Substructure and Boundary Modeling for Continuous Action Recognition

Zhaowen Wang[†] Jinjun Wang[‡] Jing Xiao[‡] Kai-Hsiang Lin[†] Thomas Huang[†]
[†]Beckman Institute
University of Illinois at Urbana-Champaign
{wang308, klin21, huang}@ifp.uiuc.edu

[‡] Algorithm Group, ASD
Epson Research and Development, Inc.
{jwang, xiaoj}@erd.epson.com

Abstract

This paper introduces a probabilistic graphical model for continuous action recognition with two novel components: substructure transition model and discriminative boundary model. The first component encodes the sparse and global temporal transition prior between action primitives in state-space model to handle the large spatial-temporal variations within an action class. The second component enforces the action duration constraint in a discriminative way to locate the transition boundaries between actions more accurately. The two components are integrated into a unified graphical structure to enable effective training and inference. Our comprehensive experimental results on both public and in-house datasets show that, with the capability to incorporate additional information that had not been explicitly or efficiently modeled by previous methods, our proposed algorithm achieved significantly improved performance for continuous action recognition.

1. Introduction

Understanding continuous human activities from videos, *i.e.* simultaneous segmentation and classification of actions, is a fundamental yet challenging problem in computer vision. Many existing works approach the problem using bottom-up methods [31], where segmentation is performed as preprocessing to partition videos into coherent constituent parts, and action recognition is then applied as an isolated classification step. Although a rich literature exists for segmentation of time series, such as change point detection [12], periodicity of cyclic events modeling [7] and frame clustering [40], the methods tend to detect local boundaries and lack the ability to incorporate global dynamics of temporal events, which leads to under or over segmentation that severely affects the recognition performance, especially for complex actions with diversified local motion statistics [13].

The limitation of the bottom-up approaches has been addressed by performing concurrent top-down recognition using variants of Dynamic Bayesian Network (DBN), where the dynamics of temporal events are modeled as transitions in a latent [24, 18] or partially observed state space [14, 27]. The technique has been successfully used in speech recognition and natural language processing, while the performance of existing DBN based approaches for action recognition [27, 10, 32, 33, 17, 26] tends to be relatively lower [13], mostly due to the difficulty in interpreting the physical meaning of latent states. Thus, it becomes difficult to impose additional prior knowledge with clear physical meaning into an existing graphical structure to further improve its performance.

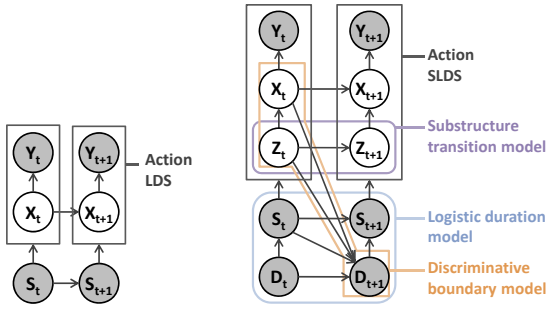


Figure 1. (a) Tradition SLDS model for continuous action recognition, where each action is represented by an LDS; (b) the structure of our proposed model, in which each action is represented by an SLDS with substructure transition, and the inter action transition is by controlled by discriminative boundary model.

To tackle the problem, in this paper, we show how two additional sources of information with clear physical interpretations can be considered in a general graphical structure for State-Space Model (SSM) in Figure 1. Compared to a standard Switching Linear Dynamic System (SLDS) [27] model in Figure 1.(a), where X, Y and S are respectively the hidden state, observation and label, the proposed model in Figure 1.(b) is augmented with two additional nodes, Z and D, to describe the substructure transition and duration statistics of actions.

Substructure transition: Rather than a uniform mo-

tion type, a real-world human action is usually characterized by a set of inhomogeneous units with some instinct structure, which we call *substructure*. Action substructure arises from two factors: (1) the hierarchical nature of human activity, where one action can be temporally decomposed into a series of primitives with spatial-temporal constraints; (2) the large variance of action dynamics due to differences in kinematical property of subjects, feedback from environment, or interaction with objects. For the first factor, Hoai *et al*. [13] used multi-class Support Vector Machine (SVM) with Dynamic Programming to recognize coherent motion constituent parts in an action; Liu *et al*. [22] applied latent-SVM for temporal evolving of "attributes" in actions; Sung *et al*. [33] introduced a two-layer Maximum Entropy Markov Models to recognize the correspondence between sub-activities and human skeletal features. For the second factor, considerations have been paid to the substructure variance caused by subject-object interaction using Connected Hierarchic Conditional Random Field (CRF) [17], and the substructure variance caused by pose using Latent Pose CRF [26].

In more general cases, Morency *et al*. presented the Latent Dynamic CRF (LDCRF) algorithm by adding a "latent-dynamic" layer into CRF for hidden substructure transition [25]. The limitation of CRF as a discriminative method is that, one single pseudo-likelihood score is estimated for an entire sequence which is incapable to interpret the probability of each individual frame. To solve the problem, we instead design a generative model as in Figure.1.(b), with extra hidden node Z gating the transition amongst a set of dynamic systems, and the posterior for every action can be inferred strictly under Bayesian framework for each frame. The dimension of state space increases geometrically with an extra hidden node, so we introduce effective transition prior constraints in Section 2 to avoid over-fitting on a limited amount of training data.

Duration model: The duration statistics of actions is important in determining the boundary where one action transits to another in continuous recognition tasks. Duration model has been widely adopted in Hidden Markov Model (HMM) based methods, such as the explicit duration HMM [9] or more generally the Hidden Semi Markov Model (HSMM) [39]. Incorporating duration model into SSM is more challenging than HMM because SSM has continuous state space, and exact inference in SSM is usually intractable [20]. Some works reported in this line include Cemgil *et al*. [5] for music transcription and Chib and Dueker [6] for economics. Oh *et al*. [28] imposed the duration constraint at the top level of SLDS and achieved improved performance for honeybee behavior analysis [27]. In general, naive integration of duration model into SSM is not effective, because duration patterns vary significantly across visual data and limited training samples may bias the model with incorrect duration patterns.

To address this problem, in Figure 1.(b) we correlate duration node D with the continuous hidden state node X and the substructure transition node Z via logistic regression as explained in Section 3. In this way, the proposed duration model becomes more discriminative than conventional generative models, and the data-driven boundary locating process can accommodate more variation in duration length.

In summary, the major contribution of the paper is to incorporate two additional models into a general SSM, namely the Substructure Transition Model (STM) and the Discriminative Boundary Model (DBM). We also design a Rao-Blackwellised particle filter for efficient inference of proposed model in Section 4. Experiments in Section 5 demonstrate the superior performance of our proposed system over several existing state-of-the-arts in continuous action recognition. Conclusion is drawn in Section 6.

2. Substructure Transition Model

Linear Dynamic Systems (LDS) is the most commonly used SSM to describe visual features of human motions. LDS is modeled by linear Gaussian distributions:

$$p(Y_t = \mathbf{y}_t | X_t = \mathbf{x}_t) = \mathcal{N}(\mathbf{y}_t; \mathbf{B}\mathbf{x}_t, \mathbf{R}) \quad (1)$$

$$p(X_{t+1} = \mathbf{x}_{t+1} | X_t = \mathbf{x}_t) = \mathcal{N}(\mathbf{x}_{t+1}; \mathbf{A}\mathbf{x}_t, \mathbf{Q}) \quad (2)$$

where Y_t is the observation at time t, X_t is a latent state, $\mathcal{N}(\mathbf{x}; \mu, \mathbf{\Sigma})$ is multivariate normal distribution of $\mathbf{x}$ with mean μ and covariance $\mathbf{\Sigma}$. To consider multiple actions, SLDS [27] is formulated as a mixture of LDS's with the switching among them controlled by action class S_t. However, each LDS can only model an action with homogenous motion, ignoring the complex substructure within the action. We introduce a discrete hidden variable $Z_t \in \{1, ..., N_Z\}$ to explicitly represent such information, and the *substructured* SSM can be stated as:

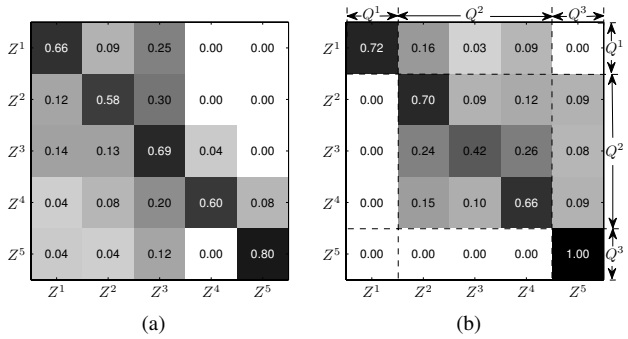
$$p(Y_t = \mathbf{y}_t | X_t = \mathbf{x}_t, S_t^i, Z_t^j) = \mathcal{N}(\mathbf{y}_t; \mathbf{B}^{ij}\mathbf{x}_t, \mathbf{R}^{ij}) \quad (3)$$

$$p(X_{t+1} = \mathbf{x}_{t+1} | X_t = \mathbf{x}_t, S_{t+1}^i, Z_{t+1}^j) = \mathcal{N}(\mathbf{x}_{t+1}; \mathbf{A}^{ij}\mathbf{x}_t, \mathbf{Q}^{ij}) \quad (4)$$

where $\mathbf{A}^{ij}$, $\mathbf{B}^{ij}$, $\mathbf{Q}^{ij}$, and $\mathbf{R}^{ij}$ are the LDS parameters for the j^{th} action primitive in the substructure of i^{th} action class. $\{Z_t\}$ is modeled as a Markov chain and the transition probability is specified by multinomial distribution:

$$p(Z_{t+1}^j | Z_t^i, S_{t+1}^k) = \theta_{ijk} \quad (5)$$

In the following, the term STM may refer to either the transition matrix in Eq. (5) or the overall substructured SSM depending on its context. Some examples of STM are given in Fig. 2, which are to be explained in detail in the remainder of this section.



(a) (b)

Figure 2. STM trained for action "move-arm" in stacking dataset using (a) sparse and (b) block-wise sparse constraints, with $N_Z = 5$ and $N_Q = 3$. STM in (b) better captures global ordering.

2.1. Sparsity Constrained STM

We use simplified notation $\mathbf{\Theta} = \{\theta_{ij}\}$ for the STM within a single action. An unconstrained $\mathbf{\Theta}$ implies that the substructure of action primitives can be organized in an arbitrary way. For most real-world human actions, however, there is a strong temporal ordering associated with the primitive units. Such order relationship can be vital to accurate action recognition as well as robust model estimation.

There have been some attempts to encode a fixed order relationship among primitive units by restricting the locations of non-zero elements in transition matrix $\mathbf{\Theta}$; examples include the left-to-right HMM [2], switching HMM (SHMM) [14], and factorial HMM [11]. In many cases, it is difficult to specify the temporal ordering *a priori*, and a more practical approach is to impose a sparse transition constraint while leaving the discovery of exact order relationship to training phase. Along this direction, negative Dirichlet distribution has been proposed in [4] as a prior for each row $\boldsymbol{\theta}_i$ in $\mathbf{\Theta}$: $p(\boldsymbol{\theta}_i) \propto \prod_j \theta_{ij}^{-\alpha}$, where α is a pseudo count penalty. The MAP estimation of parameter is

$$\hat{\theta}_{ij} = \frac{\max(\xi_{ij} - \alpha, 0)}{\sum_t \max(\xi_{it} - \alpha, 0)} \tag{6}$$

where ξ_{ij} is the sufficient statistics of $\langle Z_t^i, Z_{t+1}^j \rangle$. When the number of transitions from z^i to z^j in training data is less than α, the probability θ_{ij} is set to zero. The sparsity enforced in this way often leads to local transition patterns sensitive to noise and incomplete data, as shown in Fig. 2 (a). Also, the penalty term α introduces bias to the proportion of non-zero transition probabilities, *i.e.* $\frac{\hat{\theta}_{ij}}{\hat{\theta}_{ik}} \neq \frac{\xi_{ij}}{\xi_{ik}}$. This bias can be severe especially when ξ_{ij} is small.

2.2. Block-wise Sparse STM

For tradeoff between model sparsity and flexibility, we propose a block-wise sparse STM to regularize the global topology of action substructure. The idea is to divide an action into several stages and each stage comprises of a subset of action primitives. The transition between stages is encouraged to be sequential and sparse, such that the global action structure can be modeled. At the same time, the action primitives within each stage can propagate freely from one to another so that variation in action style and parameter is also preserved.

Formally, define discrete variable $Q_t \in \{1, ..., N_Q\}$ as the current stage index of action, and assume a surjective mapping $g(\cdot)$ is given which assigns each action primitive Z_t to its corresponding stage Q_t:

$$\begin{cases} p(Q_t^q, Z_t^i) > 0\,, & \text{if } g(i) = q \\ p(Q_t^q, Z_t^i) = 0\,. & \text{otherwise} \end{cases} \tag{7}$$

The choice of $g(\cdot)$ depends on the nature of action. Intuitively, we can assign more action primitives to a stage with diversified motion patterns and less action primitives to a stage with restricted pattern. The joint dynamic transition distribution of Q_t and Z_t is defined as:

$$p(Q_{t+1}, Z_{t+1}|Q_t, Z_t) = p(Q_{t+1}|Q_t)p(Z_{t+1}|Q_{t+1}, Z_t) \tag{8}$$

The second term of Eq. (8) specifies the transition between action primitives, which we want to keep as flexible as possible to model diversified local action patterns. The first term captures the global structure between different action stages, and therefore we impose an *ordered* negative Dirichlet distribution as its hyper-prior:

$$p(\mathbf{\Phi}) \propto \prod_{q \neq r, q+1 \neq r} \phi_{qr}^{-\alpha} \tag{9}$$

where $\mathbf{\Phi} = \{\phi_{qr}\}$ is the stage transition probability matrix, $\phi_{qr} = p(Q_{t+1}^r|Q_t^q)$, and α is a constant for pseudo count penalty. The ordered negative Dirichlet prior encodes both sequential order information and sparsity constraint. It promotes statistically a global transition path $Q^1 \rightarrow Q^2 \rightarrow ... \rightarrow Q^{N_Q}$ which can be learned from training data rather than heuristically defined as in left-to-right HMM [2]. An example of the resulting STM is shown in Fig. 2 (b). Note that no in-coming/out-going transition is encouraged for Q^1/Q^{N_Q}, which stands for starting/terminating stage. The identification of these two special stages is helpful for segmenting continuous actions, as will be discussed in Subsection 3.2.

2.3. Learning STM

The MAP model estimation requires to maximize the product of likelihood (8) and prior (9) under the constraint of (7). There are two interdependent nodes, Q and Z, involved in the optimization, which make the problem complicated. As shown in [36], Eq. (8) can be replaced with the transition distribution of single variable Z in Eq. (5), and a constraint exists for the relationship between $\mathbf{\Theta}$ and $\mathbf{\Phi}$. Therefore, the node Q (and the associated parameter $\mathbf{\Phi}$) serves only for conceptual purpose and can be eliminated in

final model construction. The MAP estimation can be converted to the following constrained optimization problem:

$$\max_{\mathbf{\Theta}} \quad \mathcal{L}(\mathbf{\Theta}) = \sum_{i,j} \xi_{ij} \log \theta_{ij} - \sum_{\substack{q \neq r \\ q+1 \neq r}} \alpha \log \phi_{qr} \quad (10)$$

$$\text{s.t.} \quad \phi_{qr} = \Sigma_{j \in \mathcal{G}(r)} \theta_{ij}, \quad i \in \mathcal{G}(q), \ \forall r$$
$$\Sigma_j \theta_{ij} = 1, \ \ \forall i \qquad \theta_{ij} \geq 0, \ \ \forall i, j$$

where ξ_{ij} is the sufficient statistics of $\langle Z_t^i, Z_{t+1}^j \rangle$, $\mathcal{G}(q) \triangleq \{i | g(i) = q\}$, and $\{\phi_{qr}\}$ are just auxiliary variables. The optimal solution is

$$\hat{\theta}_{ij} = \hat{\phi}_{g(i),g(j)} \frac{\xi_{ij}}{\sum_{j' \in \mathcal{G}(r)} \xi_{ij'}} \quad (11)$$

$$\hat{\phi}_{qr} = \frac{\max(\sum_{i \in \mathcal{G}(q), j \in \mathcal{G}(r)} \xi_{ij} - \alpha_{qr}, 0)}{\sum_{r'} \max(\sum_{i \in \mathcal{G}(q), j \in \mathcal{G}(r')} \xi_{ij} - \alpha_{qr'}, 0)}$$

where α_{qr} is equal to α if $q \neq r$ or $q+1 \neq r$, and 0 otherwise. As we can see, the resultant $\hat{\mathbf{\Theta}}$ is a block-wise sparse matrix, which can characterize both the global structure and local detail of action dynamics. Also, within each block (stage), there is no bias in $\hat{\theta}_{ij}$.

3. Discriminative Boundary Model

It is straightforward to use a Markov chain to model the transition of action S_t where $p(S_{t+1}^j | S_t^i) = a_{ij}$. The duration information of the i^{th} action is naively incorporated into its self-transition probability a_{ii}, which leads to an exponentially-distributed action duration model:

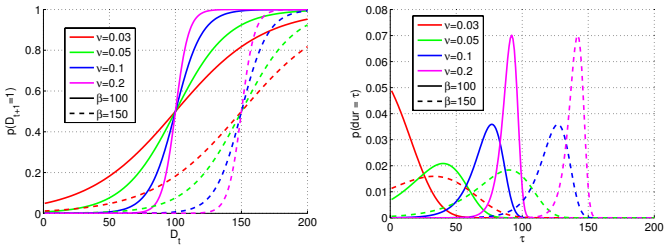
$$p(dur_i = \tau) = a_{ii}^{\tau - 1}(1 - a_{ii}), \quad \tau = 1, 2, 3...$$

Unfortunately, only a limited number of real-life events have an exponentially diminishing duration. Inaccurate duration modeling can severely affect our ability to segment consecutive actions and identify their boundaries.

Non-exponential duration distribution can be implemented with duration-dependent transition matrix, such as the one used in HSMM[39]. Fitting a transition matrix for each epoch within the maximum length of duration is often impossible given a limited number of training sequences, even when parameter hyperprior such as hierarchical Dirichlet distribution [35] is used to restrict model freedom. Parametric duration distributions such as gamma [21] and Gaussian [38] provide a more compact way to represent duration and show good performance in signal synthesis. However, they are less useful in inference because the corresponding transition probability is not easy to evaluate.

3.1. Logistic Duration Model

Here a new logistic duration model is proposed to overcome the above limitations. We introduce a variable D_t to represent the length of time current action has been lasting.



(a) (b)

Figure 3. (a) Resetting probability $p(D_{t+1} = 1 | D_t, S_t)$ and (b) duration distribution for logistic duration model. Plotted with different color/line style for different ν/β.

$\{D_t\}$ is a counting process starting from 1, and the beginning of a new action is triggered whenever it is reset to 1:

$$p(S_{t+1}^j | S_t^i, D_{t+1}^d) = \begin{cases} \delta(j - i), & \text{if } d > 1 \\ a_{ij}, & \text{if } d = 1 \end{cases} \quad (12)$$

where a_{ij} is the probability of transiting from previous action i to new action j. Notice that the same type of action can be repeated if we have $a_{ii} > 0$.

Instead of modeling action duration distribution directly, we model the transition distribution of D_t as a logistic function of its previous value:

$$p(D_{t+1}^c | S_t^i, D_t^d) = \frac{e^{\nu_i (d - \beta_i)} \delta(c - 1) + \delta(c - d - 1)}{1 + e^{\nu_i (d - \beta_i)}} \quad (13)$$

where ν_i and β_i are positive logistic regression weights. Eq. (13) immediately leads to the duration distribution for action class i:

$$p(dur_i = \tau) = \prod_{d=1}^{\tau} \frac{1}{1 + e^{\nu_i (d - \beta_i)}} \times e^{\nu_i (\tau - \beta_i)} \quad (14)$$

Fig. 3 (a) shows how the resetting probability of D_{t+1} changes as a function of D_t with different parameter sets, and the corresponding duration distributions are plotted in (b). The increasing probability of transiting to a new action leads to a peaked duration distribution, with center and width controlled by β_i and ν_i, respectively.

3.2. Discriminative Boundary Model

Stacking the logistic duration layer (D-S) onto the STM layer (Z-X-Y) simply leads to a generative SSM, which is unable to utilize contextual information for accurate action boundary segmentation. Discriminative graphic models, such as MEMM [24] and CRF [18], are generally more powerful in such classification problem except that they ignore data likelihood or suffer from label bias problem.

To integrate discriminative power into our action boundary model and at the same time keep the generative nature of the action model itself, we construct DBM by further augmenting the duration dependency with the contextual infor-

mation from latent states X and Z:

$$p(D^1_{t+1}|S^i_t, D^d_t, X^{\mathbf{x}}_t, Z^j_t) = \frac{e^{\nu_i(d-\beta_i)+\boldsymbol{\omega}^T_{ij}\mathbf{x}}}{1+e^{\nu_i(d-\beta_i)+\boldsymbol{\omega}^T_{ij}\mathbf{x}}} \quad (15)$$

where ν_i, β_i have the same interpretation as in Eq. (13), and $\boldsymbol{\omega}_{ij}$ are the additional logistic regression coefficients. When $\boldsymbol{\omega}^T_{ij}\mathbf{x} = 0$, no information can be learned from X_t and Z_t, and the DBM reduces to a generative model as Eq. (13). A similar logistic function has been employed in augmented SLDS [3], where the main motivation is to distinguish between transitions to different states based on latent variable. Our DBM is specifically designed for locating the boundary between contiguous actions. It relies on both real valued and categorical inputs.

As constrained by the STM in Subsection 2.2, each action is only likely to terminate in stage N_Q. Therefore, D_{t+1} can be reset to 1 only when the current action is in this terminating stage, and we can modify Eq. (15) as:

$$p(D^1_{t+1}|S^i_t, D^d_t, X^{\mathbf{x}}_t, Z^j_t) = \begin{cases} \text{Eq. (15)}, & g(j) = N_Q \\ 0, & \text{otherwise} \end{cases} \quad (16)$$

In this way, the number of parameters is greatly reduced and the label unbalance problem is also ameliorated. Now, the construction of our model for continuous action recognition has been completed, with the overall structure shown in Figure 1 (b).

3.3. Learning DBM

To learn the parameters ν, β and $\boldsymbol{\omega}$, we use coordinate descent method to iterate between $\{\nu, \beta\}$ and $\boldsymbol{\omega}$. For ν and β, given a set of training state sequences $\{\mathbf{S}_n\}$, we can easily obtain the labels for all $\{\mathbf{D}_n\}$ according to Eq. (12) and (13). Then fitting the logistic duration model of Eq. (13) equals to performing logistic regression with input feature $x = D_t$ and output $y = \delta(S_{t+1} - S_t)$. The action transition probability $\{a_{ij}\}$ can be obtained trivially.

To estimate $\boldsymbol{\omega}_{ij}$, let $\{T^{(n)}\}_{n=1\ldots N}$ be our training set, where each data sample $T^{(n)}$ is a realization of all the nodes involved in Eq. (15) at a particular time instance $t^{(n)}$ and $S_{t^{(n)}} = i$. Since $X_{t^{(n)}}$ and $Z_{t^{(n)}}$ are hidden variables, their posterior $p(Z^j_{t^{(n)}}|\cdot) = p^{(n)}_Z$ and $p(X^{\mathbf{x}}_{t^{(n)}}|Z^j_{t^{(n)}}, \cdot) = \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}^{(n)}, \boldsymbol{\Sigma}^{(n)})$ are first inferred from single action STM, where the posterior of $X_{t^{(n)}}$ is approximated by a Gaussian. The estimation of $\hat{\boldsymbol{\omega}}_{ij}$ is obtained by maximizing the expected log likelihood:

$$\max_{\boldsymbol{\omega}_{ij}} \sum_n \mathrm{E}_{p(X^{\mathbf{x}}_{t^{(n)}}, Z^j_{t^{(n)}}|\cdot)}\left[\log l^{(n)}(\mathbf{x}, \boldsymbol{\omega}_{ij})\right] \quad (17)$$

$$= \max_{\boldsymbol{\omega}_{ij}} \sum_n p^{(n)}_Z \int_{\mathbf{x}} \log l^{(n)}(\mathbf{x}, \boldsymbol{\omega}_{ij}) \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}^{(n)}, \boldsymbol{\Sigma}^{(n)}) d\mathbf{x}$$

where

$$l^{(n)}(\mathbf{x}, \boldsymbol{\omega}) = \frac{e^{(c^{(n)}+\boldsymbol{\omega}^T\mathbf{x})b^{(n)}}}{1+e^{c^{(n)}+\boldsymbol{\omega}^T\mathbf{x}}} \quad (18)$$

and $b^{(n)} = p(D_{t^{(n)}+1} = 1)$, $c^{(n)} = \nu_i(D_{t^{(n)}} - \beta_i)$. The integral in Eq. (17) cannot be solved analytically. Instead, we use unscented transform [15] to approximate the Gaussian $\mathcal{N}(\mathbf{x}; \boldsymbol{\mu}^{(n)}, \boldsymbol{\Sigma}^{(n)})$ using a set of sigma points $\{\mathbf{x}^{(n)}_k\}_{k=0\ldots 2M}$. Therefore, Eq. (17) converts to a weighted logistic regression problem with features $\{\mathbf{x}^{(n)}_k\}$, labels $\{b^{(n)}\}$ and weights $\{p^{(n)}_Z/(2M+1)\}$.

4. Rao-Blackwellised Particle Filter Inference

In testing, given an observation sequence $\mathbf{y}_{1:T}$, we want to find the MAP action labels $\hat{S}_{1:T}$ and the boundaries defined by $\hat{D}_{1:T}$; we are also interested in the style of actions which can be revealed from $\hat{Z}_{1:T}$. Evaluating the full posterior $p(S_{1:T}, D_{1:T}, Z_{1:T}|\mathbf{y}_{1:T})$ is a non-trivial job given the complex hierarchy of our model. We propose to use particle filtering [1] for online inference due to its capability in non-linear scenario and temporal scalability. Note the latent variable X_t can be marginalized by Rao-Blackwellisation [8], and the computation of particle filtering is significantly reduced since Monte Carlo sampling is only conducted in the joint space of (S_t, D_t, Z_t) which has a low dimension and highly compact support.

Formally, we decompose the posterior distribution of all the hidden nodes at time t as

$$\begin{aligned} & p(S_t, D_t, Z_t, X_t|\mathbf{y}_{1:t}) \\ = \; & p(S_t, D_t, Z_t|\mathbf{y}_{1:t}) p(X_t|S_t, D_t, Z_t, \mathbf{y}_{1:t}) \end{aligned} \quad (19)$$

In Rao-Blackwellised particle filter [16], a set of N_P samples $\{(s^{(n)}_t, d^{(n)}_t, z^{(n)}_t)\}^{N_P}_{n=1}$ and the associated weights $\{w^{(n)}_t\}^{N_P}_{n=1}$ are used to approximate the intractable first term in Eq. (19), while the second term is represented by $\{\chi^{(n)}_t(\mathbf{x})\}^{N_P}_{n=1}$, which are analytical distributions conditioned on corresponding samples:

$$\chi^{(n)}_t(\mathbf{x}) \triangleq p(X_t = \mathbf{x}|s^{(n)}_t, d^{(n)}_t, z^{(n)}_t, \mathbf{y}_{1:t}) \quad (20)$$

In our model, $\chi^{(n)}_t(\mathbf{x}) = \mathcal{N}(\mathbf{x}; \hat{\mathbf{x}}^{(n)}_t, \mathbf{P}^{(n)}_t)$ is a Gaussian distribution. Thus, the posterior can be represented as

$$\begin{aligned} & p(S_t, D_t, Z_t, X_t|\mathbf{y}_{1:t}) \\ \approx & \sum_{n=1}^{N_P} w^{(n)}_t \delta_{S_t}(s^{(n)}_t) \delta_{D_t}(d^{(n)}_t) \delta_{Z_t}(z^{(n)}_t) \chi^{(n)}_t(\mathbf{x}) \end{aligned} \quad (21)$$

where the approximation error approaches to zero as N_P increases to infinite.

Given the samples $\{(s^{(n)}_{t-1}, d^{(n)}_{t-1}, z^{(n)}_{t-1}, \chi^{(n)}_{t-1}(\mathbf{x}))\}$ and weights $\{w^{(n)}_{t-1}\}$ at time $t-1$, the posterior of (S_t, D_t, Z_t) at time t can be evaluated as

$$\begin{aligned} p(S_t, D_t, Z_t|\mathbf{y}_{1:t}) \propto & \sum_n w^{(n)}_{t-1} p(S_t|D_t, s^{(n)}_{t-1}) \\ & \times p(Z_t|S_t, D_t, z^{(n)}_{t-1}) \mathcal{L}^{(n)}_t(S_t, D_t, Z_t) \end{aligned} \quad (22)$$

where

$$\mathcal{L}_t^{(n)}(S_t, D_t, Z_t) = \int p(\mathbf{y}_t|\mathbf{x}_{t-1}, S_t, Z_t)\chi_{t-1}^{(n)}(\mathbf{x}_{t-1}) \times p(D_t|s_{t-1}^{(n)}, d_{t-1}^{(n)}, z_{t-1}^{(n)}, \mathbf{x}_{t-1})d\mathbf{x}_{t-1} \quad (23)$$

Eq. (23) is essentially the integral of a Gaussian function with a logistic function. Although not analytically solvable, it can be well approximated by a re-parameterized logistic function according to [23]. Nevertheless, it is hard to draw sample from Eq. (23). Therefore, we draw new samples $(s_t^{(n)}, d_t^{(n)}, z_t^{(n)})$ from a proposal density defined as:

$$q(S_t, D_t, Z_t|\cdot) = p(S_t|D_t, s_{t-1}^{(n)})p(Z_t|S_t, D_t, z_{t-1}^{(n)}) \times p(D_t|s_{t-1}^{(n)}, d_{t-1}^{(n)}, z_{t-1}^{(n)}, \hat{\mathbf{x}}_{t-1}^{(n)}) \quad (24)$$

The new sample weights are then updated as

$$w_t^{(n)} \propto w_{t-1}^{(n)} \frac{\mathcal{L}_t^{(n)}(s_t^{(n)}, d_t^{(n)}, z_t^{(n)})}{p(d_t^{(n)}|s_{t-1}^{(n)}, d_{t-1}^{(n)}, z_{t-1}^{(n)}, \hat{\mathbf{x}}_{t-1}^{(n)})} \quad (25)$$

Once we get $s_t^{(n)}$ and $z_t^{(n)}$, $\chi_t^{(n)}(\mathbf{x})$ is simply updated by Kalman filter. Re-sampling and normalization procedures are applied after all the samples are updated as in [8].

5. Experimental Results

Our model is tested on four datasets for continuous action recognition. In all the experiments, we have used parameters $N_Q = 3$, $N_Z = 5$, $N_P = 200$. First STM is trained independently for each action using the segmented sequences in training set; then DBM is learned from the inferred terminal stage of each sequence. The overall learning procedure follows EM paradigm where the beginning and terminating stages are initially set as the first and last 15% of each sequence, and the initial action primitives are obtained from K-means clustering. The EM iteration stops when the change in likelihood falls below a threshold. In testing, after the online inference using particle filter, we further adjust each action boundary using an off-line inference within a local neighborhood of length 40 centered at the initial boundary; in this way, the locally "full" posterior in Section 4 is considered. We evaluate the recognition performance by per-frame accuracy. Contribution from each model component (STM and DBM) is analyzed separately.

5.1. Public Datasets

The first public dataset used is the IXMAS dataset [37]. The dataset contains 11 actions, each performed 3 times by 10 actors. The videos are acquired using 5 synchronized cameras from different angles, and the actors freely changed their orientation in acquisition. We calculate dense optical flow in the silhouette area of each subject, from which Locality-constrained Linear Coding features (LLC)[1] [34] are extracted as the observation in each frame. We have used 32 codewords and 4×4, 2×2 and 1×1 spatial pyramid [19]. Table 1 reports the continuous action recognition results, in comparison with SLDS[2] [27], CRF[1] [18] and LDCRF[1] [25]. Our proposed model (and each of its components) achieves a recognition accuracy higher than all the other methods by more than 10%.

Table 1. Continuous action recognition for IXMAS dataset

SLDS	CRF	LDCRF	STM	DBM	STM+DBM
53.6%	60.6%	57.8%	70.2%	74.5%	**76.5**%

Table 2. Continuous action recognition for CMU MoCap dataset

SLDS	CRF	LDCRF	[29]	[30]
80.0%	77.2%	82.5%	72.3%	90.9%

STM	DBM	STM+DBM
81.0%	**93.3**%	92.1%

The second public dataset used is the CMU MoCap dataset [3]. For comparison purpose, we report the results from the complete subset of subject 86. The subset has 14 sequences with 122 actions in 8 category. Quaternion feature is derived from the raw MoCap data as our observation for inference. Table 2 lists the continuous action recognition results, in comparison with the same set of benchmark techniques as in the first experiment, as well as [29, 30]. Similarly, results from this experiment demonstrated the superior performance of our method. It is interesting to note that, in Table 2, the frame-level accuracy by using DBM alone is a little higher than its combination with STM. This is because there's only one subject in this experiment and no significant variation in substructure is presented in each action type, so temporal duration plays a more important role in recognition. Nevertheless, the result attained by STM+DBM is superior than all benchmark methods.

5.2. In-house Datasets

In addition to the above two public datasets, two in-house datasets were also captured. The actions in these two sets feature stronger hierarchical substructure. The first dataset contains videos of stacking/unstacking three colored boxes, which involves actions of "move-arm", "pick-up" and "put-down". 13 sequences with 567 actions were recorded in both RGB and depth videos with one Microsoft Kinect sensor [4] (Fig. 4). Then object tracking and 3-D reconstruction were performed to obtain the 3D trajectories of two hands and three boxes. In this way an observation sequence in $\mathbb{R}^{15}$ is generated. In the experiments, leave-one-out cross-validation was performed on the 13 sequences. The continuous recognition results are listed in Table 3. It is noticed that, among the four benchmark techniques, the performance of SLDS and CRF are comparable, while LDCRF achieved the best performance. This is reasonable because

[1] implementation from author's website

[2] implementation based on BNT from http://code.google.com/p/bnt/

[3] http://mocap.cs.cmu.edu/

[4] http://www.xbox.com/kinect

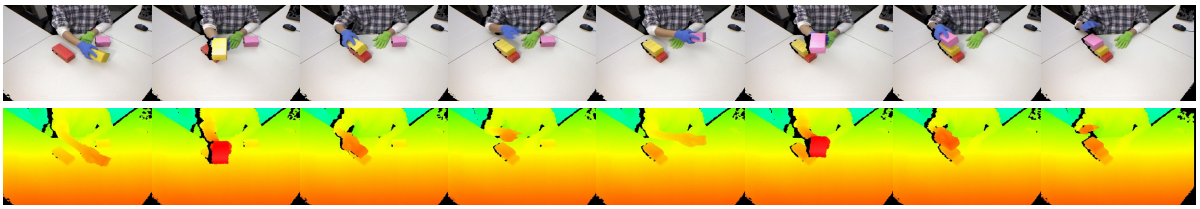


Figure 4. Example frames from the "stacking" dataset. *top-row: RGB images, bottom-row: aligned depth images.*

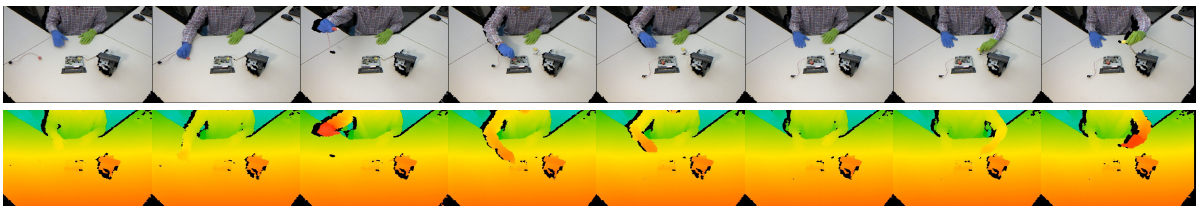


Figure 5. Example frames from the "assembling" dataset. *top-row: RGB images, bottom-row: aligned depth images.*

Table 3. Continuous action recognition for Set I: Stacking

SLDS	CRF	LDCRF	STM	DBM	STM+DBM
64.4%	79.6%	90.3%	88.5%	81.3%	**94.4%**

Table 4. Continuous action recognition for Set II: Assembling

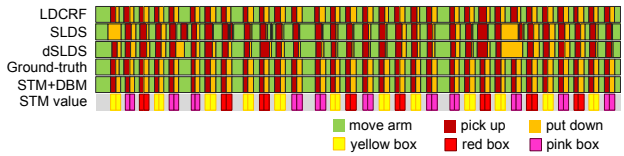
SLDS	CRF	LDCRF	STM	DBM	STM+DBM
68.2%	77.7%	88.5%	88.7%	69.0%	**92.9%**

during the stacking process, each box can be moved/stacked at any place on the desk, which leads to large spatial variations that cannot be well modeled by a Bayesian Network of only two layers. LDCRF applied a third layer to capture such "latent dynamics", and hence achieved best accuracy. For our proposed models, the STM alone brings SLDS to a comparable accuracy to LDCRF because it also models the action substructure. By further incorporating duration information, our model outperforms all benchmark approaches.

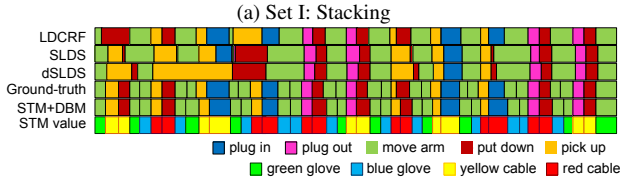
The second in-house dataset is more complicated than the first one. It involves five actions, "move-arm", "pick-up", "put-down", "plug-in" and "plug-out", in a printer part assembling task (Fig. 5). The 3D trajectories of two hands and two printer parts were extracted using the same Kinect sensor system. 8 sequences were recorded and tested with leave-one-out cross-validation. As can be seen from Table 4, our proposed model with both STM and DBM outperforms other benchmark approaches by a large margin.

5.3. Discussion

To provide more insightful comparison between the proposed algorithm and other benchmark algorithms, we show two examples of continuous action recognition results from the in-house datasets in Fig. 6. The result given by SLDS contains short and frequent switchings between incorrect action types. This is caused by the false matching of motion patterns to an incorrect action model. dSLDS [28] and LDCRF eliminate the short transitions by considering additional context information; however, their performances degrade severely around noisy or ambiguous action periods (*e.g.* the beginning of the sequence in Fig. 6.(b)) due



(a) Set I: Stacking



(b) Set II: Assembling

Figure 6. Continuous recognition for in-house datasets

to false duration prior or overdependence on discriminative classifier. Our proposed STM+DBM approach does not suffer from any of these problems, because STM helps to identify all action classes disregarding their variations, and DBM further helps to improve the precision of boundaries with both generative and discriminative duration knowledge. Another interesting finding shown in the last rows of (a) and (b) is that the substructure node Z can be interpreted by concrete physical meanings. For all the actions in these experiments, we find different object involved in an action corresponds to a different value of Z that has the highest probability in the inferred values $\hat{Z}_{1:T}$. Therefore, in addition to estimating action class, we can also find the object associated with the action by majority voting based on $\hat{Z}_{1:T}$. In our experiments, all the inferred object identities agree with ground truth.

6. Conclusion and Future Work

In this paper, we introduce an improved SSM with two added layers modeling the substructure transition dynamics and duration distribution for human action. The first layer encodes the sparse and global temporal transition structure of action primitives; and the second layer exploits discrimi-

native information to discover action boundaries adaptively. Experimental results validate the effectiveness of both two layers in continuous action recognition. As future work we plan to apply our model to actions in less constrained scenarios and use more advanced low-level descriptors to deal with unreliable observations.

Acknowledgments: This work was initialized and developed in Epson R&D Inc.. The authors of UIUC were supported in part by ONR Grant N000141210122, and by U.S. ARL and ARO under grant number W911NF-09-1-0383.

References

[1] M. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filters for on-line nonlinear/non-Gaussian Bayesian tracking. *IEEE Trans. on Signal Processing*, 50(2):174–188, 2002. 5

[2] R. Bakis. Continuous speech word recognition via centisecond acoustic states. *unpublished paper presented at the meeting of the Acoustics Society of America*, 1976. 3

[3] D. Barber. Expectation correction for smoothed inference in switching linear dynamical systems. *Journal of Machine Learning Research*, pages 2515–2540, 2006. 5

[4] M. Bicego, M. Cristani, and V. Murino. Sparseness achievement in hidden Markov models. *Proc.of ICIAP'07*, pages 67–72, 2007. 3

[5] A. Cemgil, H. Kappen, and D. Barber. A generative model for music transcription. In *IEEE Transactions on Audio, Speech, and Language Processing*, pages 679–694, 2006. 2

[6] S. Chib and M. J. Dueker. Non-Markovian regime switching with endogenous states and time-varying state strengths. Econometric Society 2004 North American Summer Meetings 600, Econometric Society, 2004. 2

[7] R. Cutler and L. Davis. Robust real-time periodic motion detection, analysis, and applications. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 22(8):781–796, 2000. 1

[8] A. Doucet, N. d. Freitas, K. P. Murphy, and S. J. Russell. Rao-Blackwellised particle filtering for dynamic Bayesian networks. In *Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence*, pages 176–183, 2000. 5, 6

[9] J. Ferguson. Variable duration models for speech. *Symp. Application of Hidden Markov Models to Text and Speech, Institute for Defense Analyses, Princeton, NJ*, pages 143–179, 1980. 2

[10] E. Fox, E. Sudderth, M. Jordan, and A. Willsky. Nonparametric Bayesian learning of switching linear dynamical systems. *Proc. of NIPS'09*, 2009. 1

[11] Z. Ghahramani and M. Jordan. Factorial hidden Markov models. *Proc.of NIPS*, 1996. 3

[12] Z. Harchaoui, F. Bach, and E. Moulines. Kernel changepoint analysis. *Proc. of NIPS'09*, 2009. 1

[13] M. Hoai, Z. Lan, and F. Torre. Joint segmentation and classification of human action in video. *Proc. of CVPR'11*, 2011. 1, 2

[14] M. Hoffken, D. Oberhoff, and M. Kolesnik. Switching hidden Markov models for learning of motion patterns in videos. *Proc. of ICANN'09*, pages 757–766, 2009. 1, 3

[15] S. Julier and J. Uhlmann. A new extension of the Kalman filter to nonlinear systems. *Proc of AeroSense: The 11th International Symposium on Aerospace/Defence Sensing, Simulation and Control*, pages 182–193, 1997. 5

[16] Z. Khan, T. Balch, and F. Dellaert. A Rao-Blackwellized particle filter for EigenTracking. In *Proceedings of the IEEE computer society conference on Computer vision and pattern recognition*, pages 980–987, 2004. 5

[17] H. Kjellstrom, J. Romero, D. Martinez, and D. Kragic. Simultaneous visual recognition of manipulation actions and manipulated objects. *Proc. of ECCV'08*, 2008. 1, 2

[18] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: probabilistic models for segmenting and labeling sequence data. *Proc. of ICML'01*, 2001. 1, 4, 6

[19] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: spatial pyramid matching for recognizing natural scene categories. *Proc. of CVPR'06*, 2006. 6

[20] U. Lerner and R. Parr. Inference in hybrid networks: theoretical limits and practical algorithms. In *In UAI*, pages 310–318, 2001. 2

[21] S. Levinson. Continuously variable duration hidden Markov models for automatic speech recognition. *Comput. Speech Lang.*, pages 29–45, 1986. 4

[22] J. Liu, B. Kuipers, and S. Savarese. Recognizing human actions by attributes. *Proc. of CVPR'11*, 2011. 2

[23] P. Maragakis, F. Ritort, C. Bustamante, M. Karplus, and G. E. Crooks. Bayesian estimates of free energies from nonequilibrium work data in the presence of instrument noise. *Journal of Chemical Physics*, 129, 2008. 6

[24] A. McCallum, D. Freitag, and F. Pereira. Maximum entropy Markov models for information extraction and segmentation. *Proc. of ICML'00*, pages 591–598, 2000. 1, 4

[25] L. Morency, A. Quattoni, and T. Darrell. Latent-dynamic discriminative models for continuous gesture recognition. *Proc. of CVPR'07*, 2007. 2, 6

[26] H. Ning, W. Xu, Y. Gong, and T. Huan. Latent pose estimator for continuous action recognition. *Proc. of ECCV'08*, 2008. 1, 2

[27] S. Oh, J. Rehg, T. Balch, and F. Dellaert. Learning and inference in parametric switching linear dynamic systems. *Proc. of ICCV'05*, 2005. 1, 2, 6

[28] S. M. Oh, J. M. Rehg, and F. Dellaert. Parameterized duration modeling for switching linear dynamic systems. *Proc. of CVPR'06*, pages 1–8, 2006. 2, 7

[29] N. Ozay, M. Sznaier, and C. O. Sequential sparsificarion for change detection. *Proc. of CVPR'08*, 2008. 6

[30] M. Raptis, K. Wnuk, and S. Soatto. Spike train driven dynamical models for human actions. *Proc. of CVPR'10*, pages 2077–2084, 2010. 6

[31] S. Satkin and M. Hebert. Modeling the temporal extent of actions. *Proc. of ECCV'10*, 2010. 1

[32] C. Sminchisescu, A. Kanaujia, Z. Li, and D. Metaxas. Conditional models for contextual human motion recognition. *Proc. of ICCV'05*, 2005. 1

[33] J. Sung, C. Ponce, B. Selman, and A. Saxena. Human activity detection from RGBD images. *Proc. of AAAI'11*, 2011. 1, 2

[34] J. Wang, J. Yang, F. Lv, and K. Yu. Locality-constrained linear coding for image classification. *Proc. of CVPR'10*, 2010. 6

[35] Z. Wang, E. Kuruoglu, X. Yang, Y. Xu, and S. Yu. Event recognition with time varying hidden Markov model. *Proc.of ICASSP'09*, pages 1761–1764, 2009. 4

[36] Z. Wang, J. Wang, J. Xiao, K.-H. Lin, and T. Huang. Substructure and boundary modeling for continuous action recognition. *Technical report,* http://arxiv.org/pdf/1203.1985v1, 2012. 3

[37] D. Weinland, R. Ronfard, and E. Boyer. Free viewpoint action recognition using motion history volumes. *Computer Vision and Image Understanding*, 2006. 6

[38] T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura. Duration modeling for HMM-based speech synthesis. *Proc. of ICSLP'98*, 1998. 4

[39] S.-Z. Yu. Hidden semi-Markov models. *Artificial Intelligence*, pages 215–243, 2010. 2, 4

[40] F. Zhou, F. Torre, and J. K. Hodgins. Aligned cluster analysis for temporal segmentation of human motion. *Proc. of IEEE Conference on Automatic Face and Gestures Recognition*, 2008. 1