lower and upper bound is

$$\hat{l}_b^{\cap}(X_{e_i}, Y_{e_j}, l_{min}, l_{max}) = -\sum_{h=1}^{H} \min\left(\frac{\overline{X}_{e_i}^h}{|\underline{X}_{e_i}^h|}, \frac{\overline{Y}_{e_j}^h}{|\underline{Y}_{e_j}^h|} \right) \tag{17}$$

$$\hat{u}_b^{\cap}(X_{e_i}, Y_{e_j}, l_{min}, l_{max}) = -\sum_{h=1}^{H} \min\left(\frac{\underline{X}_{e_i}^h}{|\overline{X}_{e_i}^h|}, \frac{\underline{Y}_{e_j}^h}{|\overline{Y}_{e_j}^h|} \right) \tag{18}$$

Bounding χ^2 distance: χ^2 distance is defined as

$$d_{\chi^2}(\phi_X^H, \phi_Y^H) = \sum_{h=1}^{H} \frac{\left(\hat{X}^h - \hat{Y}^h\right)^2}{\hat{X}^h + \hat{Y}^h}. \tag{19}$$

Using the normalized bounds on l_1 distance i.e. (14) and (15) one can easily prove

$$\hat{l}_b^{\chi^2}(X_{e_i}, Y_{e_j}, l_{min}, l_{max}) = \sum_{h=1}^{H} \frac{\left(\max(0, \hat{l}_b^{l_1})\right)^2}{\frac{\overline{X}_{e_i}^h}{|\underline{X}_{e_i}^h|} + \frac{\overline{Y}_{e_j}^h}{|\underline{Y}_{e_i}^h|}} \tag{20}$$

$$\hat{u}_b^{\chi^2}(X_{e_i}, Y_{e_j}, l_{min}, l_{max}) = \sum_{h=1}^{H} \frac{(\hat{u}_b^{l_1})^2}{\frac{\underline{X}_{e_i}^h}{|\overline{X}_{e_i}^h|} + \frac{\underline{Y}_{e_j}^h}{|\overline{Y}_{e_j}^h|}} \tag{21}$$

2.4. Fast Segmental Matching (Fast-SM)

We propose a recursive algorithm that starts matching from the end of the contrasting sequences. Each segmental matching is effectively finding the joint likelihood of x_i and y_i. Within each matching we search over all possible segmentation up to a maximum segment length. That is, given l_{max} and l_{min}, for $i = L, \dots 1$, $j = L, \dots 1$ and considering uniform prior on segments the likelihood of matching is

$$P(x_{e_i}, y_{e_j}) = \max_{l_{min} \le k,z \le l_{max}} \left\{ \exp(-D(X_{e_i-k:e_i}, Y_{e_j-z:e_j})) P(x_{e_i-k-1}, y_{e_j-z-1}) \right\}. \tag{22}$$

In other words, (22) is the maximum likelihood of all possible segmentations limited by l_max and l_min. Thus, we search for all segmenations ending in x_{e_i} and y_{e_j} multiplied by the likelihood of the matching up to the starting point of those segments.

We assume that likelihood of correspondences in the local neighborhood is approximately constant. Therefore, before executing a recursion we examine its approximated likelihood against the best one found so far. We abuse the notation and redefine P^* as the maximal likelihood calculated for the immediate preceding segment to $(X_{e_i-k:e_i}, Y_{e_j-z:e_j})$, we have

$$P^* = P(x_{e_i-k-1}, y_{e_j-z-1}) \cdot exp(D(X^*_{b_{i-1}:e_{i-1}}, Y^*_{b_{j-1}:e_{j-1}})) \tag{23}$$

where $X^*_{b_{i-1}:e_{i-1}}$ and $Y^*_{b_{j-1}:e_{j-1}}$ denote the best $i-1$ and $j-1$ segments that naturally extent to $e_i - k - 1$ and $e_j - z - 1$, respectively. Therefore, P^* is the optimal segmentation and matching from the beginning of the sequences up to segments $i-1$ and $j-1$ (excluding those segments). Note that all elements required to compute P^* is already calculated and no extra effort is needed to determine it. The bounding is then defined as

$$\tilde{P}(x_{e_i-k-1}, y_{e_j-z-1}) \le P^* \exp(-l_b(Y_{e_i-k-1}, Y_{e_j-z-1}, l_{min}, l_{max})) \tag{24}$$

where l_b is the corresponding lower bound defined in subsection 2.3. The idea is illustrated in Figure 2. That is, we propose to bound the likelihood of a segment by the the production of the maximal likelihood in its neighborhood and the upper bound on the likelihood of matching any two segments extended within its boundaries. Therefore, using (24) one can obtain an approximated upper bound on $P(x_{e_i-k-1}, y_{e_j-z-1})$ and compare it against the best likelihood obtained for the previous segment. We use the term "*approximated upper bound*" since we have made the assumption of smoothness on the local likelihood. If $\tilde{P}(y_{e_i-k-1}, y_{e_j-z-1})$ is lower than the best likelihood for the preceding segment obtained so far then we do not expand the recursion and set that correspondence likelihood

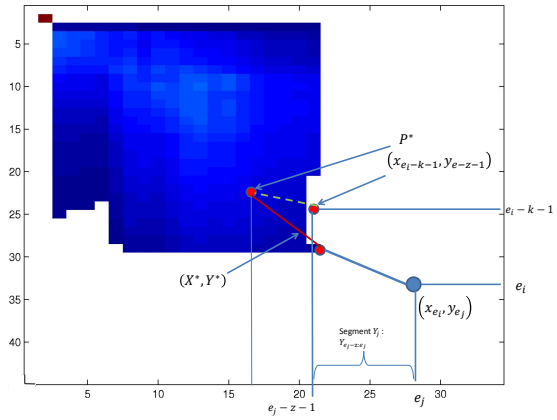


Figure 2. Approximate bounding of the likelihood. Axes show the index (time) of contrasting sequences. At segment (X_i, Y_j) we are verifying whether we should consider the new segment extending from $(x_{e_i} - k - 1, y_{e_j} - z - 1)$. So far in the process, the best likelihood is achieved by connecting to segment (X^*, Y^*). Therefore, we can find P^* which is the likelihood of segmentation up to the beginning of (X^*, Y^*). Then we assume the smoothness (almost constant likelihood) on the neighbourhood of (X^*, Y^*) and extend a hypothetical segment to $(x_{e_i} - k - 1, y_{e_j} - z - 1)$. The distance associated with that hypothetical segment can then be bounded and contribute to our approximated bound on the likelihood of all possible segmentation up to $(x_{e_i} - k - 1, y_{e_j} - z - 1)$.

to its minimum by

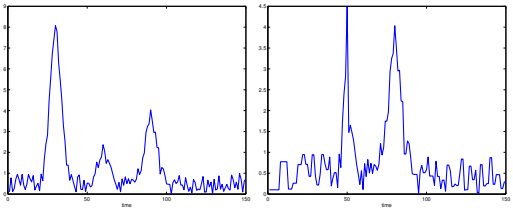
$$P(x_{e_i-k-1}, y_{e_j-z-1}) = P^* \exp(-u_b(X_{e_i-k-1}, Y_{e_j-z-1}, l_{min}, l_{max})). \quad (25)$$

By setting $P(x_{e_i-k-1}, y_{e_j-z-1})$ to the minimum likelihood we avoid further expansion of this path even if this point is visited again during the segmentation. Using this bounding technique approximately half of the required computations could be pruned away in the experiments as evident by the speedup gains demonstrated in the Section 3.

Another technique that contributes to improving the computational performance of our approach stems from the BOTW representation. This representation allows us to use the idea of *integral image* [18] to calculate the cumulative sum of the histograms and thus obtain the required segment using a single subtraction operation. That is, if R is a sequence of such cumulative sums ($R_f = R_{f-1} + x_f$ $1 \leq f \leq T$ for a video of T frames) one can obtain a segment from b_i to e_i simply by $R_{b_i:e_i} = R_{e_i} - R_{b_i-1}$.

3. Experiments

In this section we demonstrate conclusive empirical results on the utility of our approach. We show that a single state alignment model coupled with segmentation is able to approximate the true alignment of sequences. We also show that the proposed bounding technique is effective in reducing the required computation while keeping the model accurate. The first experiment is on synthetic data and verifies our claim that a segmental matching is able to effectively approximate the true alignment of two sequences while keeping the computational cost low. The second experiment is on Motion Capture data by which we show that our method is able to classify such data better than a rival method. Finally, our results on UT-iteration dataset [11] are presented and analyzed.



(a) Original (b) Warped

Figure 3. An instance of a generated sequence and its corresponding warped sequence.

3.1. Synthetic Data

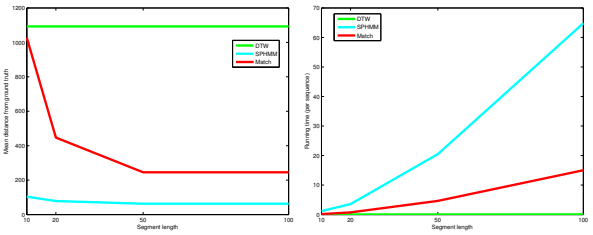
To show that our adaptive segmental match model is able to approximate a complete alignment model we have designed the following synthetic experiment. 100 sequences are generated from the model

$$Q_j(t) = \sum_{i=1}^{3} (\pi_i + \nu_t) \exp\left((t - \mu)^2\right) + \omega_t \quad (26)$$

The time length of all sequences is 150. Peaks in the sequences occur at mean times $\mu = [30, 60, 90]$. The weights are set to $\pi = [7, 1, 3]$ and are corrupted by white independent noise. $\omega_t, \nu_t = N(0, 1)$. We use a monotonic function for the alignment ground truth such that

$$f(t) = \begin{cases} 1 + 0.01 \cdot t^2 & t \leq 50 \\ 60 + 100 \cdot tanh(\frac{t}{100} + .5) & t > 50. \end{cases} \quad (27)$$

For every time-series the contrasting sequence is generated by nearest neighbour interpolation at time points given by (27). A sample of the sequence and its warped version are shown is Figure 3 where the signal in Figure 3(b) is generated from the signal shown in Figure 3(a) using the warping function (27). Sequences samples are then clustered and a codebook is generated and BOTW are constructed for each sequence. We have compare Dynamic Time Warping [12] (DTW), SPHMM [15] and Segmental Match (SM). We have examined four different maximum segment lengths of 10, 20, 50, 100 and the minimum length of 1 to show how our approximation of alignment improves as this parameter increases. The histogram distance metric is l_1 for all



(a) Distance from the ground truth (b) Running time per sequence

Figure 5. Quantitative comparison of DTW, SPHMM and SM in terms of closeness to the ground truth and running time in seconds

methods. Similar results are achieved using histogram intersection and χ^2.

Figure 4 illustrates a qualitative comparison of competing methods where all method attempt to recover the true warping path, i.e. the blue line. To be able to perform a *quantitative* comparison of the sequence of segments produces by SPHMM and SM which has pairs of correspondences that might be l_{max} away from each other with the ground truth which is defined for every pair of samples, we have used linear interpolation on the segment indexes. Then the l_1 distance between the all methods alignment paths and the ground truth is measured. The average l_1 distance over all 100 pairs of sequences and the running time for all methods is depicted in Figure 5. Obviously, DTW is not affected by changing maximum segment length and is basically unable to recover the true alignment. The reason relies on the rapid change of the warped sequence which was not captured by DTW. SPHMM on the other hand can successfully recover the alignment but its running time grows fast as the segment length is increased. Segmental match however, is able to recover the true alignment much better than DTW (even when maximum segment length is 10) and its running time is well below SPHMM.

3.2. Motion Capture

The unsupervised temporal commonality discovery proposed in [2] (TCD) extracts common sub-sequences of two contrasting time-series represented by BOTW without considering time monotonicity or other constraints. We show in this experiment that time monotonicity implemented by our method may be critical when such sub-sequence are used for classification purposes.

We used a subset of CMU Motion Capture [1] dataset to compare our method with [2]. We selected 62 sequences containing more than 40000 frames of 8 different actions: *walking, running, boxing, jumping, marching, dancing, sitting down and shaking hands*. Each class contains 7, 10, 8, 6, 10, 10, 7 and 4 sequences, respectively. Classes were selected with actions performed by different subjects. Sequence lengths range from 125 to 8000. We used a leave-one-out setting and nearest neighbour (NN) classifier.

Figure 6. Sample frames from UT-interation dataset #1.

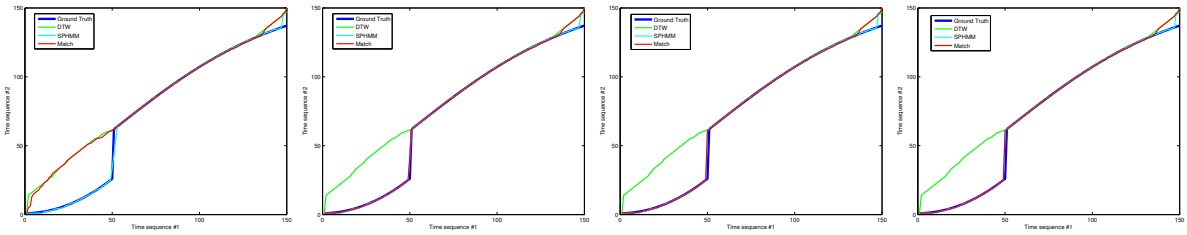
Each human motion was represented as the root position, orientation, and 29 relative joint angles. 3-D Euler angles were transformed to 3-D quaternion to provide a continuous representation and then BOTW was constructed for each sequence where the codebook size is 50.

For each pair of sequences we applied TCD to retrieve the most common sub-sequences. In each iteration after discovering common sub-sequences we removed them from the contrasting time-series and repeated this process five times or until one of the sequences is consumed. The sum of distances of all 5 common sub-sequences is then used as a similarity measure between each pair. For Fast-SM, the maximum segment length is set to 50 .

Using TCD we were able to achieve 40.32% classification accuracy while Fast-SM was able to classify the sequences with 66.13% accuracy. The running time of both methods were comparable even though Fast-SM did a full matching of every pair in the dataset. The low recognition rates might be the result of the chosen codebook size. Also histograms might not be the best representation for such multidimensional time-series. In fact, [15] reports much higher recognition rate on this dataset, that is 90.32%. If we also adopt the same joint angle representation and use SM instead of Fast-SM, which is still much faster than SPHMM, we attain 87.09% in accuracy. The result shows that our method is able to discover and match common segments and provide a better measure of similarity between pairs of sequences.

3.3. UT-Interaction

To apply segmental matching we needed to pick a dataset of reasonable length and complexity so we could try different segmentation lengths and observe how the recognition rate is affected. Therefore, popular action recognition datasets such as KTH [13] or Weizmann [5] datasets were not suitable for our settings because they contain short periodic actions and only a few frames are sufficient for a reliable recognition. Instead, we use the first subset of publicly available UT-interaction dataset containing 10 sequences (60 after segmentation of actions). Within each sequence, six actions, *hand shaking, hugging, kicking, pointing, punching* and *pushing* are performed by 10 different actors. The videos involve camera jitter. Pedestrians are present in the video which makes the recognition more difficult (Figure 6). We have used spatio-temporal interest points (Cuboids) [4] as the descriptors. Then k-means is



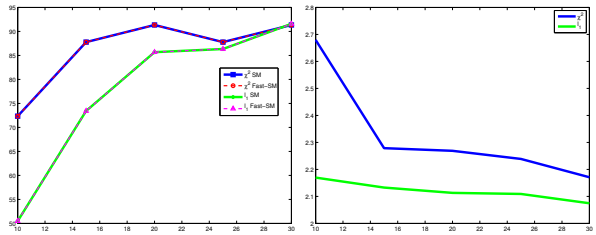
(a) $l_{max} = 10$ (b) $l_{max} = 20$ (c) $l_{max} = 50$ (d) $l_{max} = 100$

Figure 4. Qualitative comparison of DTW, SPHMM and SM. The thick blue line is the ground truth. SM approaches to the ground truth as soon as $l_{max} = 20$.

applied on the resulting features to produce an 800 element codebook.

We use a nearest neighbour classifier to compare with [9]. Leave-one-sequence-out cross-validation by holding one sequence for testing and using the remaining nine for training. Each action in the test set is matched with all training sequences. As a baseline we report the results on SVM using the same feature set and also the results reported in [9]. We have used l_1 and χ^2 histogram distances. The results on the l_1 distance metric are reported in Table 1. It is evident from the results that our approach significantly outperforms other methods. Using either l_1 or χ^2 distance metrics SM and Fast-SM were able to achieve the best result when the maximum segment length was 30. χ^2 achieved the best result even with maximum segment length of 20. We tried different maximum segment lengths, namely, 10,15,20, 25 and 30. Figure 7 illustrates how the resulting accuracy and speedup, gained by bounding the distance (Fast-SM), change as the maximum segment length increases applying l_1 and χ^2 histogram distance metrics. It is interesting to note that the recognition rates of Fast-SM and SM are identical in all cases eliciting the fact that the bounding technique and the smoothness assumption on the local likelihoods are in fact effective. In addition, Fast-SM achieves at least a 2-fold speedup compared to SM. As shown in 7(a), χ^2 achieves better results in smaller maximum segment lengths pointing to it as a more suitable measure of distance on segment histograms. Unfortunately, as the maximum segment length increases the bounds on the histogram distances become looser, resulting in reduced speedup. However, one should notice that the shortest sequence is 24 frames long and our final maximum segment length (30) already exceeds this limit. This implies that the model has the option to effectively considers a single BOTW representation as an alternative.

We also applied SPHMM to observe whether a complete alignment model is able to achieve better performance compared to SM and Fast-SM. The result showed that SPHMM cannot advance the recognition rate beyond 91.57% yet is at least 3 times slower than SM and 6 time slower than Fast-SM.



(a) Accuracy (b) Speedup

Figure 7. Accuracy and speedup results for l_1 and χ^2 distances. l_1 is depicted as green and χ^2 as blue. Accuracy result of Fast-SM for distance metric is identical to SM.

Table 1. Recognition rates on UT-interaction dataset #1

Method	Accuracy
Segmental Match	**91.57%**
Dynamic BOW [9]	85.0%
SVM	85.0%
Voting [19]	88.0%

4. Conclusion

In this paper we proposed a simplified segmental alignment model that was able to classify human activities accurately while remaining computationally efficient. We showed that an alignment model which consists of a single match operation when coupled with adaptive segmentation is able to approximate the true alignment of two warped sequences. We also used bounds on histogram distances to further accelerate our algorithm without compromising the classification performance.

References

[1] http://mocap.cs.cmu.edu/. 6

[2] W.-S. Chu, F. Zhou, and F. D. la Torre. Unsupervised temporal commonality discovery. *European Conference on Computer Vision (ECCV)*, pages 373–387, 2012. 2, 3, 6

[3] H. Ding, G. Trajcevski, P. Scheuermann, X. Wang, and E. Keogh. Querying and mining of time series data: experimental comparison of representations and distance measures. *Proceedings of the VLDB Endowment*, 1(2):1542–1552, 2008. 1

[4] P. Dollar, V. Rabaud, G. Cottrell, and S. Belongie. Behavior recognition via sparse spatio-temporal features. In *Proceedings of the 14th International Conference on Computer Communications and Networks*, ICCCN '05, pages 65–72, Washington, DC, USA, 2005. IEEE Computer Society. 1, 6

[5] L. Gorelick, M. Blank, E. Shechtman, M. Irani, and R. Basri. Actions as space-time shapes. *Transactions on Pattern Analysis and Machine Intelligence*, 29(12):2247–2253, December 2007. 6

[6] C. H. Lampert, M. B. Blaschko, and T. Hofmann. Efficient subwindow search: A branch and bound framework for object localization. *IEEE Trans. Pattern Anal. Mach. Intell.*, 31(12):2129–2142, Dec. 2009. 3

[7] J. C. Niebles, H. Wang, and L. Fei-Fei. Unsupervised learning of human action categories using spatial-temporal words. *Int. J. Comput. Vision*, 79(3):299–318, 2008. 1

[8] H. Riemenschneider, M. Donoser, and H. Bischof. Bag of optical flow volumes for image sequence recognition. In *BMVC*, 2009. 2

[9] M. S. Ryoo. Human activity prediction: Early recognition of ongoing activities from streaming videos. *IEEE Conference on Compupter vision*, 2011. 1, 2, 7

[10] M. S. Ryoo and J. K. Aggarwal. Spatio-temporal relationship match: Video structure comparison for recognition of complex human activities. In *ICCV*, pages 1593–1600, 2009. 1

[11] M. S. Ryoo and J. K. Aggarwal. UT-Interaction Dataset, ICPR contest on Semantic Description of Human Activities (SDHA). http://cvrc.ece.utexas.edu/SDHA2010/Human_Interaction.html, 2010. 5

[12] H. Sakoe and C. Chiba. Dynamic Programming Algorithm Optimization for Spoken Word Recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 26(1):43–49, 1978. 3, 5

[13] C. Schuldt, I. Laptev, and B. Caputo. Recognizing human actions: A local svm approach. In *Proceedings of the Pattern Recognition, 17th International Conference on (ICPR'04) Volume 3 - Volume 03*, ICPR '04, pages 32–36, Washington, DC, USA, 2004. IEEE Computer Society. 1, 6

[14] S. Shariat and V. Pavlovic. Isotonic CCA for Sequence Alignment and Activity Recognition. *IEEE Conference on Compupter vision*, 2011. 1

[15] S. Shariat and V. Pavlovic. Improved sequence classification using adaptive segmental sequence alignment. *Journal of Machine Learning Research - Proceedings Track*, 25:379–394, 2012. 1, 2, 5, 6

[16] J. Sivic and A. Zisserman. Video google: A text retrieval approach to object matching in videos. In *Proceedings of the Ninth IEEE International Conference on Computer Vision - Volume 2*, ICCV '03, pages 1470–, Washington, DC, USA, 2003. IEEE Computer Society. 1, 2

[17] P. Turaga, R. Chellappa, V. S. Subrahmanian, and O. Udrea. Machine recognition of human activities: A survey. *Circuits and Systems for Video Technology, IEEE Transactions on*, 18(11):1473–1488, 2008. 1

[18] P. Viola and M. J. Jones. Robust real-time face detection. *Int. J. Comput. Vision*, 57(2):137–154, May 2004. 5

[19] D. Waltisberg, A. Yao, J. Gall, and L. Van Gool. Variations of a hough-voting action recognition system. In *Proceedings of the 20th International conference on Recognizing patterns in signals, speech, images, and videos*, ICPR'10, pages 306–312, Berlin, Heidelberg, 2010. Springer-Verlag. 7

[20] T.-H. Yu, T.-K. Kim, and R. Cipolla. Real-time action recognition by spatiotemporal semantic and structural forest. In *Proceedings of the British Machine Vision Conference*, pages 52.1–52.12. BMVA Press, 2010. doi:10.5244/C.24.52. 1

[21] F. Zhou and F. de la Torre. Canonical time warping for alignment of human behavior. *Advances in Neural Information Processing Systems (NIPS)*, pages 1–9, 2009. 1

[22] F. Zhou and F. De la Torre. Generalized time warping for multi-modal alignment of human motion. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012. 1