Discriminative Human Action Segmentation and Recognition using Semi-Markov Model

Qinfeng Shi
NICTA and Australian National University
Canberra, Australia
qinfeng.shi@rsise.anu.edu.au

Li Wang
Southeast University
Nanjing, China
wang.li.seu.nj@gmail.com

Li Cheng, Alex Smola
NICTA and Australian National University
Canberra, Australia
li.cheng, alex.smola@nicta.com.au

Abstract

Given an input video sequence of one person conducting a sequence of continuous actions, we consider the problem of jointly segmenting and recognizing actions. We propose a discriminative approach to this problem under a semi-Markov model framework, where we are able to define a set of features over input-output space that captures the characteristics on boundary frames, action segments and neighboring action segments, respectively. In addition, we show that this method can also be used to recognize the person who performs in this video sequence. A Viterbi-like algorithm is devised to help efficiently solve the induced optimization problem. Experiments on a variety of datasets demonstrate the effectiveness of the proposed method.

1. Introduction

Our goal in this paper is to segment and recognize elementary actions such as run, walk and draw on board, from a video sequence where one person performs a sequence of such actions. This is a fundamental problem in human action understanding and has a wide range of applications in *e.g.* surveillance, video retrieval and intelligent interface. It is nevertheless challenging due to the high variability of appearances, shapes and possible occlusions, and things are further complicated for continuous action recognition in our case where it is necessary to segment the input video sequence into continuous action segments.

The Models We consider a discriminative learning approach. To better motivate our proposed model, we will describe in turn three types of statistical models that can be used in human action analysis, as illustrated in Figure 2 (from top to bottom panels).

The first type of models (Figure 2 top row, *e.g.* KNN, support vector machine (SVM)) simply ignores the temporal dependencies among video frames thus each frame is assumed to be statistically independent from the rest. This however limits its prediction ability, particularly when there exist ambiguities in some video sequences (*e.g.* Figure 1 top), where it is difficult to identify an appropriate action label for one frame until we have knowledge of its temporal context. This is partially solved in [9, 13, 19] where the feature descriptors incorporate the spatial-temporal characteristics of each type of action subsequences, and a variant of the first model is applied in this feature space to decide to which category a new action segment belongs. This, however, requires pre-segmentation of into action segments.

The second model is a Markov chain model (Figure 2 middle, *e.g.* HMM, conditional random field (CRF) and SVM-HMM [2, 8, 15]) that considers statistical dependencies over all adjacent frames and shows good performance on pre-segmented datasets. We argue that it is not well suited to the video sequences we consider in this paper. First, continuous action recognition is inherently a segmentation problem, where each action starts, lasts for a period of frames and then transits to another action. Second, although the Markov chain model considers local interaction between adjacent frames, it does not have access to characteristics over the entire action segment, such as the length of the segment.

Figure 1: Walk or draw on the board? **Left** illustrates one frame extracted from an exemplar video sequence. It generates ambiguity between two possible actions: walk or draw on the board, which will only become clear by observing the context of the sequence, shown in **Right**.

We instead propose to use a semi-Markov model (SVM-SMM, Figure 2 bottom), which exploits the segmentation nature of the problem, where the modeling emphasis is on the properties within individual segments and between adjacent action segments of variable length. In particular, our SVM-SMM approach makes use of three types of features to: (a) relate to the boundary frames of each segment, (b) encode content characteristics about segments, and (c) capture interactions between neighboring segments. More details about our feature representation are provided in Section 3.

The main contributions of this paper are two-fold:

- A large-margin discriminative approach is proposed to address the problem of action segmentation and recognition under a semi-Markov model framework, where a Viterbi-like algorithm is devised for efficient inference. In addition, we show that this method can also be used to recognize the person who performs in a video sequence.
- Based on a codebook object representation that incorporates SIFT [7] and shape context [1] features, a set of feature functions are defined over input-output space that encode characteristics of boundary frames, segments as well as neighboring segments.

Related Work Generative statistical approaches, especially Markov models [2, 3, 4, 5, 8, 20] have come in wide use for modeling and analyzing human actions, *e.g.* HMMs and its variants such as coupled HMMs [2, 20]. Besides, [8] uses HMMs and AdaBoost to segment and recognize motion capture sequences. [3] applies a two-layered extension of SMM to model high-level daily activities, while [4] tackles the problems of 2D tracking and 3D motion capture using a variable-length Markov model.

Recently, large margin based discriminative learning schemes[18] are extended to cases where there are

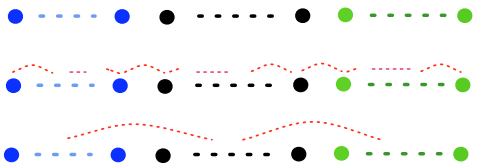


Figure 2: We compare three types of statistical models on a sequence that contains a series of action segments (in different colors). The dependency in each model is illustrated as red arcs. **Top**: an *iid* model where each frame label is independent from others. **Middle**: a Markov chain model where each frame label depends on its adjacent frame labels. **Bottom**: the proposed semi-Markov model where frames in one segment share one label, and this label depends on its adjacent segment labels.

structured dependencies among the outputs [10, 16, 17] (*e.g.* SVM-HMM where the output could be time series sequences), and encouraging results are obtained in bio-informatics and natural language processing related applications. As far as we are aware, there is not much work along this line conducted in the field of video action analysis. In particular, a SMM approach is proposed by [10] for gene structure prediction applications. They propose a two-stage learning algorithm where binary SVM classifiers are firstly used to identify segment boundaries and the *content* of each segment is recognized separately in the second stage. Clearly this procedure is quite different from the efficient Viterbi-like algorithm we propose in section 2 where action segmentation and recognition can be addressed jointly. We note in the passing that conditional random field (CRF) is another discriminative model that deals with structured outputs, which has been applied to action recognition [15, 19], and recently Semi-Markov CRF [12] has been proposed for natural language processing problems.

Paper Outline In section 2 we give a probabilistic account of the proposed discriminative framework. To solve the induced optimization problem, we introduce and analyze an efficient inference algorithm. We proceed to provide details of the feature functions in section 3. A number of experimental and comparative results are presented and discussed in section 4, followed by a summary in section 5.

2. The Approach

Define the set of action labels as $\mathcal{C} = \{1, \cdots, C\}$, and the set of persons $\mathcal{I} = \{1, \cdots, I\}$. Without loss of generality, we assume that there is exactly one person $P \in \mathcal{I}$ in a given video sequence. In this paper, the human action analysis problem is formulated as an optimization problem over a probabilistic graphical

model.

Graphical model definition: *Consider a graph defined on the action sequence Y for person $P \in \mathcal{I}$. In particular we consider a semi-Markov model, where one node in this graph corresponds to a segment of video frames having the same action label, and one edge captures the statistical dependency between two adjacent segments. Given a video sequence of length m as $X = \{x_k\}_{k=0}^{m-1}$, we attach a dummy node x_m to this sequence, denote l the number of segments, and define a set of segment boundaries $\{n_k\}_{k=0}^{l-1}$ with $n_{k-1} < n_k < n_{k+1}, \forall k$, and fix $n_0 = 0$, $n_l = m$ to satisfy boundary conditions. As a consequence, the first segment is $[0, n_1)$, and the last segment is $[n_{l-1}, m)$. The action label sequence can be equivalently represented as $Y = \{(n_k, c_k)\}_{k=0}^{l-1}$, where each pair (n_k, c_k) denotes the starting position and the corresponding action label for the kth segment $[n_k, n_{k+1})$.*

During training we have access to a set of T video sequences $\mathcal{X} = \{X_t\}_{t=1}^{T}$, as well as corresponding labels $\mathcal{Y} = \{\cdots, Y_t, \cdots\}$, accordingly. We further assume that the *conditional* distribution over the label Y given the observed sequence $X = X_t$ can be written as an exponential family model:

$$\log p(Y|X, W) = \langle W, \Phi(X, Y)\rangle - A_W(X). \tag{1}$$

By the independence assumption among action sequences, the joint *conditional* probability over all training sequences can be factorized as $p(\mathcal{Y}|\mathcal{X}, W) = \prod_t p(Y_t|X_t, W)$. Here $A_W(X)$ is the normalization constant to ensure $p(Y|X, W)$ respects a valid probability distribution, and W is the parameter vector. $\Phi(X, Y)$ is a feature map over the joint input-output space, which can be decomposed with respect to the SMM graph structure (illustrated in Figure 2 bottom) as

$$\Phi(X, Y) = \Big(\sum_{i=0}^{l-1} \phi_1(X, n_i, c_i), \sum_{i=0}^{l-1} \phi_2(X, n_i, n_{i+1}, c_i), \sum_{i=0}^{l-1} \phi_3(X, n_i, n_{i+1}, c_i, c_{i+1}) \Big), \tag{2}$$

As mentioned in the beginning of this paper, ϕ_1 and ϕ_2 capture the observation-label dependency in current action segment, where ϕ_1 concentrates on a segment's boundary frame and ϕ_2 takes care of global characteristics of the segment. The interaction between two neighboring segments is encoded in ϕ_3. W can as well be decomposed in this manner.

In what follows, we present a probabilistic account of applying the large-margin discriminative framework to structured data [16, 17]. In prediction phase, for an unseen video sequence X, its action sequence is obtained by maximum likelihood decoding of the *conditional* probability

$$Y^* = \arg\max_Y \log p(Y|X, W) = \arg\max_Y F(X, Y). \tag{3}$$

where $F(X, Y) \triangleq \langle W, \Phi(X, Y)\rangle$ is the discriminant function. Therefore the optimal assignment of Y^* amounts to choosing the maximum of the discriminant function.

Learning in our SVM-SMM framework is accomplished by solving a regularized optimization problem with respect to the parameter W: we would like W be bounded to avoid over-fitting, and to maximize the minimum log ratio of the *conditional* probabilities, as

$$\min_W \frac{\|W\|^2}{2} \quad s.t. \quad \log \frac{p(Y_t|X_t, W)}{p(Y|X_t, W)} \geq \Delta(Y_t, Y) \ \ \forall t, Y \tag{4}$$

for the set of video sequences $\{t : t \in 1, \cdots, T\}$, where the label loss $\Delta(Y_t, Y)$ is the margin between the two feasible label assignments.

We invoke (1), and add the slack variable ξ to account for the non-separable case. As the normalization terms cancel out, the optimization problem states, for $\eta > 0$,

$$\begin{aligned} \min_{W,\xi} \quad & \frac{\|W\|^2}{2} + \frac{\eta}{T} \sum_t \xi_t \\ s.t. \quad & \langle W, \triangle\Phi(X_t, Y)\rangle \geq \Delta(Y_t, Y) - \xi_t \quad \forall t, Y, \end{aligned} \tag{5}$$

where $\triangle\Phi(X_t, Y) = \Phi(X_t, Y_t) - \Phi(X_t, Y)$. Its dual program is

$$\begin{aligned} \max_\alpha \quad & \sum_{t,Y} \alpha_{t,Y} \Delta(Y_t, Y) - \frac{\eta}{2} \Big\| \sum_{t,Y} \alpha_{t,Y} \triangle\Phi(X_t, Y) \Big\|^2 \\ s.t. \quad & \alpha_{t,Y} \in \mathcal{M} \quad \forall t, \end{aligned} \tag{6}$$

where $\mathcal{M}$ denotes the probability simplex constraint. Applying the Representer theorem [6] yields a dual representation of the discriminant function as

$$F(X, Y) = \sum_{t,Y'} \alpha_{tY'} \left\langle \triangle\Phi(X_t, Y'), \Phi(X, Y) \right\rangle. \tag{7}$$

F can also be decomposed into three components: $f_i(X, Y) = \langle w_i, \phi_i(X, Y)\rangle, \forall i = \{1, 2, 3\}$ as

$$\sum_{i=0}^{l-1} \Big(f_1(X, n_i, c_i) + f_2(X, n_i, n_{i+1}, c_i) + f_3(X, n_i, n_{i+1}, c_i, c_{i+1}) \Big). \tag{8}$$

Notice the proposed model is very general and contains several existing models as special cases. Let $M \geq 1$ upper-bound the maximum number of frames a segment would last. By fixing $M = 1$ and using only features ϕ_1 and ϕ_2 (*i.e.* setting $\phi_3 = 0$), we recover the multi-class SVM (Figure 2 top). When fixing $M = 1$ and utilizing all three features, we obtain the SVM-HMM [17] (Figure 2 middle).

2.1. The Algorithm

Both the primal (5) and the dual problem (6) are in fact intractable, as the configuration space of $\mathcal{Y}$ is in the order of $T \times C^m$, thus the number of constraints grows exponentially as the length of training sequences increases. However, this problem can be approximately solved by using an optimization technique known as column generation [17]. Here, rather than solving (6) immediately, one finds the most violated constraint (column generation) using current solution of (6), and iteratively adds these constraints to the optimization problem. This iterative procedure is guaranteed to converge to the the optimal solution [17], and it approximates the optimal solution to arbitrary precision in polynomial number of iterations. Now, for column generation, we need to solve

$$Y^* = \underset{Y \in \mathcal{Y}}{\operatorname{argmax}} \ \Delta(Y_t, Y) + F(X_t, Y), \tag{9}$$

which gives the *most violated* constraint as long as $Y^* \neq Y_t$. Here we devise a Viterbi-like dynamic programming scheme as presented in Algorithm 1. Besides, we use the Hamming distance to measure the label loss $\Delta(Y, Y')$ between alternative action sequence labels as

$$\sum_{k=0}^{m-1} (1 - \delta(y_k = y'_k)),$$

where $\delta(x)$ is the indicator function.

Take any segment i, we denote its related boundaries as $n_- \triangleq n_{i-1}$ and $n \triangleq n_i$. Similarly the related labels are $c_- \triangleq c_{i-1}$ and $c \triangleq c_i$. Now, we maintain a partial score $S(X, n, c)$ that sums up to segment i (*i.e.* starts at position 0 and ends with the segment $[n_-, n)$ with labels c_- (for n_-) and c (for n), respectively), and it is defined as

$$\max_{c_-, \max\{0, n-M\} \leq n_- < n} \{ \ S(X, n_-, c_-) + g(X, n_-, n, c_-, c) \ \}, \tag{10}$$

where the increment $g(X, n_-, n, c_-, c)$ equals to

$$f_1(X, n_-, c_-) + f_2(X, n_-, n, c_-) + f_3(X, n_-, n, c_-, c) + 1 - \sum_{k=n_-}^{n-1} \delta(y_k = c_-).$$

It is easy to verify that in the end, the sum of two terms in the RHS of (9) amounts to $S(m, c_m)$. After slight modification, this algorithm is also used to solve the ML problem of (3) in prediction phase.

This column generation algorithm is very efficient: its time complexity is $O(mMC^2)$ thus is linear with respect to the sequence length m, and its memory complexity is $O(m(C+2))$. In our experiments, it is implemented in C++ and performs on average 0.05 seconds per frame at running time using an PC with an Intel Pentium 4 3.0GHz processor and with 512M memory. This enables our method to efficiently work with video data.

Algorithm 1 Column Generation

Input: sequence X_t with length m, its true label Y_t, and maximum length of a segment M
Output: score s, optimal label Y^*
Initialize matrices $S \in \mathbb{R}^m \times C$, $J \in \mathbb{Z}^m$, and $L \in \mathbb{Z}^m$ to 0, $Y^* = \emptyset$
for $i = 1$ **to** m **do**
  for $c_i = 1$ **to** C **do**
    $(J_i, L_i) = \underset{j, c_j}{\operatorname{argmax}} S(j, c_j) + g(j, i, c_j, c_i)$
    $S(i, c_i) = S(j^*, c^*_{j^*}) + g(j^*, i, c^*_{j^*}, c_i)$
  end for
end for
$c^*_m = \underset{c_m}{\operatorname{argmax}} S(m, c_m)$
$s = S(m, c^*_m)$
$Y^* \leftarrow \{(m, c^*_m)\}$
$i \leftarrow m$
repeat
  $Y^* \leftarrow \{(J_i, L_i), Y^*\}$
  $i \leftarrow J_i$
until $i = 0$

2.2. Extension and Variants

In addition to action recognition, our method can also be used to recognize which person appears in the given sequence. This is achieved by extending the labels to $\mathcal{Y} = \{\cdots, (Y_t, P_t), \cdots\}$, where for the label pair (Y, P) of a sequence, Y still denotes the sequence of frame-by-frame action labels, and P is the person label of the entire sequence. Similarly, the joint feature map can be extended to

$$\Phi(X, Y, P) = \Big(\sum_{i=0}^{l-1} \phi_1(X, n_i, c_i, P), \sum_{i=0}^{l-1} \phi_2(X, n_i, n_{i+1}, c_i, P), \sum_{i=0}^{l-1} \phi_3(X, n_i, n_{i+1}, c_i, c_{i+1}, P) \Big), \tag{11}$$

and the discriminant function to $F(X, Y, P)$ accordingly. For column generation, we now need to solve

$$(Y^*, P^*) = \underset{(Y,P) \in \mathcal{Y}}{\operatorname{argmax}} \ \Delta((Y_t, P_t), (Y, P)) + F(X_t, Y, P).$$

The label loss $\Delta((Y, P), (Y', P'))$ between alternative sequence labels becomes

$$\sum_{k=0}^{m-1} (1 - \delta(y_k = y'_k)) + \lambda(1 - \delta(P = P')),$$

where $\lambda \geq 0$ is a trade-off constant. This leads to an extended partial score $S(X, n, c, P)$

$$\max_{c_-, \max\{0, n-M\} \leq n_- < n} \{\ S(X, n_-, c_-, P) + g(X, n_-, n, c_-, c, P)\ \}, \tag{12}$$

where the increment $g(X, n_-, n, c_-, c, P)$ equals to

$$f_1(X, n_-, c_-, P) + f_2(X, n_-, n, c_-, P) + f_3(X, n_-, n, c_-, c, P) + 1 - \sum_{k=n_-}^{n-1} \delta(y_k = c_-).$$

Obviously the task of action segmentation and recognition is a special case of this extended framework. Moreover, in standard datasets such as Mobo [11] each sequence is commonly pre-processed to contain multiple cycles of *one* action, rather than continuous action subsequences. Our method is used in this scenario to segment a sequence frame-by-frame into atomic action cycles as well as to recognize the action performed over the entire sequence. These allow us to carry out a variety of experiments reported in section 4 where we are able to choose to either segment and recognize actions, or additionally recognize persons.

3. Feature Representation

In this section, we discuss in details the implementation of the feature map Φ (2) in our context.

The foreground object in each image is obtained using background subtraction. By running the SIFT [7] key points detector, the object is represented as a set of key feature points extracted from the foreground and each point has 128-dim SIFT features, which are known to be relatively invariant to illumination and view-angle changes, and importantly, are insensitive to the objects' color appearance by capturing local image textures in the gradient domain. In addition, 60-dim shape context [1] features are constructed for each feature point, which roughly encode how each point "sees" the rest points. The two sets of features are then concatenated with proper scaling to form a 188-dim vector. This point-set object representation are further transformed into a 50-dim codebook using K-means, similar to the visual vocabulary approach of [14].

Now, as a new frame is presented, each of its points is projected into this codebook space with a cluster assignment, and the object is therefore represented as a 100-dim histogram vector h. Typical results of this codebook representation is illustrated in Figure 4 bottom, where we randomly choose four codebook clusters and plot the assigned feature point locations on individual images. This convincingly shows that each cluster can pick up reasonably similar patches over time and across people.

Equipped with this codebook representation, we construct feature functions ϕ_1, ϕ_2 and ϕ_3 as follows. Note that for Mobo dataset [11] we use a different set of features to better represent one action segment now containing one atomic action cycle.

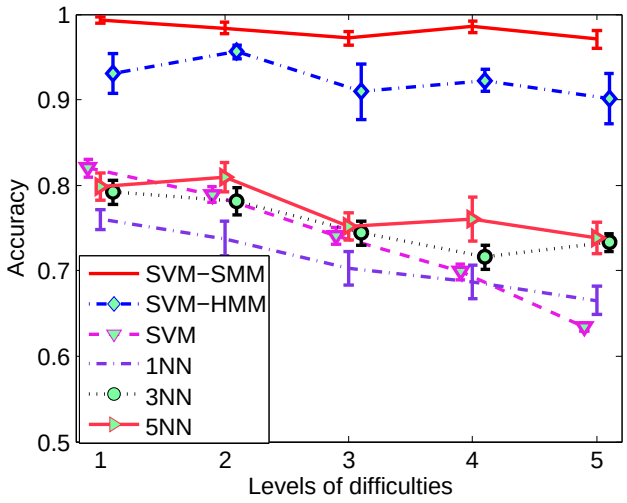
Boundary Frame Features $\phi_1(X, n_i, c_i) = \psi_1(X, n_i) \otimes c_i$, where $\otimes$ denotes a tensor product. ψ_1 is a concatenation of two features. The first is a constant 1 which acts as the bias term. The second part is obtained from a sliding window of size w_s centered on the boundary frame. When $w_s = 1$ it becomes the single histogram vector h_{n_i}. For Mobo dataset [11], by using $w_s = 3$ our second part is thus a concatenation of the three consecutive histogram vectors.

Node Features on Segment Node features are devised to capture the characteristics of the segment. ϕ_2 is defined as $\phi_2(X, n_i, n_{i+1}, c_i) = \psi_2(X, n_i, n_{i+1}) \otimes c_i$. $\psi_2(X, n_i, n_{i+1})$ contains three components: the length of this segment, the empirical mean and variance of the histogram vector of the segment (*i.e.* over frames from n_i to $n_{i+1} - 1$).

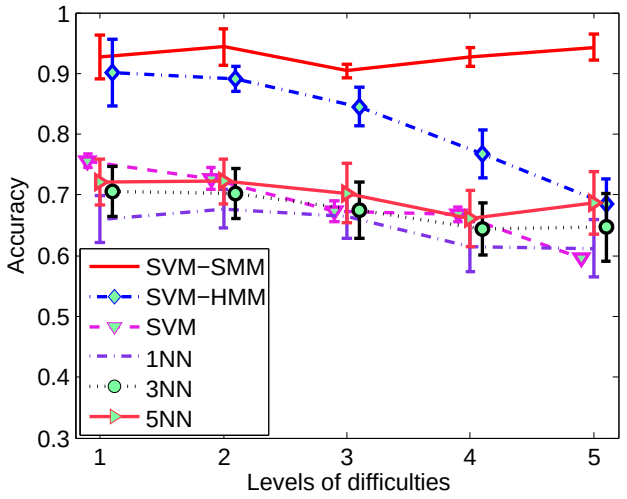
For Mobo dataset [11], we use instead the following features: the length of this segment, h_{n_i} of the first frame, the element-wise difference between the first and the middle frames $h_{n_i} - h_{\lfloor \frac{n_i + n_{i+1}}{2} \rfloor}$, as well as the difference between the first and the last frames $h_{n_i} - h_{n_{i+1}-1}$. Here the last two features aim to encode informative properties within one atomic action cycle, for example, often the object of the first frame poses similarly to that of the last frame, but very differently from that of the middle frame.

Edge Features on Neighboring Segments As in practice we do have prior knowledge about how long a segment would at least last, we define the minimum duration of a segment as d. Similarly $\phi_3(X, n_i, n_{i+1}, c_i, c_{i+1}) = \psi_3(X, n_i, n_{i+1}) \otimes c_i \otimes c_{i+1}$, and it is a concatenation of the following components: (1) the empirical mean of the histogram vector from frames n_i to $n_{i+1} - 1$, and (2) from frames n_{i+1} to $n_{i+1} + d$, as well as (3) the empirical variance of the histogram vector from n_i to $n_{i+1} - 1$, and (4) from n_{i+1} to $n_{i+1} + d$.

A different set of ϕ_3 features are constructed catering for Mobo dataset [11] where we aim to capture the interaction between two neighboring action cycles, as: $h_{n_{i+1}} - h_{n_i}$, $h_{n_{i+1}+d} - h_{n_i}$, $h_{n_{i+1}+d} - h_{n_i+d}$, and $h_{n_{i+1}+d} - h_{\lfloor \frac{n_i + n_{i+1}}{2} \rfloor}$. The intuition behind the features here is in line with what we have in ϕ_2.



(a) Recognize actions



(b) Recognize actions and persons

Figure 3: Comparing six methods on two scenarios using a synthetic dataset. See text for details.

4. Experiments

Experiments are conducted on three datasets, where the proposed method (SVM-SMM) is compared with five other algorithms: KNN (where K=1, 3, 5), SVM multiclass and SVM-HMM. To segment and recognize actions on a test sequence[1], a frame-by-frame classification strategy is adopted for all the comparison algorithms. Furthermore, we conduct several experiments to show that this method can also be used to recognize the person who performs in the sequence, where we enforce a rather strict evaluation scheme: a prediction on one frame is regarded correct only when both the right person and the right action are identified. For a fair comparison, each method is tuned separately to obtain best performance.

[1]which we also call in short as action recognition when no confusion occurs.

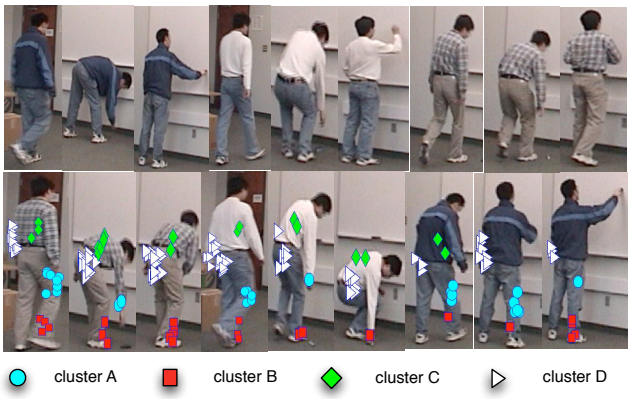


Figure 4: A Walk-Bend-Draw (WBD) dataset. **Top** shows some sample frames of the dataset (see also Figure 1 bottom). **Bottom** displays the assignments of image feature points on four randomly chosen codebook clusters over time and across person.

Synthetic dataset We consider a controlled environment where we are able to quantitatively measure the performance of comparison algorithms by varying the difficulty level of problems from easy to difficult. We do this by constructing a two-person two-action synthetic dataset which consists of five trials (each trial has a set of ten sequences and corresponds to a certain level of difficulty. This dataset is available at sml.nicta.com.au/~licheng/aSR/).

Here each person P equals to one semi-Markov model (SMM) containing its own Gaussian emission probabilities $\mathcal{N}(\mu_{c,P}, \sigma_{c,P})$ and duration parameters $\lambda_{c,P}$ for the two actions $c = 1, 2$, respectively. Each sequence of length 150 is generated by sampling from a SMM model, and as a result contains continuous action subsequences.

Now, we build five trials as follows. For each trial, five sequences are generated from each person's model, and in the end we have ten sequences. While across trials, we vary the levels of difficulties by moving μ_2 toward μ_1 and fixing other parameters of the models.

On this dataset, we conduct two controlled experiments where 5-fold cross-validation are used. The results are displayed in Figure 3, where the first experiment considers solely action recognition, and the second is to recognize both actions and persons.

SVM-SMM consistently performs the best, seconded by SVM-HMM while the rest methods have inferior performance. An interesting observation is made from the second experiment: here SVM-SMM retains a stable classification rate over different levels of difficulties, whereas other methods, especially SVM-HMM, deteriorate. This mainly dues to that SVM-SMM is capable

	1NN	3NN	5NN	SVM	SVM-HMM	SVM-SMM
Action Recognition	0.82 ± 0.02	0.80 ± 0.03	0.77 ± 0.03	0.84 ± 0.03	0.87 ± 0.02	0.91± 0.02
Action and Person	0.60 ± 0.10	0.54 ± 0.11	0.50 ± 0.12	0.69 ± 0.07	0.85 ± 0.03	0.86 ± 0.03

Table 1: A summary of the first two experiments conducted on the WBD dataset: the first row shows performance on action recognition, and the second row displays results of jointly recognize actions and persons.

truth vs. predict	walk	bend	draw
walk	0.78	0.22	0.00
bend	0.07	0.91	0.02
draw	0.03	0.11	0.86

Table 2: Confusion matrix of applying SVM-SMM on the WBD dataset to jointly recognize actions and persons.

of encoding segment-level characteristics such as the length and empirical variance of the segment.

Walk-Bend-Draw dataset In addition to the standard datasets (*e.g.* [11]) where videos have been pre-segmented to allow one action per sequence, to evaluate the empirical performance of the proposed approach on sequences that each contains continuous actions, we construct a Walk-Bend-Draw (WBD) dataset (some sample frames are displayed in Figure 1 and 4, this dataset is available at at `sml.nicta.com.au/~licheng/aSR/`). This indoor video dataset contains three subjects, each performs six action sequences at 30 fps with resolution 720x480, and each sequence consists of three continuous actions: slow *walk*, *bend* body and *draw* on board, and on average each action lasts about 2.5 seconds. We subsample each sequence to obtain 30 key frames, and manually label the ground truth actions.

To measure the performance, 6-fold cross-validation is used for each comparison method. Table 1 reports on two experimental scenarios: action recognition, and simultaneous recognition of actions and persons, where SVM-SMM still consistently delivers the best results. In specific, Table 2 displays the confusion matrix of SVM-SMM in the second experiment, where the two actions – *walk* and *draw* – seem to be rarely confused with each other, nevertheless both sometimes are mis-predicted as bend. This is to be expected, as although *walk* and *draw* appear to be more similar to human observer in isolated images (see Figure 1), it nevertheless can be learned by SVM-SMM from a training set of videos that *walk*, *bend* and *draw* are usually conducted in order.

CMU Mobo dataset [11] This dataset contains 24 individuals[2] walking on a treadmill. The subjects perform four different actions: *slow* walk, *fast* walk, *incline* walk and slow walk with a *ball*. Each sequence has been pre-processed to contain several cycles of a *single* action and we additionally manually label the boundary positions of these cycles. The task on this dataset is to automatically partition a sequence into atomic action cycles, as well as predict the action label of this sequence. We evaluate the action recognition and segmentation performance separately. To measure segmentation performance, we adopted the F_1-score, which is often used in information retrieval tasks, and is given by $(2 \times \text{Precision} \times \text{Recall})/(\text{Precision} + \text{Recall})$.

[2]The dataset is originally consisted of 25 subjects. We drop the last person since we have problems obtaining the sequences of this individual walking with balls.

	1NN	SVM	SVM-HMM	SVM-SMM
act.	0.65 ± 0.02	0.67 ± 0.03	0.75 ± 0.06	0.75 ± 0.03
seg.	0.16 ± 0.05	0.15 ± 0.03	0.43 ± 0.01	0.59 ± 0.03

Table 3: Comparison on CMU Mobo dataset. The first row presents action recognition rate on a sequence, while the second row gives F_1-score for segmentation measurement. See text for details.

Table 3 presents the results after 6-fold cross-validation. To save space the results of 3NN and 5NN are not displayed as they are very similar to 1NN. Here both methods beat the baseline methods including KNN (K=1,3,5) and SVM by a large margin. Besides, SVM-SMM shows competitive performance where it outperforms SVM-HMM on action label prediction as well as on segmentation of action cycles.

5. Outlook and Future Work

We present a novel semi-Markov discriminative approach to human action analysis, where we intent to segment and recognize continuous action sequences. In addition, we show that this method can also be used to recognize the person who performs in this video sequence. By employing a Viterbi-like column generation algorithm, this approach allows us to explicitly encode segment-level properties into feature representation and still be solved efficiently. Experimental results on a variety of dataset demonstrate that our approach is flexible to cater for different scenarios, yet it is competitive comparing to the state-of-the-art methods.

Our approach can be extended in several directions. It is promising to explore the dual representation in order to incorporate matching cost between point sets. We also plan to apply this approach to closely related problems, *e.g.* to detect unusual actions from a long-

period video dataset.

Acknowledgements

We thank Liang Wang for helping us improve the presentation of this paper and Baochun Bai and Cheng Lei for creating the WBD dataset. NICTA is funded by the Australian Government as represented by the Department of Broadband, Communications and the Digital Economy and the Australian Research Council through the ICT Centre of Excellence program. This work is supported by the IST Program of the European Community, under the Pascal Network of Excellence, IST-2002-506778.

References

[1] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Trans. Pattern Analysis and Machine Intell.*, 24(4):509–522, 2002. 2, 5

[2] M. Brand, N. Oliver, and A. Pentland. Coupled hidden markov models for complex action recognition. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, page 994, Washington, DC, USA, 1997. IEEE Computer Society. 1, 2

[3] T. V. Duong, H. H. Bui, D. Q. Phung, and S. Venkatesh. Activity recognition and abnormality detection with the switching hidden semi-markov model. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pages 838–845, Washington, DC, USA, 2005. IEEE Computer Society. 2

[4] A. Galata, N. Johnson, and D. Hogg. Learning variable length markov models of behaviour. *Computer Vision and Image Understanding: CVIU*, 81(3):398–413, 2001. 2

[5] A. Kale, A. Sundaresan, A. Rajagopalan, N. Cuntoor, A. RoyChowdhury, V. Kruger, and R. Chellappa. Identification of humans using gait. *IEEE Trans. on Image Processing*, pages 1163–1173, Sept. 2004. 2

[6] G. S. Kimeldorf and G. Wahba. Some results on Tchebycheffian spline functions. *J. Math. Anal. Appl.*, 33:82–95, 1971. 3

[7] D. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004. 2, 5

[8] F. Lv and R. Nevatia. Recognition and segmentation of 3-d human action using hmm and multi-class adaboost. In *European Conference on Computer Vision*, pages IV: 359–372, 2006. 1, 2

[9] J. Niebles and L. F. Fei. A hierarchical model of shape and appearance for human action classification. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pages 1–8, 2007. 1

[10] G. Ratsch and S. Sonnenburg. Large scale hidden semi-markov svms. In B. Schölkopf, J. Platt, and T. Hoffman, editors, *NIPS*, pages 1161–1168. MIT Press, 2006. 2

[11] R.Gross and J. Shi. The CMU motion of body (MoBo) database. Technical Report Tech. Report CMU-RI-TR-01-18, Robotics Institute, Carnegie Mellon University, 2001. 5, 7

[12] S. Sarawagi and W. W. Cohen. Semi-markov conditional random fields for information extraction. In *NIPS*, 2004. 2

[13] C. Schuldt, I. Laptev, and B. Caputo. Recognizing human actions: A local svm approach. In *Proc. Intl. Conf. Pattern Recognition*, pages 32–36, Washington, DC, USA, 2004. IEEE Computer Society. 1

[14] J. Sivic and A. Zisserman. Video Google: A text retrieval approach to object matching in videos. In *Proceedings of the International Conference on Computer Vision*, volume 2, pages 1470–1477, Oct. 2003. 5

[15] C. Sminchisescu, A. Kanaujia, Z. Li, and D. Metaxas. Conditional models for contextual human motion recognition. In *IEEE International Conference on Computer Vision*, pages 1808–1815, 2005. 1, 2

[16] B. Taskar, C. Guestrin, and D. Koller. Max-margin Markov networks. In S. Thrun, L. Saul, and B. Schölkopf, editors, *Advances in Neural Information Processing Systems 16*, pages 25–32, Cambridge, MA, 2004. MIT Press. 2, 3

[17] I. Tsochantaridis, T. Joachims, T. Hofmann, and Y. Altun. Large margin methods for structured and interdependent output variables. *J. Mach. Learn. Res.*, 6:1453–1484, 2005. 2, 3, 4

[18] V. Vapnik. *The Nature of Statistical Learning Theory*. Springer, New York, 1995. 2

[19] L. Wang and D. Suter. Recognizing human activities from silhouettes: Motion subspace and factorial discriminative graphical model. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pages 1–8, 2007. 1, 2

[20] J. Yamato, J. Ohya, and K. Ishii. Recognizing human action in time-sequential images using hidden markov model. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pages 379–385, 1992. 2